

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA

**Método de Posicionamento e Dimensionamento
3D Baseado em Imagens Digitais**

Autor: **Marcelo Rudek**
Orientador: **Paulo Roberto Gardel Kurka**

62/2006

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
DEPARTAMENTO DE PROJETO MECÂNICO

Método de Posicionamento e Dimensionamento 3D Baseado em Imagens Digitais

Autor: **Marcelo Rudek**
Orientador: **Paulo Roberto Gardel Kurka**

Curso: Engenharia Mecânica
Área de Concentração: Projeto Mecânico

Tese de doutorado apresentada à comissão de Pós Graduação da Faculdade de Engenharia Mecânica, como requisito para a obtenção do título de Doutor em Engenharia Mecânica.

Campinas, 2006
S.P. – Brasil

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
DEPARTAMENTO DE PROJETO MECÂNICO

TESE DE DOUTORADO

Método de Posicionamento e Dimensionamento 3D Baseado em Imagens Digitais

Autor: **Marcelo Rudek**

Orientador: **Paulo Roberto Gardel Kurka**

A Banca Examinadora composta pelos membros abaixo aprovou esta Tese:

Prof. Dr. Paulo Roberto Gardel Kurka, Presidente
DPM/FEM/UNICAMP

Prof. Dr. Siome Klein Goldenstein
IC/UNICAMP

Prof. Dr. Luís Gonzaga Trabasso
ITA/São José dos Campos

Prof. Dr. Auteliano Antunes dos Santos Júnior
DPM/FEM/UNICAMP

Prof. Dr. Renato Pavanello
DMC/FEM/UNICAMP

Campinas, 14 de julho de 2006

Dedicatória

Para minha esposa Stela e meu filho João Victor

Agradecimentos

Eu gostaria de agradecer de forma especial as seguintes pessoas, que contribuíram com o desenvolvimento desta tese de doutorado.

Primeiro, agradeço a meu orientador Dr. Paulo Roberto Gardel Kurka que teve papel essencial para a conclusão desta tese, pela sua pronta colaboração através da troca de idéias e do compartilhamento de experiências e por ter mostrado os caminhos que levam à pesquisa e à produção de conhecimento. Agradeço também aos seus familiares pelo acolhimento recebido, durante a realização de reuniões de trabalho em sua casa.

À Pontifícia Universidade Católica do Paraná por permitir a utilização de equipamentos e laboratórios para a execução dos experimentos, e aos colegas professores do curso de Engenharia Mecatrônica e seus diretores, que sempre colaboraram com o progresso desta pesquisa. Em especial aos professores Osiris Canciglieri Junior e Leandro Coelho, pelo apoio e constante incentivo à realização deste projeto.

Gostaria de agradecer também aos meus pais Valdemiro e Marli pela ajuda prestada em todos os momentos em que estive ausente de casa, e a minha esposa Stela e meu filho João Victor por compreenderem o esforço necessário à conclusão deste trabalho.

A Deus, por permitir conhecer e trabalhar com estas pessoas, que eternamente farão parte de minha vida.

Tudo que sabemos fazer aprendemos fazendo.
(Aristóteles)

Resumo

Rudek, Marcelo, *Método de Posicionamento e Dimensionamento 3D Baseado em Imagens Digitais*, Campinas: Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2006. 165 p. Tese (Doutorado)

O trabalho apresenta um método baseado em imagens digitais para a determinação da posição espacial e do volume ocupado por um sólido. O objetivo é conhecer estes parâmetros sem a necessidade da reconstrução tridimensional completa do objeto. A aplicação da técnica proposta é baseada nas projeções da imagem do objeto a ser medido. O processo se baseia na utilização de uma caixa de contorno (*bounding box*) que pode ser ajustada sobre uma região de interesse na imagem. A posição desta caixa de contorno descreve a orientação espacial do objeto, e as suas dimensões espaciais definem o volume aproximado do sólido. O trabalho propõe um método para ajustar as arestas da caixa de contorno sobre as bordas do objeto, usando como referencial os pontos de fuga que formam os vértices da caixa de contorno. Os vértices formam um conjunto reduzido de pontos e a reconstrução tridimensional destes pontos permite definir um polígono retangular no espaço que representa as informações métricas do objeto real. O contexto do trabalho se situa no campo da visão computacional e processamento de imagens utilizando um conjunto de ferramentas aplicadas à teoria de reconstrução 3D. São apresentados os resultados obtidos da aplicação do método para imagens reais, bem como é descrita uma aplicação do método em um ambiente de manufatura integrada.

Palavras Chave

Imagens, reconstrução 3D, caixa de contorno

Abstract

Rudek, Marcelo, *3D Positioning and Dimensioning Method Based on Digital Images*, Campinas: Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, 2006. 165 p. Tese (Doutorado)

This work presents a method based on digital images to determine the spatial position and volume occupied by a solid. The objective is to find these parameters without a complete tri-dimensional reconstruction of the object. The application of this technique is based on the image projections of a solid target. The process is based on a contour bounding box that is adjusted around a region of interest. The box position describes the object spatial orientation and its spatial dimensions can define the approximated volume. The work proposes a method to adjust the contour box lines on the object borders using vanishing points as reference to form each vertex of the bounding box. The vertices form a reduced set of points and the 3D reconstruction of such points can define a rectangular polygon in the space representing the metric information about the real object. The work is within the context of computer vision and image processing by the use of a set of 3D reconstruction tools. Results obtained of the application of this method to real images are presented, as well as an application to an integrated manufacturing process.

Key Words

Images, 3D reconstruction, bounding box

Índice

Lista de Figuras	iv
Lista de Tabelas	viii
Nomenclatura	ix
1. Introdução	1
1.1. Proposta	5
1.2. Organização do texto	6
2. Representação Digital da Imagem	7
2.1. Formação da Imagem Digital	7
2.2. Características das Imagens	8
2.3. Modelo Geométrico da Câmera	10
2.3.1. Representação do Espaço Tridimensional	10
2.3.2. Representação do Corpo Rígido	11
2.3.3. Representação em Coordenadas Homogêneas	13
2.4. Modelos de Projeção	14
2.4.1. Projeção em perspectiva	14
2.5. Modelagem Geométrica da Câmera	16
2.5.1. Parâmetros da Câmera	18
2.5.1.1. Parâmetros Intrínsecos	18
2.5.1.2. Parâmetros Extrínsecos	21
2.6. Distorção Radial	23
2.7. Resumo do Capítulo	24

3. Reconstrução Tridimensional	25
3.1. Geometria de múltiplas vistas	26
3.2. Geometria Epipolar	27
3.3. Estimação da matriz essencial	29
3.3.1. Cálculo da Posição e Orientação	29
3.3.2. Algoritmo dos 8 pontos	31
3.4. Calibração de Câmeras	34
3.5. Resumo do Capítulo	36
4. Técnicas de Segmentação da Imagem	37
4.1. Transformação e Normalização	37
4.2. Limiarização	38
4.3. Detecção de Bordas	39
4.3.1. Erosão binária	39
4.3.2. Dilatação binária	41
4.3.3. Extração de contorno	42
4.4. Resumo do Capítulo	43
5. Caixas de contorno (<i>Bounding Box</i>)	44
5.1. Caixa de contorno alinhada (AABB)	46
5.2. Caixa de contorno orientada (OBB)	48
5.2.1. Análise dos Componentes Principais	49
5.2.2. Obtenção dos eixos principais planos pelos momentos de inércia	55
5.3. Resumo do capítulo	58
6. Método de Reconstrução Proposto	59
6.1. Procedimento de Calibração da Câmera	60
6.2. Estimação dos parâmetros de pose da câmera	63
6.3. Enquadramento da caixa de contorno por eixos alinhados (AABB)	65
6.3.1. Determinação dos Pontos de Fuga	65
6.3.2. Identificação da região de interesse	69
6.3.3. Aproximação das Faces	74
6.3.4. Resumo do capítulo	77
7. Aplicação do Método	78

7.1. Ajuste de uma caixa de contorno tipo AABB	78
7.2. Ajuste de uma caixa de contorno tipo OBB	88
7.2.2. Enquadramento de OBB pelas direções principais	94
7.2.2.1. Reconstrução espacial do centro de gravidade	98
7.2.2.2. Obtenção da caixa de contorno	102
7.3. Estimativa do volume	109
7.4. Resumo do capítulo	111
8. Aplicação do método em uma célula de manufatura	112
8.1. Descrição do funcionamento da estação de soldagem da célula de manufatura	112
8.2. Aplicação do método da caixa de contorno	116
8.3. Resumo do capítulo	122
9. Conclusão	124
Referências Bibliográficas	126
Anexo I	133
Anexo II	139
Anexo III	146

Lista de Figuras

Figura 2.1: Exemplo de uma imagem digital como uma função $f(x,y)$.	7
Figura 2.2: Identificação dos eixos do sistema cartesiano conforme a regra da mão direita.	10
Figura 2.3: Relação entre o sistema de coordenadas do mundo W e do objeto C .	12
Figura 2.4: Representação dos raios óticos através do orifício (<i>pinhole</i>).	14
Figura 2.5: Modelo da projeção perspectiva.	15
Figura 2.6: Projeção de um ponto P (3D) no plano de imagem Π (2D).	16
Figura 2.7: Relação geométrica para obtenção de p .	17
Figura 2.8: Transformação de coordenadas da imagem para coordenadas em <i>pixel</i> .	20
Figura 2.9: Relação entre coordenadas do mundo e da câmera adaptado de Trucco (1998).	22
Figure 2.10: (a) imagem original com distorção radial. (b) imagem corrigida.	23
Figura 3.1: Ambiente de visão com duas câmeras. A figura apresenta as projeções 2D p_1 e p_2 do ponto $P(X,Y,Z)$ em 3D.	26
Figura 3.2: Representação da geometria epipolar adaptado de Ma(2004).	28
Figura 3.3: Cubo com padrão xadrez utilizado para calibração de câmeras.	35
Figura 4.1: (a) imagem de exemplo $A_{M \times N}$; (b) elemento estruturante $B_{3 \times 3}$ em forma de cruz.	40
Figura 4.2: (a) imagem A original; (b) imagem A com colunas deslocadas à esquerda; (c) imagem A com colunas deslocadas à direita; (d) imagem A com linhas deslocadas para cima; (e) imagem A com colunas deslocadas para baixo.	40
Figura 4.3: (a) imagem A original; (b) imagem A com colunas deslocadas à esquerda; (c) imagem A com colunas deslocadas à direita; (d) imagem A com linhas deslocadas para cima; (e) imagem A com colunas deslocadas para baixo. (f) imagem dilatada	41
Figura 4.4: (a) Imagem resultante a partir da expressão 4.3. (b) imagem resultante a partir da expressão 4.4.	42
Figura 5.1: Exemplo de uma caixa de contorno ajustado a um objeto.	45

Figura 5.2: Esboço para um AABB com ajuste mínimo.	47
Figura 5.3: Esboço para um AABB sem ajuste mínimo.	47
Figura 5.4: Esboço de um OBB orientado segundo os eixos de simetria do objeto.	48
Figura 5.5: Representação gráfica das distâncias máxima e mínima dada pelos autovalores.	50
Figura 5.6: Representação do centro de massa.	51
Figura 5.7: Representação dos componentes principais ajustada à nuvem de pontos.	52
Figura 5.8: Seqüência de operações para obtenção dos eixos principais.	53
Figura 5.9: Geração dos eixos principais para uma imagem sintética.	54
Figura 5.10: Translação de eixos e localização do centróide cg do objeto.	55
Figura 5.11: Eixos principais de inércia.	56
Figura 6.1: Marcação dos eixos do plano de calibração.	60
Figura 6.2: Marcação dos cantos do <i>grid</i> de calibração.	61
Figura 6.3: Exemplo da detecção de cantos pelo método de Zhang (2000) para as duas vistas.	62
Figura 6.4: Caixa de calibração. a) imagem esquerda. b) imagem direita	64
Figura 6.5: Exemplo dos parâmetros de pose estimados para as imagens da figuras 6.4.	65
Figura 6.6: Transformação perspectiva do objeto e a representação do ponto de fuga Vp (vanishing point) sobre o plano da imagem. Ilustração adaptada de Rogers (1976).	66
Figura 6.7: Identificação das arestas e vértices da caixa virtual.	67
Figura 6.8: Posicionamento dos pontos de fuga F, dos vértices v, e das arestas k. As linhas escuras são as arestas conhecidas. Os pontos de fuga são obtidos a partir destas arestas.	68
Figura 6.9: Imagens de exemplo para determinação da posição e do volume do objeto. a) imagem esquerda. b) imagem direita	70
Figura 6.10: Imagem de teste (a) cena sem objeto a ser medido e (b) com o objeto a ser medido	71
Figura 6.11: Imagem resultante da subtração das imagens 6.10 (a) e (b).	72
Figura 6.12: Imagem resultante após aplicação de operações de erosão e dilatação binárias.	73
Figura 6.13: Imagem resultante final após a detecção de bordas.	73
Figura 6.14: Seleção da face do cubo a ser movida.	74
Figura 6.15: Ilustração do processo de aproximação da face às bordas do objeto.	75
Figura 6.16: Exemplo de ajuste da caixa de contorno ao objeto para o par de imagens (a) e (b).	76
Figura 7.1: Imagens do gabarito de calibração com as arestas e vértices em evidência.	79

Figura 7.2: Imagens do objeto e das arestas da caixa virtual.	80
Figura 7.3: Ajuste do AABB sobre o objeto caixa metálica. (a) imagem da esquerda. (b) imagem da direita.	84
Figura 7.4: Par de imagens do furador de papel.	85
Figura 7.5: Ajuste do AABB sobre o objeto perfurador de papel. (a) imagem da esquerda. (b) imagem da direita.	85
Figura 7.6: Par de imagens desalinhadas em relação ao gabarito.	86
Figura 7.7: Objeto a ser enquadrado sem alinhamento com o gabarito.	86
Figura 7.8: Objeto a ser enquadrado. (a) imagem da esquerda. (b) imagem da direita.	87
Figura 7.9: Objeto com orientação diferente das direções principais do gabarito de calibração. (a) imagem da esquerda. (b) imagem da direita.	88
Figura 7.10: Eixos principais para o par de imagens do objeto.	89
Figura 7.11: Produto vetorial entre dois vetores \mathbf{u} e \mathbf{v} .	90
Figura 7.12: Representação 3D dos eixos de orientação do objeto.	91
Figura 7.13: Definição dos vetores $\mathbf{v}_{k,i}$ em cada uma das direções.	91
Figura 7.14: (a) rotação sobre o eixo x. (b) rotação sobre o eixo y. (c) rotação sobre o eixo z.	92
Figura 7.15: Rotação da caixa de calibração conforme orientação do objeto	93
Figura 7.16: Par de imagens para o exemplo do ajuste.	94
Figura 7.17: Direções principais do objeto para as duas vistas da imagem.	95
Figura 7.18: Obtenção dos pontos de interseção.	96
Figura 7.19: Posição dos centros de gravidade para as duas imagens.	97
Figura 7.20: Centro de Gravidade médio (G').	98
Figura 7.21: Projeção do G' sobre os dois planos de imagem.	101
Figura 7.22: Correção dos eixos principais a partir do novo $cg'i$.	102
Figura 7.23: Projeção dos eixos principais xyz nos planos de imagem (a) e (b).	104
Figura 7.24: Construção espacial da caixa de contorno.	105
Figura 7.25: Ajuste do OBB para as imagens de exemplo.	106
Figura 7.26: Exemplo de obtenção do eixo Y^* a partir de Y .	107
Figura 7.27: Caixa de contorno virtual para referência inicial do ajuste.	107
Figura 7.28: Resultado final do ajuste da caixa de contorno.	108
Figura 7.29: Identificação dos vértices para a caixa de contorno 3D.	109

Figura 8.1: Imagens da peça a ser posicionada.	112
Figura 8.2: Vista parcial da estação de soldagem.	113
Figura 8.3: (a) esteira de transporte. (b) bancada. (c) peça sendo pega pelo robô. (d) peça sendo transportada.	114
Figura 8.4: (a) Peça na posição correta. (b) peça com posição incorreta.	115
Figura 8.5: Configuração do par de imagens para a calibração do sistema.	117
Figura 8.6: Detecção de cantos pelo método de Zhang (2000) para uma das imagens.	117
Figura 8.7: (a) imagem da bancada com a peça. (b) imagem da bancada sem a peça	119
Figura 8.8: (a) Imagem resultante da subtração. (b) imagem com segmentada.	119
Figura 8.9: Imagens resultantes após o ajuste do <i>bouding box</i> AABB.	120
Figura 8.10: Diferença de volume ocupado em função de um posicionamento indevido.	121

Lista de Tabelas

Tabela 6.1: Exemplo da calibração gerado pelo Toolbox em Matlab, de Zhang (2000).	64
Tabela 6.2: Coordenadas (x,y,z) para 10 pontos de projeção da caixa de calibração.	65
Tabela 7.1: Coordenadas dos vértices do gabarito para a imagem 1.	80
Tabela 7.2: Coordenadas dos vértices do gabarito para a imagem 2.	80
Tabela 7.3: Tabela comparativa de valores de volume medido e calculado.	111
Tabela 7.4: Tabela comparativa de erros de volume.	112
Tabela 8.1: Resultado da calibração das câmeras pelo método de Zhang.	119
Tabela 8.2: Resultados dos parâmetros de pose.	119

Nomenclatura

Letras Latinas

f – Função imagem.

x – Posição horizontal do *pixel* na imagem.

y – Posição vertical do *pixel* na imagem.

l – Valor da intensidade do *pixel*.

d – Módulo da distância.

t – Tempo.

g – Transformação Euclidiana espacial.

f – Distância focal.

P – Ponto no espaço.

X – Vetor de coordenadas do ponto no espaço.

p – Projeção do ponto P .

X^w, Y^w, Z^w – coordenadas do ponto P no espaço.

O^w – coordenada do centro do *frame* de referência.

O^c – coordenada do centro da câmera na espaço.

M – Número de linhas da imagem.

N – Número de colunas da imagem.

\mathbb{Z}^2 – Conjunto dos valores reais inteiros.

\mathbb{R}^3 – Conjunto dos valores reais.

R – Matriz de rotação.

T – Vetor de translação.

K – Matriz de parâmetros intrínsecos.

Letras Gregas

Π - Plano de imagem.

σ - valores singulares.

Abreviações

AABB – *Axis Aligned Bounding Box.*

OBB – *Oriented Bounding Box.*

Pixel – *Picture element.*

RGB – *Red, Green and Blue.*

CG – Centro de Gravidade.

CCD – *Charge Coupled Device.*

3D – espaço tridimensional.

2D – plano bidimensional.

SVD – Singular Value Decomposition.

ROI – Region of Interest.

Capítulo 1

Introdução

Muitas das tarefas executadas pelo homem são guiadas principalmente pela sua capacidade de enxergar. Através da visão é possível identificar, classificar e compreender as coisas que o cercam. A interpretação visual do mundo é algo natural ao ser humano, mas é de difícil representação computacional. As tecnologias atualmente disponíveis para manipulação de imagens têm colaborado significativamente na ampliação da capacidade da visão humana, bem como vem produzindo ferramentas que servem de apoio à tomadas de decisão, ou mesmo, na substituição da intervenção humana em processos repetitivos.

Quando se tenta dar a uma máquina o sentido da visão, se está preocupado em como uma cena deve ser interpretada por ela. Assim, a visão computacional trata de um conjunto de eventos que operam em seqüência, e vão desde a captura da imagem, passando pela sua decodificação através de um processamento da imagem de onde são extraídas as informações importantes para a interpretação de algum evento, até a execução de alguma ação de interação com um processo.

As ferramentas computacionais que operam com imagens digitais são usadas de acordo com duas necessidades fundamentais. Primeiro, na melhoria da imagem para interpretação humana, no que diz respeito à correção de iluminação e distorções, suavização de regiões, detecção de contornos, entre outros. Segundo, na criação de modelos para interpretação robótica tratando do processo de aquisição e formação da imagem para guiar um processo automatizado. Muitas tecnologias relacionadas aos sistemas de visão computacional e processamento de imagens têm sido utilizadas com sucesso em aplicações relevantes na manufatura assistida, robótica de manipuladores, robótica móvel e em sistemas produtivos. Estas aplicações utilizam

diversas ferramentas computacionais para o pré-processamento, segmentação de imagens e identificação de objetos. Dentre as diversas aplicações da visão computacional, a automação industrial aparece como um campo de pesquisa de grande interesse.

Apesar do crescente desenvolvimento industrial, pela diversificação dos recursos computacionais aplicados à automação da produção, muitas tarefas complexas ou meramente repetitivas, são realizadas ou monitoradas por operadores humanos. É fato identificado na pesquisa de Orth (1998), que o rendimento do trabalho de um operador humano decresce durante a sua jornada de trabalho, conseqüentemente afetando o resultado final da produção. Com base nesta justificativa, vem se buscando tornar os sistemas computacionais e de robótica, capazes de auxiliar operadores humanos em tarefas que podem ser automatizadas. Neste contexto, os sistemas de visão computacional, têm contribuído significativamente em diversas fases do ciclo produtivo, como na montagem de componentes, correção de trajetórias, reconstrução tridimensional, posicionamento e medição de objetos, visando o controle de qualidade. A inspeção visual computadorizada permite o exame automático e exaustivo de produtos, o que não acontece com a inspeção humana, que usualmente se baseia em aproximações estatísticas, como apresenta Salvi (2002).

Como a indústria em geral tem interesse em ter seus processos automatizados, muitas ferramentas matemáticas e computacionais estão sendo propostas. O processamento e a análise de imagens criam uma ciência que permite manipular, analisar e modificar imagens digitais, a partir de procedimentos que envolvem, morfologia matemática como em Facon (1996) e Hasan (2000); técnicas para o processamento de informação geométrica e estatística em Shen (2000); análise de componentes principais como em Zhang (2001); algoritmos de processamento de imagens implementados em hardware, descritos em Spitz (2000) e Gasteratos (2000); transformada rápida de Fourier em Oppenheim (1989); extração de características (*features*) em Gordon (1984); *wavelets* em Quak (1994), e paradigmas da inteligência computacional, tais como redes neurais descritas em Augusteijn (1996), algoritmos genéticos em Wang (2001), e sistemas nebulosos em Cheng (2000), entre outros. Assim, os sistemas de visão artificial contemplam um conjunto de métodos que dão suporte ao desenvolvimento de produtos confiáveis ou processos suficientemente eficientes, que possam ser empregados em situações práticas de um ambiente de

manufatura. Em um esforço para aumentar a eficiência e a autonomia dos processos de manufatura, como em Abrams (1996), muitos pesquisadores têm procurado desenvolver estratégias automáticas para inspeção de produtos manufaturados usando uma variedade de sensores. Frequentemente os sensores de visão como as câmeras CCD (*Charge Coupled Device*) são usadas para se obter posição e orientação dos objetos em uma cena.

Os sistemas de visão podem operar com uma ou mais câmeras, e que podem ser móveis ou fixas, como descreve Motta (2002). Embora seja possível se ter uma idéia do mundo 3D usando uma única imagem, por um processo direcionado de segmentação como em Ryoo (2004), procedimentos mais consistentes podem ser obtidos a partir de duas imagens, que podem ser obtidas de uma mesma câmera que muda de posição, como descrevem Jain (1995), Trucco (1998), Forsyth (2003) e Ma (2004).

Um dos objetivos da visão computacional é criar um modelo do mundo real a partir de imagens. Devido às particularidades e ao conjunto de restrições de cada tipo de aplicação, as tecnologias usadas em visão robótica, são direcionadas a solução de um problema específico, adaptando o conjunto de ferramentas matemáticas e os sistemas computacionais para a resolução de cada tipo de problema. Não é possível ter um sistema de visão genérico que possa ser usado em vários propósitos, mas a aplicação de várias técnicas de modo escalonado permite dar solução aos problemas mais complexos. Um desses problemas é relativo à recuperação da informação tridimensional de objetos a partir de imagens. A determinação de dados em três dimensões (3D) a partir de imagens em duas dimensões (2D) é uma área muito importante em visão computacional e vem sendo bastante estudada. Um sistema de visão computacional recupera informação sobre uma cena a partir de suas projeções bidimensionais. Como as imagens são projeções bidimensionais do mundo real tridimensional, qualquer informação sobre a cena não é obtida diretamente, mas, pode ser estimada por triangulação como apresentado em Jain (1995). Existem alguns problemas bem definidos relacionado com a obtenção da informação 3D de uma cena.

O primeiro problema é chamado localização de correspondência, que é a forma de estabelecer uma correlação entre duas imagens do mesmo objeto obtidas a partir de pontos de vista diferentes. Um ponto projetado na primeira imagem deve corresponder a um ponto na

segunda imagem, que é a projeção do mesmo ponto da cena real. Inúmeros métodos para determinação automática desta correspondência têm sido propostos. Faugeras (1995) descreve que para o caso de duas câmeras, a teoria está bastante completa e apresenta os procedimentos algébricos e geométricos para o caso de imagens obtidas a partir de mais de duas câmeras. Erros de correspondência podem ocorrer quando os eixos óticos da câmera não são paralelos e a correspondência entre um modelo e o conjunto de pontos tridimensional do objeto podem restringir o número de correspondências falsas como em Calafiore (1997) e Galo (1999); técnicas mais recentes são baseadas na forma do objeto como em Tavares (2000) e Zhou (2002), e na estimação da correspondência por bordas, cantos e regiões como em Xie (2004), ou usam técnicas baseadas em gradientes como em Twardowski (2004).

O segundo problema relacionado com a reconstrução tridimensional é chamado de calibração de câmeras. Como apresenta Salvi (2002), a calibração é necessária em qualquer inspeção visual onde se busca determinar a posição de objetos, ou a localização da câmera no ambiente. Normalmente, a única informação que se dispõe são as imagens dos objetos. A idéia básica da calibração é estabelecer uma relação entre os pontos da imagem e as coordenadas espaciais dadas em unidade métricas. Esta relação depende da configuração dos parâmetros óticos da câmera, bem como da estimação da posição relativa entre câmeras. A posição relativa é dada por uma matriz de rotação e um vetor de translação, que formam uma expressão matricial chamada de matriz essencial como apresentam Trucco (1998), Forsyth (2003) e Ma (2004) entre outros. A estimação da matriz essencial é um passo preliminar na obtenção do modelo tridimensional. A maneira de estimar a matriz essencial é pela sua decomposição em valores singulares (*SVD – singular value decomposition*), o que permite encontrar os parâmetros de rotação e translação como em Tsai (1984), Fiori (2000), Oisel (2003) e Ma (2004). Procedimentos para estimação da matriz essencial estão presentes nas aplicações de reconstrução 3D, onde a precisão de medidas são necessárias. Diferentes formas de estimação desta matriz são apresentadas por Salvi (2002) no contexto de calibração de câmeras.

A determinação do aspecto de objetos em termos de seu volume e posição, são descritos em vários estudos presentes na literatura da área, que tratam da extração das informações 3D a partir de imagens. Quando se faz necessário conhecer detalhes das faces do objeto, é comum a

utilização de nuvens de pontos para reconstruir determinada superfície. Como o esforço computacional neste caso é alto, vários trabalhos tem proposto a otimização da reconstrução com o objetivo de minimizar o espaço de parâmetros. Entre as proposições está o uso de marcadores de coordenadas apresentado por Wang (2002), a triangulação de Delaunay para a criação de facetas triangulares, como em Yuan (1993), e os mapas de disparidade como usado por Oisel (2003). Quando a aparência da superfície não é relevante para o processo, como na criação de mapas de navegação de robôs, correção de movimento de manipuladores, posicionamento de objetos em sistemas de transporte, a informação a ser reconstruída pode ser bastante simplificada.

Apesar de elementos cúbicos serem usados para reconstrução densa baseados na criação de superfícies poligonais, chamadas *isosurfaces* como em Bourke (1997) e Lorensen (1987), ou pela aproximação de um objeto de forma conhecida chamado *marching cube*, descrito em Nielson (2003), a proposta da tese se baseia na construção de uma caixa de contorno (*bounding box*) usada para definir os limites ocupados por objetos, como apresentado em Gottschalk (2000). Assim, pela criação de uma caixa de contorno, é possível mapear o ambiente definido pelo espaço ocupado por um objeto.

1.1. Proposta

Está sendo proposto um método para representar o espaço ocupado por um sólido usando caixas de contorno. De maneira diferente de Barequet (2001), Chu (2001), Shin (2003), Chan (2003), Majchrzak (2004) e Chan (2005), este trabalho propõe a utilização das projeções dos vértices e arestas de duas imagens de um cubo de calibração, para a criação de uma caixa virtual sobre a imagem do objeto. A caixa virtual serve de ponto de partida para a construção de uma caixa de contorno ajustada às bordas de um objeto.

O trabalho também propõe um método para o ajuste da caixa de contorno. A caixa de contorno pode ser ajustada sobre o espaço ocupado pelo objeto, pela relação entre os pontos de fuga da caixa virtual e os pontos da borda da região de interesse da imagem. No final do processo as faces da caixa de contorno estarão tangenciando as bordas do objeto. Os vértices podem ser

reconstruídos tridimensionalmente e isto permite estimar o volume aproximado e a posição de um sólido.

A técnica proposta tem aplicação na orientação da posição de manipuladores, identificação de obstáculos para robótica móvel, classificação de objetos quanto à forma e dimensões, e outras onde a posição do objeto pode interferir na qualidade final do processo.

1.2. Organização do texto

O texto está organizado da seguinte forma. No capítulo 2 são apresentados os conceitos básicos sobre a imagem digital, e sobre a modelagem geométrica da câmera e sua transformação perspectiva. O capítulo 3 trata da reconstrução 3D, apresentando a teoria sobre a geometria epipolar para múltiplas vistas, e também descrevendo a teoria sobre o procedimento de calibração dos parâmetros da câmera. O capítulo 4 apresenta as técnicas de segmentação utilizadas no processo de definição do método. O capítulo 5 define o conceito de uma caixa de contorno e identifica os requisitos para a sua construção. O capítulo 6 apresenta a proposta do método para a criação da caixa de contorno. O capítulo 7 apresenta a aplicação prática do método para imagens reais apresentando os resultados obtidos. O capítulo 8 apresenta um estudo de caso com a aplicação do método proposto, e no capítulo 9 são apresentadas as principais conclusões e considerações finais do trabalho. Em anexo constam as informações necessárias sobre processamento de imagens.

Capítulo 2

Representação Digital da Imagem

Este capítulo descreve a formação da imagem digital sobre o aspecto de sua representação bidimensional através de uma função, como também sobre o aspecto da geometria associada na transformação da realidade tridimensional para sua representação bidimensional.

2.1. Formação da Imagem Digital

Como descreve Gonzalez (2002) uma imagem é uma função $f(x,y)$ formada por uma matriz de M linhas e N colunas, onde cada posição (x,y) contém um valor discreto. Este valor representa a menor informação da imagem, chamada de *pixel*, uma abreviação da expressão *picture element*.

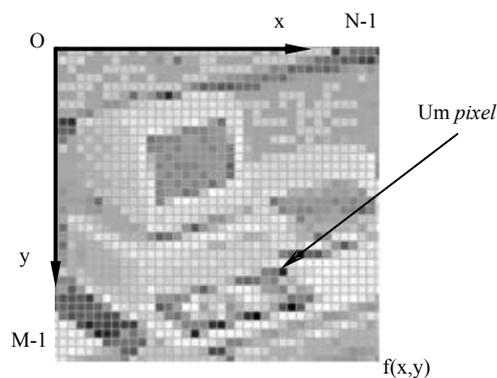


Figura 2.1: Exemplo de uma imagem digital como uma função $f(x,y)$.

A figura 2.1 apresenta o exemplo de uma imagem, de tamanho $M \times N$. A coordenada $(0,0)$ representa a origem da imagem no canto superior esquerdo, e as posições dos *pixels* aumentam da esquerda para direita, na direção do eixo x , e de cima para baixo na direção do eixo y .

Então, a imagem digital pode ser representada da maneira apresentada pela expressão (2.1),

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & \ddots & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix} \quad (2.1)$$

Por definição, a matriz da equação de $f(x,y)$ representa uma imagem digital, onde cada elemento é um *pixel*. O *pixel* contém o valor que representa um determinado nível de intensidade da iluminação percebida pelo sensor de uma câmera. Pode-se denotar a imagem de maneira mais convencional, como em Gonzalez (2002) fazendo $a_{ij} = f(x = i, y = j) = f(i,j)$ na forma,

$$A = \begin{bmatrix} a_{0,0} & a_{0,1} & \dots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \dots & a_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M-1,0} & a_{M-1,1} & \dots & a_{M-1,N-1} \end{bmatrix} \quad (2.2)$$

O processo de captura da imagem pode ser entendido como a amostragem de pares ordenados de elementos (x,y) do produto cartesiano Z^2 , onde Z é o conjunto dos valores reais inteiros. Então, $f(x,y)$ é uma imagem se (x,y) são valores inteiros em Z^2 e f é uma função que associa um valor de intensidade para cada par (x,y) . Define-se intensidade pelo símbolo I como o nível de cinza para cada ponto em uma imagem monocromática. Imagens monocromáticas e coloridas possuem características distintas, que são descritas no item 2.2.

2.2. Características das Imagens

As características das imagens se referem à forma com que elas armazenam as informações de uma cena através de uma representação numérica dos *pixels*. As imagens podem ser

classificadas como binárias (*black & white*), em níveis de cinza (*gray scale*), coloridas indexadas (*indexed*) ou RGB (*truecolor*), cujas definições são dadas por Gonzalez (2002) e apresentadas a seguir. Detalhes adicionais são fornecidos no Anexo I:

- i. Imagem Binária: É uma imagem que contém apenas dois valores discretos, que representam os *pixels* em preto ou branco. Na imagem normalizada (capítulo 4) o valor $l=0$ se refere ao preto e o valor $l=1$ ao branco. Imagens deste tipo são muito úteis para a realização da análise morfológica de um objeto (anexo II), onde a presença de cores ou texturas não é relevante. Este tipo de imagem é usado neste trabalho para a identificação de regiões de interesse, o que é apresentado em detalhes no capítulo 6.
- ii. Imagem Indexada: É uma imagem cujos valores l dos *pixels* são os índices de um mapa de cores contendo como cores primárias, o vermelho, o verde e o azul (*RGB - Red, Green, Blue*). Este mapa é formado por uma matriz de dimensão $m \times 3$. Cada linha m do mapa especifica as quantidades dos componentes em vermelho, verde e azul da cor. O mapa de cores é armazenado junto com a imagem e automaticamente carregado quando ela é manipulada.
- iii. Imagem em Níveis de Cinza: É uma imagem que contém valores de intensidade l em escala de cinza para os pixels. Este tipo de imagem não utiliza um mapa de cores e os *pixels* contêm diretamente a intensidade, onde o valor $l=0$ representa o preto e valor $l=1$ representa o branco para uma imagem normalizada (capítulo 4); e os valores intermediários com $0 < l < 1$ representam diversos os níveis de cinza.
- iv: Imagem RGB: É um tipo de imagem colorida, onde cada *pixel* possui três valores pertencentes a três planos de cores primárias, que identificam os componentes em vermelho, verde e azul (*Red, Green e Blue*). A imagem RGB não usa paleta de cores, e cada cor é uma matriz que representa um dos três planos de cores, gerando uma imagem $M \times N \times 3$. A cor é formada pela combinação dos valores de cada plano da imagem. Uma imagem RGB normalmente usa 24 bits, totalizando cerca de 16 milhões de cores. Este tipo de imagem também é chamado de *truecolor*, pois se aproxima do mundo real em termos de cores.

Em função das características apresentadas, cada tipo de imagem possui uma aplicação específica. Imagens médicas normalmente são coloridas e de alta definição, pois se busca evidenciar detalhes para obtenção precisa de diagnósticos. Para aplicações em automação e robótica, normalmente imagens em níveis de cinza ou binárias são suficientes. Como o tamanho em bytes das imagens binárias é muito pequeno em relação a imagens *truecolor*, elas são mais usadas, quando possível, para processamento em tempo real, devido ao menor requisito computacional para operá-las.

Se a aquisição é feita por câmeras coloridas, é necessário realizar a transformação das imagens para um formato mais adequado. Estas transformações são apresentadas no capítulo 4, referente às técnicas de processamento de imagens.

2.3. Modelo Geométrico da Câmera

2.3.1. Representação do Espaço Tridimensional

O espaço tridimensional pode ser representado analiticamente por um sistema cartesiano de coordenadas, onde cada ponto $P \in \mathbb{R}^3$ possui três coordenadas individuais $\mathbf{X}=[X_1, X_2, X_3]^T$, ou na forma $[X, Y, Z]^T$. A figura 2.2 apresenta a identificação do sistema de eixos cartesianos conforme a regra da mão direita. É conveniente usar a palavra *frame*, ao invés de "sistemas de eixos" para descrever a orientação de um objeto e a posição de uma referência, pois é a nomenclatura mais apropriada na literatura da área, como em Ma (2004).

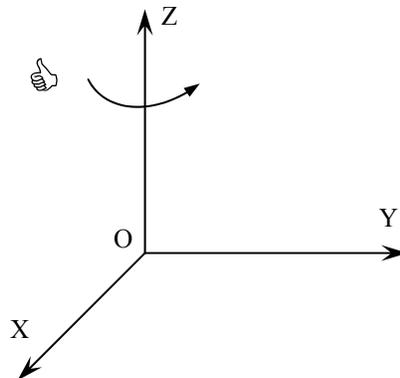


Figura 2.2: Identificação dos eixos do sistema cartesiano conforme a regra da mão direita.

2.3.2. Representação do Corpo Rígido

Considerando um objeto em movimento sendo capturado por uma câmera, a trajetória descrita para cada partícula dele, representa a trajetória do objeto como um todo, ou seja, quando a distância entre pontos deste objeto não varia com o movimento, chama-se este objeto de corpo rígido. Assim, o movimento do corpo rígido, é um mapeamento que descreve como as coordenadas de cada ponto sobre um objeto rígido variam, satisfazendo a condição:

$$\|\mathbf{X}(t) - \mathbf{Y}(t)\| \equiv d \quad (2.3)$$

Onde $\mathbf{X}(t)$ e $\mathbf{Y}(t)$ são coordenadas de dois pontos quaisquer sobre o objeto, especificados como uma função do tempo, com distância d constante e $t \in \mathbb{R}$. O mapeamento entre a configuração inicial e sua configuração final, pode ser denotada como:

$$g: \mathbb{R}^3 \rightarrow \mathbb{R}^3 ; \mathbf{X} \rightarrow g(\mathbf{X}) \quad (2.4)$$

De acordo com MA (2004), o mapeamento g que preserva a distância entre pontos é chamado movimento de corpo rígido ou transformação Euclidiana espacial, pois deve preservar a norma e o produto cruzado entre os dois vetores. Um objeto pode ser associado a um sistema de eixos chamado de *frame* do objeto, e seu movimento de corpo rígido pode ser totalmente especificado pelo movimento deste *frame*.

A figura 2.3 apresenta um ponto P no espaço de coordenadas (X^w, Y^w, Z^w) e centro O^w pertencente a um objeto de centro O^c . Quando o objeto se move, o seu *frame* também move com o objeto.

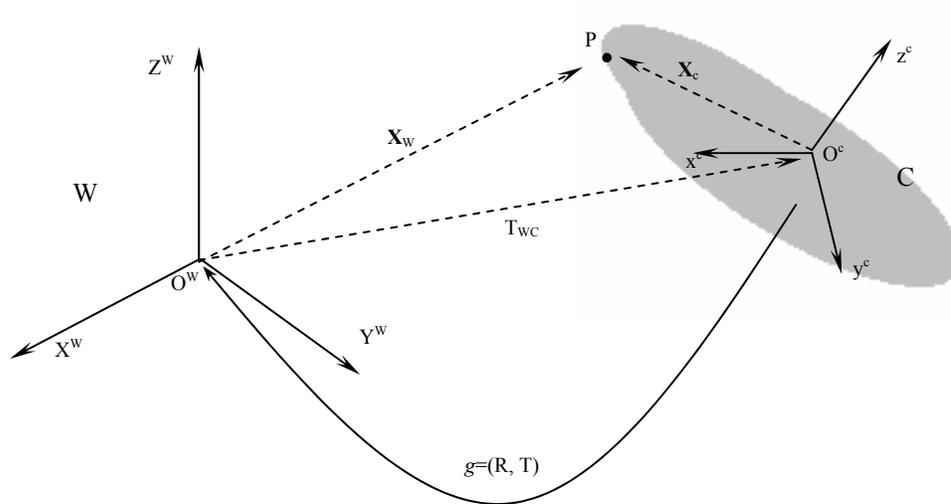


Figura 2.3: Relação entre o sistema de coordenadas do mundo W e do objeto C.

A posição do objeto é determinada por dois componentes: primeiro por um vetor definido entre a origem da referência O^w e a origem do objeto O^c , denominado componente de translação, chamado de T . E segundo, a orientação relativa do sistema de coordenadas do objeto C, denominado componente de rotação R , em relação a um *frame* de referência fixo W.

Como ilustrado pela figura 2.3, pode ser observado que o vetor X_w é a soma da translação T_{wc} da origem do *frame* C relativo ao *frame* W com o vetor X_c , expresso pelo sistema de coordenadas W. Como R_{wc} é a rotação relativa entre os dois sistemas de eixos, então X_w pode ser dado por:

$$\mathbf{X}_w = R_{wc} \mathbf{X}_c + T_{wc} \quad (2.5)$$

Onde a matriz de rotação R é dada por:

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (2.6)$$

e,

$$\begin{aligned}
r_{11} &= \cos \phi \cos k & r_{12} &= \text{sen } \omega \text{sen } \phi \cos k + \cos \omega \text{sen } k & r_{13} &= -\cos \omega \text{sen } \phi \cos k + \text{sen } \omega \text{sen } k \\
r_{21} &= -\cos \phi \text{sen } k & r_{22} &= -\text{sen } \omega \text{sen } \phi \text{sen } k + \cos \omega \cos k & r_{23} &= \cos \omega \text{sen } \phi \text{sen } k + \text{sen } \omega \cos k \\
r_{31} &= \text{sen } \phi & r_{32} &= -\text{sen } \omega \cos \phi & r_{33} &= \cos \omega \cos \phi
\end{aligned}$$

Com ω representando a rotação sobre o eixo x, θ é a rotação sobre o eixo y, e k a rotação sobre o eixo z.

2.3.3. Representação em Coordenadas Homogêneas

A transformação de coordenadas para o movimento de um corpo rígido não é linear, mas afim (*affine*) Ma (2004). Dois vetores u, v são relacionados por uma transformação linear se $u = Av$ para uma matriz A , e por uma transformação afim se $u = Av + b$, para uma matriz A e um vetor b , como descrito em Ma (2004). Pode-se converter cada transformação afim para um modelo linear usando coordenadas homogêneas. Esta representação é dada pela inserção de um fator de escala, sendo usualmente o valor 1, às coordenadas de um ponto $\mathbf{X} = [X_1, X_2, X_3]^T \in \mathbb{R}^3$. Assim,

$$\bar{\mathbf{X}} = \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ 1 \end{bmatrix} \in \mathbb{R}^4 \quad (2.7)$$

Usando esta notação a equação 2.7 pode ser reescrita da seguinte forma:

$$\bar{\mathbf{X}}_w = \begin{bmatrix} \mathbf{X}_w \\ 1 \end{bmatrix} = \begin{bmatrix} R_{wc} & T_{wc} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{X}_c \\ 1 \end{bmatrix} \quad (2.8)$$

Então, a representação homogênea do movimento do corpo rígido é dada por,

$$\bar{\mathbf{g}} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{4 \times 4} \quad (2.9)$$

2.4. Modelos de Projeção

As câmeras podem ser representadas por modelos que descrevem a forma como os objetos de uma cena são visualizados. Os modelos definem a transformação de um dado ponto no espaço, que estão em coordenadas tridimensionais $P(X,Y,Z)$ para uma representação bidimensional $p(x,y)$ na imagem formada pela câmera. Segundo Banerge (2001) e Forsyth (2003) as câmeras podem ser modeladas através do modelo em perspectiva (*perspective projection*), modelo afim (*affine projection*), perspectiva fraca (*weak perspective*) e modelo ortográfico (*orthographic projection*).

A proposta da tese trata da utilização de pontos de fuga de uma caixa de contorno para realizar o enquadramento de um objeto. Para garantir a obtenção dos pontos de fuga da imagem o ambiente de câmeras deve ser modelado pela projeção em perspectiva.

2.4.1. Projeção em perspectiva

Um caso especial do modelo projetivo, é chamada de projeção perspectiva, baseada numa representação denominada *pinhole*, onde todos os raios de projeção do objeto convertem para um centro ótico, apresentado na figuras 2.4 e na figura 2.5, como em Forsyth (2003) e Jain (1995).

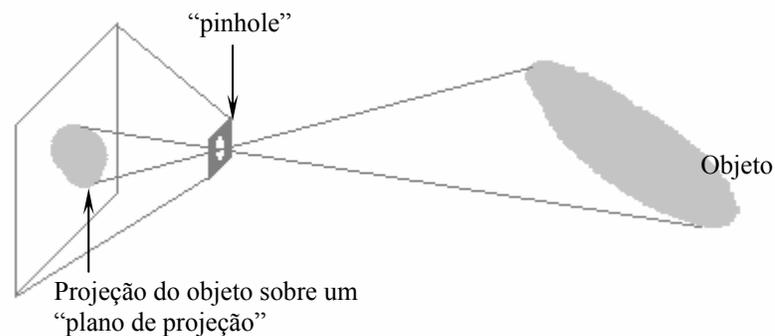


Figura 2.4: Representação dos raios óticos através do orifício (*pinhole*).

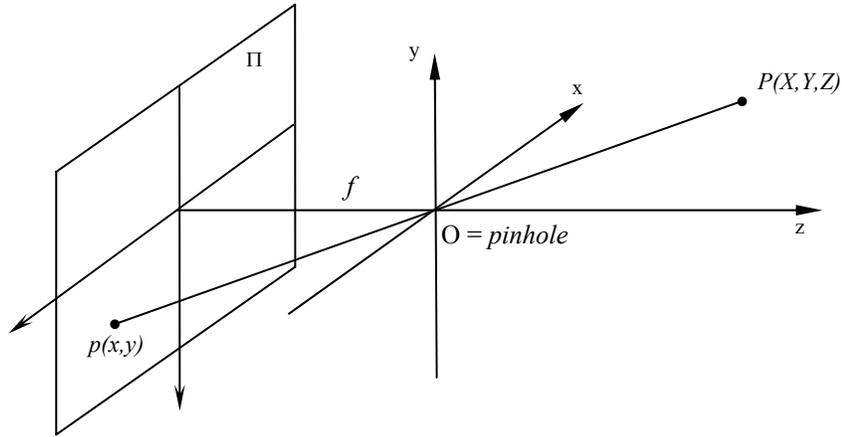


Figura 2.5: Modelo da projeção perspectiva adaptado de Forsyth (2003).

A imagem do ponto P que passa pelo centro ótico da câmara é projetada no plano invertido Π originando p .

Para este modelo, a matriz de transformação $T_{4 \times 4}$, é dada por:

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{f} & 0 \end{bmatrix} \quad (2.10)$$

onde, $T_{4 \times 4}$ é formada por uma matriz de rotação, que é a matriz interna de dimensão 3×3 de (2.10) mais a esquerda. Assim, tem-se:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z} \begin{bmatrix} X \\ Y \end{bmatrix}$$

$$(x, y) = \left(X \frac{f}{Z}, Y \frac{f}{Z} \right) \quad (2.11)$$

Cada ponto é escalonado por uma profundidade individual, e todos os raios de projeção convergem para o centro ótico.

2.5. Modelagem Geométrica da Câmera

Dentre os vários modelos de projeção, o modelo em perspectiva é um dos mais amplamente usados na representação de imagens como apresenta Ma (2004). Também neste trabalho, a modelagem da câmera é baseada no modelo em perspectiva. Isto se deve a necessidade de encontrar os vários pontos de fuga da imagem (capítulo 6) com a interseção das linhas no infinito.

No modelo em perspectiva, a formação de uma imagem, pode ser definida através da representação da câmera real por um modelo chamado *pinhole*, como apresentado na figura 2.5. A projeção perspectiva considera a representação virtual do objeto posicionada atrás do *pinhole*, sobre um plano chamado de “plano de imagem”. Por conveniência, o plano de imagem pode ser representado à frente do centro ótico da câmera, como ilustra a figura 2.6. Neste caso não é necessário operar com a imagem invertida, o que facilita a interpretação da cena. Qualquer objeto tem sua perspectiva formada sobre o plano de imagem Π da câmera.

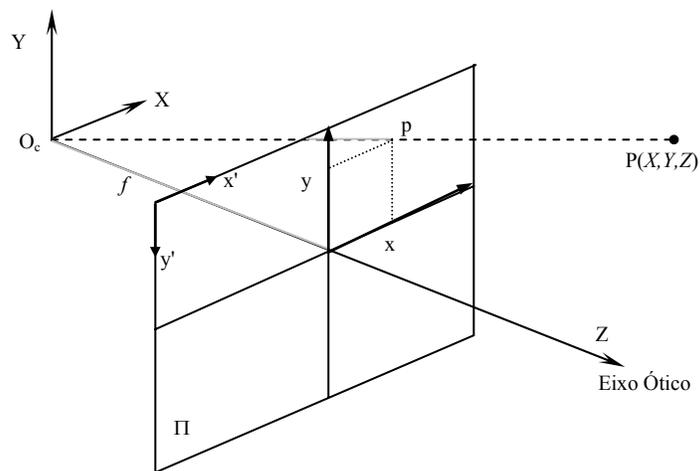


Figura 2.6: Projeção de um ponto P (3D) no plano de imagem Π (2D).

A figura 2.6, apresenta o modelo em perspectiva da câmera, que é composto de um plano de imagem Π e um ponto O_c em 3D, chamado centro de projeção (ou foco de projeção). A distância f entre o plano da imagem Π e o centro O_c é chamada distância focal. A linha através de O_c , perpendicular ao plano de imagem é chamada de eixo ótico. O ponto p é o ponto em que a

linha reta \mathbf{P} , entre P e O_c , intercepta o plano de imagem. Ou seja, $\mathbf{p}=[x,y,f]^T$ é a projeção do ponto $\mathbf{P}=[X,Y,Z]^T$ sobre o plano da imagem.

Jain (1995) descreve que a posição da projeção do ponto \mathbf{P} no plano de imagem pode ser obtida por semelhança de triângulos, como apresentado na figura 2.7.

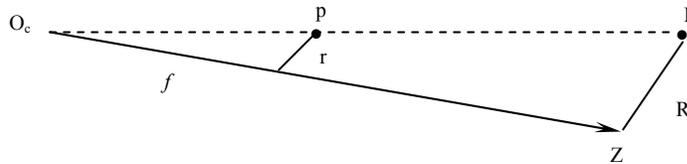


Figura 2.7: Relação geométrica para obtenção de p .

Da figura 2.7, obtém-se a relação:

$$\frac{f}{Z} = \frac{r}{R} \quad \text{e} \quad \frac{x}{X} = \frac{y}{Y} = \frac{r}{R} \quad (2.12)$$

Combinando as duas expressões de (2.12), tem-se:

$$\frac{x}{X} = \frac{f}{Z} \quad \text{e} \quad \frac{y}{Y} = \frac{f}{Z} \quad (2.13)$$

Então a posição do ponto p é dado por:

$$x = \frac{f}{Z} X \quad \text{e} \quad y = \frac{f}{Z} Y \quad (2.14)$$

A equação (2.14) representa a transformação de um ponto no espaço, em coordenadas 3D, para um ponto no plano, em coordenadas 2D. Nesta representação, as posições de x e y são relativas ao referencial da câmera, em coordenadas métricas. Entretanto, qualquer análise sobre a imagem depende da posição do ponto (em *pixels*) dado por p' (x',y'). Como mostra a figura 2.6, o

canto superior esquerdo do plano da imagem é o ponto de coordenada p' (0,0) das posições do *pixel*. Os valores de x' e y' , dependem apenas da imagem, já os valores de x e y estão associados com a posição da câmera.

2.5.1. Parâmetros da Câmera

As informações sobre as características físicas da câmera e sobre a sua posição, são divididas em duas categorias, denominadas respectivamente orientação interior, que são relacionados aos parâmetros intrínsecos, e orientação exterior, relacionados aos parâmetros extrínsecos. Estes parâmetros são definidos a seguir.

2.5.1.1. Parâmetros Intrínsecos

Os parâmetros intrínsecos, também conhecidos como valores da orientação interior são os parâmetros necessários para obter as características digitais, óticas e geométricas de uma câmera. Estes valores dizem respeito à distância focal f , a transformação entre o sistema de coordenadas da câmera \mathbf{p} e a coordenada em *pixels* \mathbf{p}' , como também a distorção geométrica das lentes.

Para a projeção perspectiva de uma câmera ideal, considerando um ponto genérico P , de coordenadas $\mathbf{P}_0=[X_0, Y_0, Z_0]^T \in \mathbb{R}^3$ relativo ao *frame* de referência¹, as coordenadas do mesmo ponto P , relativo ao *frame* da câmera são dadas pela transformação de corpo rígido, como apresentada no item 2.3.2, onde,

$$\mathbf{P} = R\mathbf{P}_0 + T \quad (2.15)$$

Através do modelo *pinhole*, o ponto \mathbf{P} projetado sobre o plano de imagem é representado por \mathbf{p} da forma,

$$\mathbf{p} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z} \begin{bmatrix} X \\ Y \end{bmatrix} \quad (2.16)$$

E, em coordenadas homogêneas, 2.16 tem a seguinte característica,

¹ Segundo Ma (2004) normalmente indica-se \mathbf{P}_0 com índice 0, o ponto relativo a posição inicial do *frame* da câmera.

$$Z\mathbf{p} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{P} \quad (2.17)$$

Onde, $\mathbf{P} = [X, Y, Z, 1]^T$ e $\mathbf{p} = [x, y, 1]^T$. Decompondo a expressão 2.17 em duas matrizes com a forma,

$$K_f = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{e} \quad \Pi_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (2.18)$$

A perspectiva de uma câmara ideal tem o seguinte formato:

$$\lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix} \quad (2.19)$$

Onde λ é um fator de escala representado por valor escalar positivo, pois a profundidade Z de P é desconhecida. Na forma matricial, 2.19 pode ser reescrita como,

$$\lambda \mathbf{p} = K_f \Pi_0 \mathbf{P} = K_f \Pi_0 \mathbf{g} \mathbf{P}_0 \quad (2.20)$$

Se $f=1$ a matriz de projeção perspectiva assume a forma,

$$\lambda \mathbf{p} = \Pi_0 \mathbf{P} = \Pi_0 \mathbf{g} \mathbf{P}_0 \quad (2.21)$$

Para tornar a equação 2.20 útil em termos práticos, é necessário especificar o relacionamento entre o *frame* de coordenadas do plano de imagem dado por \mathbf{p} , em relação à matriz de *pixels* de coordenadas (i,j) da imagem.

O plano de imagem Π da figura 2.6 pode ser representado por uma matriz de *pixels* como mostra Ma (2004) através da figura 2.8.

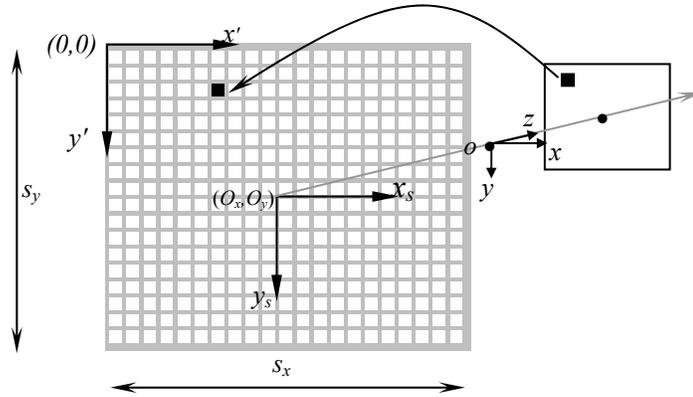


Figura 2.8: Transformação de coordenadas da imagem para coordenadas em *pixel*.

Como a coordenada (x,y) no plano de imagem é especificada em termos de unidades métricas, e (x_s, y_s) são coordenadas em *pixels*, a transformação depende do tamanho do *pixel* na direção x e y , dados por s_x e s_y , respectivamente. Para obter a transformação das coordenadas da imagem \mathbf{p}' com as coordenadas \mathbf{p} do mesmo ponto na câmera, Jain (1995) e Trucco (1998) sugerem e a seguinte relação:

$$x' = (x_s + o_x) \tag{2.22}$$

$$y' = (y_s + o_y)$$

Onde o_x e o_y são as coordenadas em *pixel* do centro do plano da imagem, e s_x e s_y representam o tamanho do *pixel* em milímetros, respectivamente na direção horizontal e vertical. Ma (2004) representa esta relação da forma matricial,

$$\begin{bmatrix} x_s \\ y_s \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{2.23}$$

A partir de (2.22) e (2.23), o seguinte sistema em coordenadas homogêneas define \mathbf{p}' como:

$$p' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2.24)$$

Agora, x' e y' se referem à coordenadas de imagem em *pixel*. Se o *pixel* não é retangular a matriz de escala é formada por um parâmetro adicional s_θ muito próximo de zero chamado *skew factor*. Então a matriz de transformação assume a forma mais genérica,

$$K_s = \begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \quad (2.25)$$

A matriz K de parâmetros intrínsecos é dada pela combinação da matriz K_f com a matriz K_s :

$$K = K_s K_f = \begin{bmatrix} s_x & s_\theta & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} fs_x & fs_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \quad (2.26)$$

Assim, pode-se relacionar diretamente um ponto em coordenadas espaciais, para coordenadas em *pixel* da imagem, como apresenta a equação (2.27):

$$\lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} fs_x & fs_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix} \quad (2.27)$$

A matriz K da equação (2.26) reúne os parâmetros intrínsecos de uma câmera. Esta matriz também é denominada, matriz de calibração da câmera.

2.5.1.2. Parâmetros Extrínsecos

Como descreve Trucco (1998), os parâmetros extrínsecos definem a localização e a orientação do *frame* de referência da câmera em relação ao *frame* de referência do mundo. O

frame de referência da câmera é freqüentemente desconhecido, e o problema é determinar a localização e orientação do sistema de eixos da câmera com relação a algum referencial conhecido, usando somente a informação da imagem.

A escolha típica para descrever a transformação entre coordenadas da câmera e coordenadas do mundo é usar, um vetor de translação 3D, denominado T , descrevendo as posições relativas das origens dos dois sistemas de eixos de referência, e uma matriz de rotação $R_{3 \times 3}$, que relaciona os eixos correspondentes dos dois sistemas de coordenadas. Isto é exemplificado através da figura 2.9.

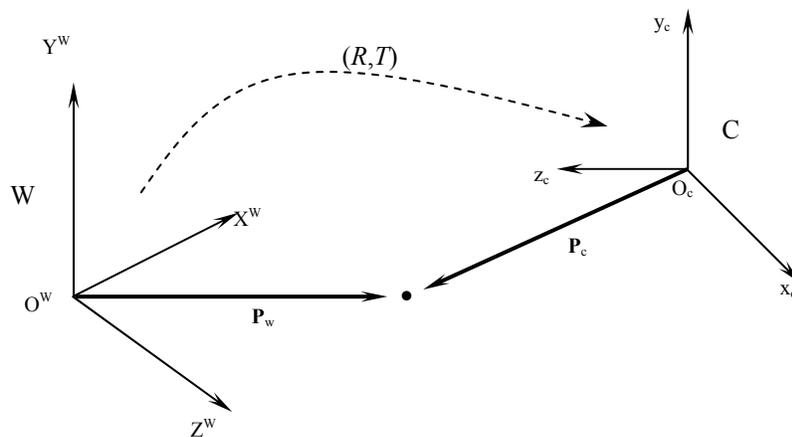


Figura 2.9: Relação entre coordenadas do mundo e da câmera adaptado de Trucco (1998).

A relação entre o *frame* W e o *frame* C é dada pela relação,

$$P_c = R(P_w - T) \quad (2.28)$$

Onde R é da forma da expressão 2.6. Os valores de R e T são os parâmetros extrínsecos da câmera.

2.6. Distorção Radial

Distorções da imagem originadas pelo uso das lentes nas câmeras CCD são visíveis onde os *pixels* mais externos, na periferia da imagem, estão a uma grande distância do centro. Este tipo de distorção é chamado de distorção radial e normalmente ocorre quando a câmera captura imagem de um objeto que está distante dela. A figura 2.10.a apresenta uma imagem com distorção aparente, onde as linhas retas aparecem curvadas na imagem. A figura 2.10.b apresenta a imagem corrigida.

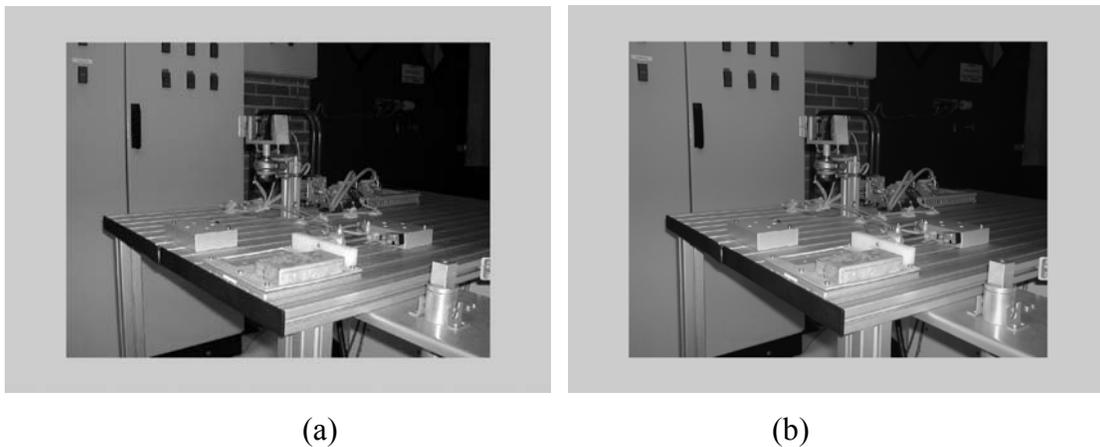


Figure 2.10: (a) imagem original com distorção radial. (b) imagem corrigida.

Para corrigir a distorção radial, Trucco (1998) e Ma (2004) apresentam as equações abaixo para imagens calibradas.

$$x = x_d(1 + k_1r^2 + k_2r^4) \quad (2.29)$$

$$y = y_d(1 + k_1r^2 + k_2r^4) \quad (2.30)$$

Onde (x_d, y_d) são as coordenadas dos pontos distorcidos, e

$$r^2 = x_d^2 + y_d^2 \quad (2.31)$$

Como descreve Forsyth (2003), os parâmetros k_1 e k_2 possuem valores muito pequenos e são assumidos empiricamente. A distorção é um deslocamento dos pontos da imagem na forma radial. O deslocamento é nulo no centro da imagem e aumenta com a distância a partir do centro. Para um sistema não calibrado, os parâmetros de distorção podem ser obtidos diretamente da imagem. Um método comumente utilizado é o definido em Deverney (1995) dado como:

$$\mathbf{x} = c + f(r)(x_d - c) \quad (2.32)$$

$$f(r) = 1 + a_1 r + a_2 r^2 + a_3 r^3 + a_4 r^4 \quad (2.33)$$

$$\mathbf{x}_d = [x_d, y_d]^T, \quad c = [c_x, c_y]^T \quad (2.34)$$

Onde \mathbf{x}_d são as coordenadas dos pontos distorcidos, e r é a distância do ponto ao centro de distorção c e $f(r)$ é o fator de correção da distorção.

Apesar de ser possível corrigir a distorção radial, o problema pode ser descartado para a aplicação proposta, pois as imagens do objeto estão próximas ao centro da imagem, onde a influência da distorção pode ser desprezada.

2.7. Resumo do Capítulo

Este capítulo proporcionou uma visão geral sobre o processo de formação da imagem, desde a transformação geométrica do objeto no espaço até a sua representação no plano de imagem. A escolha de um dos modelos de câmera apresentados, está relacionado com o propósito da aplicação. O modelo em perspectiva é usado neste trabalho devido a três justificativas. Primeiro, porque existe vasta literatura sobre a geometria envolvida, permitindo o total embasamento teórico necessário à fundamentação do objeto da tese. Segundo, porque permite a reconstrução de um objeto no espaço Euclidiano, e terceiro porque os parâmetros obtidos deste modelo permitem a obtenção dos pontos de fuga da imagem, que são essenciais para a confecção das ferramentas propostas no capítulo 6.

Capítulo 3

Reconstrução Tridimensional

A reconstrução tridimensional de um objeto trata da obtenção da medida de profundidade da cena capturada pela câmera. Normalmente se dispõe apenas da imagem de um objeto, sem nenhum outro conhecimento adicional sobre a cena. Então, o problema da reconstrução quando se opera sobre imagens de objetos, se refere à estimativa desta profundidade. Já que a imagem é a projeção 2D de uma cena 3D, a dificuldade do processo de reconstrução é estabelecer a relação de profundidade a partir da imagem projetada da cena.

Apesar de existirem tentativas de reconstrução tridimensional a partir de apenas uma imagem, através de um processo de segmentação que identifica seções da imagem (chão, teto e paredes) apresentada por Ryo (2004), a obtenção do valor da profundidade se faz de maneira eficiente através de triangulação. A obtenção da projeção perspectiva requer operações com pelo menos duas imagens do mesmo objeto, adquirida de posições diferentes, por uma câmera que se movimentou ou por duas câmeras, como mostram Jain (1995), Trucco (1998), Forsyth (2003) e Ma (2004) entre outros.

Para a compreensão sobre o processo de reconstrução espacial de um objeto, é preciso realizar um estudo preliminar sobre a geometria envolvida para as duas vistas do objeto. Os tópicos a seguir oferecem o embasamento necessário sobre o conceito de geometria epipolar e sobre todo o processo de calibração de câmeras.

3.1. Geometria de múltiplas vistas

A figura 3.1 apresenta o arranjo para um sistema de visão com duas câmeras em posições arbitrárias, contendo as projeções do ponto P no espaço. A posição de P referente a câmera 1 e a câmera 2, é dada respectivamente pelos vetores $\mathbf{P}_1=[X_1, Y_1, Z_1]^T$ e $\mathbf{P}_2=[X_2, Y_2, Y_2]^T$. Para obter as projeções de P , como descrito no capítulo 2, considera-se que o centro de coordenadas de referência O^w coincide com o centro O^c da câmera 1 ($O^w=O_1$)² e O_2 é dado pela distância b (*baseline*). Assim, com um referencial comum, os vetores $\mathbf{p}_1=[x_1, y_1, z_1]^T$ e $\mathbf{p}_2=[x_2, y_2, z_2]^T$ são conhecidos. Para todos os pontos na imagem $z_1 = f_1$ e $z_2 = f_2$.

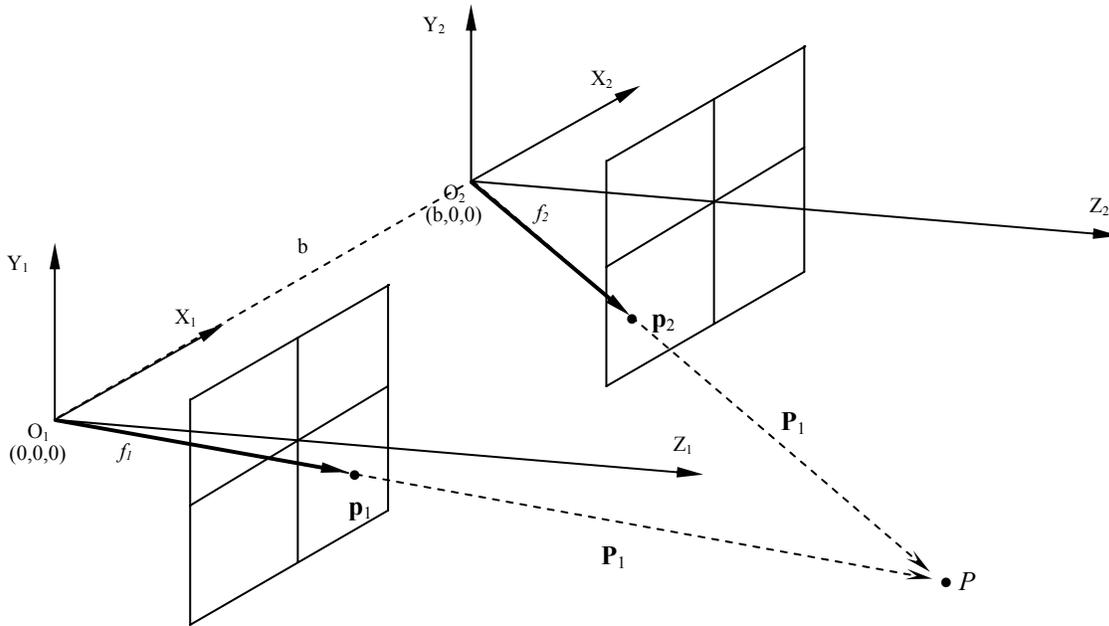


Figura 3.1: Ambiente de visão com duas câmeras. A figura apresenta as projeções 2D p_1 e p_2 do ponto $P(X,Y,Z)$ em 3D.

Da figura 3.1, pode-se verificar que os *frames* de referência das câmeras de centros O_1 e O_2 , são relacionados via parâmetros extrínsecos, contendo um vetor de translação $T = (O_2 - O_1)$ e uma matriz de rotação R .

² Por conveniência foi simplificada a notação, suprimindo-se o símbolo 'c' da identificação da câmera. Assim O^c_1 e O^c_2 se referem respectivamente aos centros da câmera 1 e 2 de centros O_1 e O_2 .

Utilizando as projeções nos planos de imagem 1 e 2, dados pelos vetores $\mathbf{p}_1=[x_1, y_1, f_1]^T$ e $\mathbf{p}_2=[x_2, y_2, f_2]^T$, a relação da câmera O_2 em função da câmera O_1 , pode ser expressa pela equação:

$$\mathbf{p}_2 = R \mathbf{p}_1 + t \quad (3.1)$$

Ou seja, o ponto \mathbf{p}_2 é projetado sobre um plano de projeção que está rotacionado (R) e transladado (t) a partir da câmera 1. A equação (3.1) que relaciona este movimento é chamada de transformação de corpo rígido, como descrito no capítulo 2.

O arranjo do ambiente multi-câmeras, pode ser representado pela geometria das projeções, chamada geometria epipolar. A teoria sobre a geometria epipolar descrita a seguir, foi baseada em Jain (1995), Trucco (1998), Forsyth (2003) e Ma (2004).

3.2. Geometria Epipolar

Considere a figura 3.2, com as imagens \mathbf{p}_1 e \mathbf{p}_2 de um ponto P observado por duas câmeras com centro ótico O_1 e O_2 . As linhas O_1P e O_2P formam um plano, denominado plano epipolar. O ponto projetado \mathbf{p}_1 está sobre a linha l_1 , onde o plano epipolar e o plano de projeção se interceptam. As linhas l_1 e l_2 são chamadas linhas epipolares, associadas ao ponto P . A linha base que une O_1 e O_2 também intercepta os dois planos de projeção. Cada ponto de interseção entre os dois, e_1 e e_2 , é chamado epipolo. O epipolo e_2 é a projeção do centro ótico O_1 da primeira câmera, na imagem observada pela segunda câmera. O epipolo e_1 é a projeção do centro ótico O_2 da segunda câmera, na imagem observada pela primeira câmera. Se \mathbf{p}_1 e \mathbf{p}_2 são imagens do mesmo ponto, então \mathbf{p}_2 deve estar sobre a linha epipolar associada com \mathbf{p}_1 . Isto é chamado restrição epipolar, e é a regra fundamental em visão estéreo e análise de movimento.

A restrição epipolar implica que os três vetores $\vec{O_1p_1}$, $\vec{O_2p_2}$ e $\vec{O_1O_2}$ sejam coplanares, e um deles está sobre o plano definido pelos outros dois, ou seja,

$$\vec{O_1p_1}[\vec{O_1O_2} \times \vec{O_2p_2}] = 0 \quad (3.2)$$

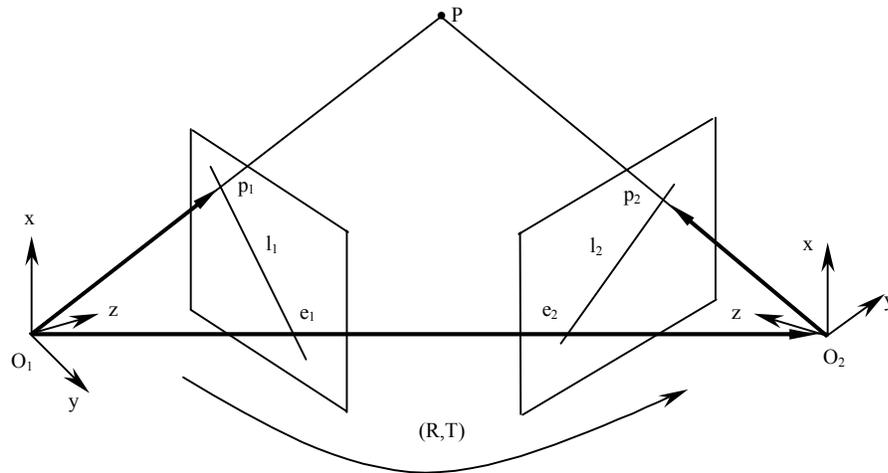


Figura 3.2: Representação da geometria epipolar adaptado de Ma(2004).

Pelas relações geométricas epipolares, pode ser dito que a projeção de um único ponto na câmera 1, corresponde a uma linha na câmera 2. Como os pontos \mathbf{p}_1 e \mathbf{p}_2 são as projeções do ponto P sobre o plano de imagem de ambas as câmeras, a projeção espacial pode ser representada pelos vetores 3D, \mathbf{p}_1 e \mathbf{p}_2 , como:

$$\mathbf{p}_1 = \begin{bmatrix} x_1 \\ y_1 \\ f \end{bmatrix} \quad \mathbf{p}_2 = \begin{bmatrix} x_2 \\ y_2 \\ f \end{bmatrix} \quad (3.3)$$

A câmera 2 tem uma rotação R , que contém a orientação de cada eixo, de acordo com o sistema de eixos da câmera 1.

O ponto P tem coordenadas espaciais \mathbf{p}_1 e \mathbf{p}_2 em relação ao sistema de eixos localizados na câmera 1 ou na câmera 2. A partir da restrição epipolar definida pela equação (3.2), a relação entre \mathbf{p}_1 e \mathbf{p}_2 é dada na forma da equação 3.4:

$$\mathbf{p}_1 \cdot [\hat{T} \times (R \mathbf{p}_2)] \quad (3.4)$$

Onde \hat{T} é uma matriz anti-simétrica descrita na equação (3.8).

Em aplicações práticas de reconstrução 3D, são conhecidas as projeções dos pontos na câmeras em coordenadas de imagem, mas a orientação da rotação relativa da câmera e a translação entre as origens não são conhecidos *a priori*. A estimação destas quantidades ocorre através da construção da matriz essencial, descrita a seguir, que é o ponto inicial para as técnicas de reconstrução 3D como afirma Ma (2004).

3.3. Estimação da matriz essencial

A restrição epipolar pode ser representada pela matriz \mathbf{E} , chamada matriz essencial, quando os parâmetros intrínsecos são conhecidos. Caso contrário, a restrição epipolar é dada pela matriz \mathbf{F} , chamada matriz fundamental como definido por Forsyth (2003) e Ma (2004). Para a estimação da matriz essencial, será considerado duas imagens da mesma cena obtidas de diferentes posições como na figura 3.1. Também está sendo assumido neste tópico, que a câmera está inicialmente calibrada, ou seja, a matriz de parâmetros intrínsecos K (definida no capítulo 2), será uma matriz identidade que relaciona \mathbf{p} e \mathbf{P} . Então,

$$\lambda \mathbf{p} = \Pi_0 \mathbf{P} \quad (3.5)$$

Onde λ é escala de profundidade desconhecida que diferencia \mathbf{p} de \mathbf{P} , já que os dois pontos estão sobre uma mesma linha, e $\Pi_0=[I,0]$ é a matriz identidade de parâmetros intrínsecos. O tópico a seguir apresenta uma descrição para obtenção da matriz de rotação R e do vetor de translação T , a partir da matriz essencial.

3.3.1. Cálculo da Posição e Orientação

As coordenadas homogêneas \mathbf{p} e as coordenadas espaciais \mathbf{P} , são relacionadas por uma escala de profundidade λ , com a seguinte configuração:

$$P_i = \lambda_i p_i, \quad i=1,2 \quad (3.6)$$

A equação de transformação de corpo rígido (3.1) pode ser reescrita em termos das profundidades como na equação (3.6), então:

$$\lambda_2 \mathbf{p}_2 = R \lambda_1 \mathbf{p}_1 + T \quad (3.7)$$

Uma matriz de translação \hat{T} pode ser criada como o produto vetorial:

$$\hat{T} = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} \quad (3.8)$$

Onde, t_x , t_y e t_z são as coordenadas do vetor de translação t . Ma (2004) define que a matriz \hat{T} tem as seguintes propriedades:

$$\hat{T}T = 0, \quad \hat{T}\mathbf{p} = T \times \mathbf{p} \quad (3.9)$$

Pré-multiplicando ambos os lados da equação (3.7) por \hat{T} obtém-se:

$$\lambda_2 \hat{T} \mathbf{p}_2 = \hat{T} R \lambda_1 \mathbf{p}_1 \quad (3.10)$$

Pré-multiplicando a equação 3.10 pelo vetor \mathbf{p}_2^T resulta:

$$\mathbf{p}_2^T E \mathbf{p}_1 = 0 \quad (3.11)$$

Onde \mathbf{E} é a matriz essencial, definida como:

$$\mathbf{E} = \hat{T}R \quad (3.12)$$

A matriz essencial contém os parâmetros de rotação e translação no sistema de referência da câmera, também conhecidos como parâmetros de *pose*. A equação (3.11) também é conhecida como restrição essencial, pois relaciona as imagens de pontos correspondentes, apesar da posição relativa entre as câmeras ser desconhecida.

É possível usar a restrição epipolar definida anteriormente, para recuperar os parâmetros de *pose* da câmera, a partir de um número de pontos correspondentes conhecidos entre as duas imagens. Para isto é demonstrado a seguir, o algoritmo dos 8 pontos apresentado inicialmente por Longuet-Higgins, como descrito em Trucco (1998), Forsyth, (2003) e Ma (2004).

3.3.2. Algoritmo dos 8 pontos

Pela descrição de Ma (2004), pode-se assumir que a matriz essencial tem a seguinte forma:

$$\mathbf{E} = \begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{bmatrix} \quad (3.13)$$

Ou, na forma de coluna:

$$(\mathbf{E}^s)^T = [e_{11} \ e_{21} \ e_{31} \ e_{12} \ e_{22} \ e_{32} \ e_{13} \ e_{23} \ e_{33}]$$

A idéia básica do algoritmo é estabelecer inicialmente k pontos correspondentes entre duas imagens. Cada correspondência de pontos forma uma equação linear homogênea. Assim, o seguinte sistema homogêneo de equações lineares é formado:

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{19} \\ a_{21} & a_{22} & \cdots & a_{29} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{k9} \end{bmatrix} \mathbf{E}^s = 0 \quad (3.14)$$

Onde, a_{kj} é o j -ésimo elemento do produto de Kronecker, do par “k” de dois vetores de projeção \mathbf{p}_1 e \mathbf{p}_2 :

$$a_k = [a_{k1} \ a_{k2} \ \cdots \ a_{kj} \ \cdots \ a_{k9}] = \mathbf{p}_1 \otimes \mathbf{p}_2 \quad (3.15)$$

Uma solução não-trivial para um sistema de equações é encontrada para $k \geq 8$ diferentes pares de projeção. Além disso, os pares a_k dos pontos projetados devem ser originados numa superfície não-coplanar do espaço. Pontos coplanares geram as chamadas superfícies críticas, que ocasionam falhas no algoritmo. Ma (2004) discute a geometria planar para o caso em que todos os pontos k pertencem ao mesmo plano 2D em \mathbb{R}^3 .

Uma solução para \mathbf{E}^S é encontrada pelo cálculo do autovetor da matriz de coeficientes do sistema homogêneo, associado ao menor autovalor. A partir da matriz essencial estimada, é possível recuperar os parâmetros de rotação da câmera R e o vetor t , usando decomposição em valores singulares (SVD - *Singular Value Decomposition*) como apresentado por Tsai (1984) e Fiori (2000). Uma matriz \mathbf{E} é uma matriz essencial, se e somente se, \mathbf{E} possui decomposição em valores singulares, $\mathbf{E} = U\Sigma V^T$, onde $U_{m \times n}$ é uma matriz de colunas ortogonais, Σ é uma matriz diagonal onde os valores da diagonal são os valores singulares de \mathbf{E} , e $V_{n \times n}$ é uma matriz ortogonal. As colunas de V são os autovetores correspondentes aos valores singulares (autovalores) de \mathbf{E} .

Ma (2004) descreve que existem exatamente quatro soluções com os parâmetros de *pose* da câmera, correspondente a uma matriz essencial de valores não nulos. Assim as expressões (3.16)³ geram 4 soluções possíveis para R e T , com \mathbf{E} e $-\mathbf{E}$ satisfazendo o mesmo conjunto de restrições epipolares.

$$(\hat{T}_1, R_1) = (UR_z(+\frac{\pi}{2})\Sigma U^T, UR_z^T(+\frac{\pi}{2})V^T) \quad (3.16)$$

$$(\hat{T}_2, R_2) = (UR_z(-\frac{\pi}{2})\Sigma U^T, UR_z^T(-\frac{\pi}{2})V^T)$$

³ A caracterização da matriz essencial dadas pelas expressões (3.16) estão completamente deduzidas em Ma (2004).

Apesar disto, somente uma das soluções garante que a profundidade de todos os pontos 3D reconstruídos são positivos em relação as duas câmeras. As demais soluções são fisicamente impossíveis e podem ser descartadas.

Como $a = \mathbf{p}_1 \otimes \mathbf{p}_2$, a restrição epipolar dada pela equação (3.11) pode ser reescrita como o produto interno de a e \mathbf{E}^s .

$$a^T \mathbf{E}^s = 0 \quad (3.17)$$

De forma prática, para um dado conjunto de pontos de correspondência $(\mathbf{p}_{k1}, \mathbf{p}_{k2})$ com $k=1,2,\dots,n$ ($n \geq 8$) o algoritmo dos 8 pontos pode ser sumariado em três passos fundamentais,

i. *Cálculo da primeira aproximação da matriz essencial:*

Através da construção da matriz homogênea (3.14) com o conjunto de pares de pontos relacionados, deve-se encontrar o vetor \mathbf{E}^s de tamanho unitário encontrando a decomposição em valores singulares de

$$a^T = U_{a^T} \Sigma_{a^T} V_{a^T}^T \quad (3.18)$$

e \mathbf{E}^s é formado pela nona coluna de V . Então o vetor coluna \mathbf{E}^s deve ser transformado numa matriz $\mathbf{E}_{3 \times 3}$.

ii. *Projeção no espaço essencial*

A matriz recuperada \mathbf{E} de maneira geral não está no espaço essencial, como apresenta Ma (2004). Ela deve ser normalizada através de $U \Sigma V^T$, onde $\Sigma = \text{diag}\{1,1,0\}$.

iii. *Recuperação do deslocamento a partir da matriz essencial*

Finalmente U e V podem ser usados para extrair R e T da matriz essencial da forma,

$$R = UR_z^T\left(\pm\frac{\pi}{2}\right)V^T \quad \text{e} \quad \hat{T} = UR_z\left(\pm\frac{\pi}{2}\right)\Sigma U^T \quad (3.19)$$

$$R_z^T\left(\pm\frac{\pi}{2}\right) = \begin{bmatrix} 0 & \pm 1 & 0 \\ \mp 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.20)$$

Como apresentado pelas equações 3.16 e mais especificamente pelas equações (3.19), o teorema para o algoritmo está baseado no princípio de que na decomposição em valores singulares de \mathbf{E} , as matrizes U e V são matrizes de rotação. Mas isto não é sempre verdade se houver presença de ruídos nas informações da imagem. Por isto o procedimento de estimação de \mathbf{E} pelo algoritmo dos 8 pontos, é formulado em termos de parâmetros e projeções da imagem de uma câmera ideal. Na prática embora o procedimento básico da estimação 3D permaneça o mesmo, algumas condições para os parâmetros de projeção devem ser estabelecidos para compensar a distorção de lentes e a deformação de *pixels* de uma câmera.

3.4. Calibração de Câmeras

O principal problema da reconstrução de um sólido é conhecer suas características geométricas, como a forma e medidas, a partir de sua projeção nos planos de imagens de um conjunto de câmeras. Num ambiente com duas câmeras, a partir dos pontos projetados, para saber a posição dos pontos no espaço é necessário fazer inicialmente a calibração da câmera para depois recuperar a estrutura 3D da cena. Os métodos de calibração mais utilizados calculam a posição relativa da câmera e seus parâmetros intrínsecos usando coordenadas 3D de pontos a partir de uma referência de calibração conhecida.

Um gabarito de referência para calibração, denominado *calibration rig*, como mostra a figura 3.3, pode ser construído de forma a oferecer um determinado conjunto conhecido de pontos relativos a um *frame* de referência. Este método de calibração pode ser usado quando um objeto puder ser colocado na cena, sem nenhuma alteração na configuração das câmeras.

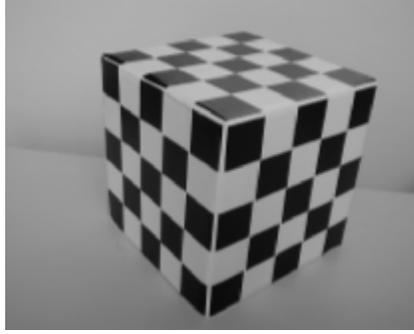


Figura 3.3: Cubo com padrão xadrez utilizado para calibração de câmeras.

Segundo o que apresentam Trucco (1998), Salvi (2002) e Forsyth (2003), a calibração se refere à obtenção das equações de projeção relacionando coordenadas conhecidas de um conjunto de pontos 3D e suas projeções. O processo pode ser decomposto em duas fases:

- i. Cálculo da matriz de projeção perspectiva associada a cada câmera, no sistema de coordenadas da câmera, como apresentado no capítulo 2;
- ii. Estimação dos parâmetros intrínsecos e extrínsecos da câmera, a partir desta matriz.

Segundo Horn (2000), a calibração interior é a recuperação da posição do centro de projeção no sistema de coordenada da imagem, chamada orientação interior. Trucco (1998) separa a calibração em intrínseca e extrínseca. Com o objetivo de evitar dificuldade de interpretação em relação as definições sobre a palavra "calibração" dadas por diversos autores, neste trabalho o termo calibração se refere a obtenção dos parâmetros intrínsecos da câmera como em Ma (2004), ou seja, a estimação da matriz K conforme equação 2.35. A estimação dos parâmetros extrínsecos, que é referida como "calibração extrínseca" por alguns autores, é referida neste trabalho pelo termo estimação dos parâmetros de *pose*, como descrito no item 3.3.1.

Conforme apresenta a equação (3.21), K é chamado de matriz de parâmetros intrínsecos ou matriz de calibração.

$$k = \begin{bmatrix} fs_x & fs_\theta & O_x \\ 0 & fs_y & O_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3.21)$$

Com os parâmetros intrínsecos K de cada câmera, podem ser relacionados os pontos de projeção \mathbf{p}' em coordenadas de imagem e o ponto \mathbf{p} em coordenadas de câmera. Em muitos processos de visão os parâmetros intrínsecos não se alteram se o foco é fixo, e conseqüentemente o problema da calibração pode ser reduzido para a determinação dos parâmetros extrínsecos como em Tommaselli (1999). Se a matriz K é uma matriz identidade, então no caso ideal, $\mathbf{p} = \mathbf{p}'$. Entretanto, num processo prático, a matriz K deve ser conhecida através de um processo de calibração. O processo de calibração destes parâmetros é apresentado por Zhang (2000).

3.5. Resumo do Capítulo

Mesmo conhecendo-se os parâmetros intrínsecos e extrínsecos, a maior dificuldade de uma análise estereoscópica, é estabelecer a correspondência entre as duas imagens. Então a restrição epipolar, dada pela matriz \mathbf{E} , limita a área de busca. Assim, qualquer ponto da uma imagem deve estar sobre a linha epipolar da segunda imagem e vice-versa.

Foi visto neste capítulo, que quando o sistema está calibrado, ou seja, a matriz de calibração K é conhecida, duas vistas do objeto com um número n de pontos são suficientes para recuperar os parâmetros de *pose* da câmera. Se K é diferente de uma identidade I , o problema pode ser adaptado, estimando-se a matriz K , de forma que os resultados ainda podem ser aplicados. Então, multiplicando ambos os lados da equação (3.28) se tem $\mathbf{x} = K^{-1}\mathbf{x}'$ o que permite retornar ao caso calibrado da equação (2.30). Em termos práticos a matriz K é normalmente desconhecida. Então o método de Zhang (2000) pode ser aplicado. Este método de calibração se baseia no fato de que existe um gabarito de configuração conhecida, que permite extrair as informações da câmera. De maneira geral, a cena vista pelo gabarito está calibrada, e então um objeto pode ser introduzido no campo de visão das câmeras de forma a obter sua reconstrução 3D. O problema da recuperação da estrutura Euclidiana de uma cena é fortemente associado, com a estimação dos parâmetros internos da câmera.

Capítulo 4

Técnicas de Segmentação da Imagem

Qualquer sistema baseado em imagens digitais pode necessitar em um momento específico de algum processamento sobre a imagem. Pode ser em relação a filtragem de ruídos, identificação de regiões importantes, detecção de contornos e extração de atributos. Um processo conhecido como segmentação é usado para decompor a imagem em informações coerentes normalmente separando os objetos da imagem em relação ao plano de fundo. Dependendo do que se precisará realizar a partir da imagem, uma das técnicas de morfologia matemática (ou a combinação delas) descritas neste capítulo e no anexo II, pode ser requerida.

4.1. Transformação e Normalização

Se um sistema de aquisição captura uma imagem colorida, e a aplicação não requer o valor da cor como informação relevante, pode-se diminuir o esforço computacional convertendo a imagem para níveis de intensidade em cinza, ou a segmentando em preto e branco. Os níveis de cinza são suficientes para permitir a distinção entre os objetos da imagem (*foreground*) em relação ao plano de fundo da cena (*background*).

Dependendo da quantidade de bits usados para representar a intensidade l , a imagem pode possuir um valor para cada *pixel* entre $0 < l < L$, onde L representa o nível máximo, e normalmente é uma potência de 2, ou seja, $L = 2^k$, com k representando a quantidade de bits por *pixel* como descrito em Gonzalez (2002). Assim, uma imagem com $k=8$ contém 256 níveis distintos de cinza, com $k=16$ são 65536 níveis diferentes.

É conveniente no processamento de imagens, sobretudo nos procedimentos de segmentação, operar com a imagem normalizada no intervalo [0,1]; assim, consegue-se uma escala genérica de intensidades, independente de qualquer valor de k .

Além da normalização, a transformação da imagem para 256 níveis de cinza pode ser importante. Uma imagem RGB, de dimensão $M \times N \times 3 \times k$ pode ser convertida em uma imagem $M \times N \times k$, pois reduz-se significativamente a quantidade de informação para o processamento. Como descrito no capítulo 2, a imagem RGB possui três planos de intensidades, um para cada cor primária. Estes três planos podem gerar uma imagem de um único plano de dimensão $M \times N$, onde o valor da intensidade de cada *pixel* é dado por:

$$I = \frac{1}{3}(R + G + B)$$

Assim, cada *pixel* irá conter a média dos valores entre os planos RGB, representando um valor de intensidade para cada ponto da imagem. Através das intensidades da imagem é possível, ainda, realizar um processo de limiarização (*thresholding*).

4.2. Limiarização

Para realizar a identificação de atributos da imagem, como a forma do contorno do objeto, ou extrair atributos como arestas e eixos de simetria, é suficiente se operar sobre uma imagem binária, ao invés de uma imagem em níveis de cinza ou colorida. Atributos geométricos da imagem podem ser obtidos através de morfologia matemática (anexo II) aplicada às imagens binárias.

O processo de limiarização permite transformar uma imagem de níveis de cinza, em uma imagem binária correspondente, onde os *pixels* do objeto de interesse possuem o valor $l=1$ (branco), e o plano de fundo da imagem contendo o valor $l=0$ (preto). Deve ser definido um valor de divisão de intensidades, chamado limiar, de forma que todos os *pixels* com valores abaixo do limiar devem ser convertidos para $l=0$, e todos os demais com valores acima do limiar, devem ser

alterados para o valor com $l=1$. Assim gera-se uma imagem binária resultante contendo apenas valores 0 ou 1 para cada um dos *pixels*.

4.3. Detecção de Bordas

A extração do contorno do objeto pode ser bastante útil, para identificação de atributos geométricos da imagem, como obtenção de formas, cálculo de áreas, definição de simetria, etc. No contexto do trabalho, as bordas do objeto são utilizadas no cálculo de coeficientes angulares do processo descrito no capítulo 6. A detecção de bordas em imagens binárias é derivada de um processo morfológico de erosão ou de dilatação, descrito a seguir.

4.3.1. Erosão binária

Dados dois conjuntos de pontos, sendo uma imagem A e um elemento estruturante B , pertencentes a um espaço bidimensional Z^2 (capítulo 2), onde cada elemento dos conjuntos tem coordenadas (x,y) associado a um valor de intensidade $l=0$ ou $l=1$, denota-se a erosão de A por B , como $A \ominus B$, da forma:

$$A \ominus B = \{Z \mid (B)_z \subseteq A\} \quad (4.1)$$

A equação 4.1 indica que a erosão de A por B é o conjunto de todos os pontos z de forma que B transladado por z está contido em A .

A figura 4.1-a apresenta uma imagem de exemplo A de dimensão $M \times N$ que se deseja erodir pelo elemento estruturante B (figura 4.1-b). O elemento estruturante em forma de cruz define o sentido de deslocamento das linhas e colunas de A para as sucessivas interseções.

O processo é ilustrado na figura 4.2. Cada direção do elemento estruturante, indicado pelo símbolo \bullet , gera uma nova imagem de A com linhas e colunas deslocadas de acordo com os valores 1 de B conforme as figuras 4.2-a,b,c,d,e. A interseção destes conjuntos gera a imagem resultante 4.2-f contendo os *pixels* de valor 1 pertencentes a todas elas simultaneamente.

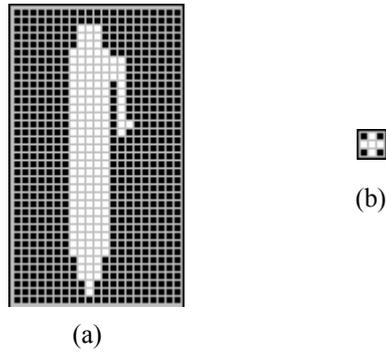


Figura 4.1: (a) imagem de exemplo $A_{M \times N}$; (b) elemento estruturante $B_{3 \times 3}$ em forma de cruz.

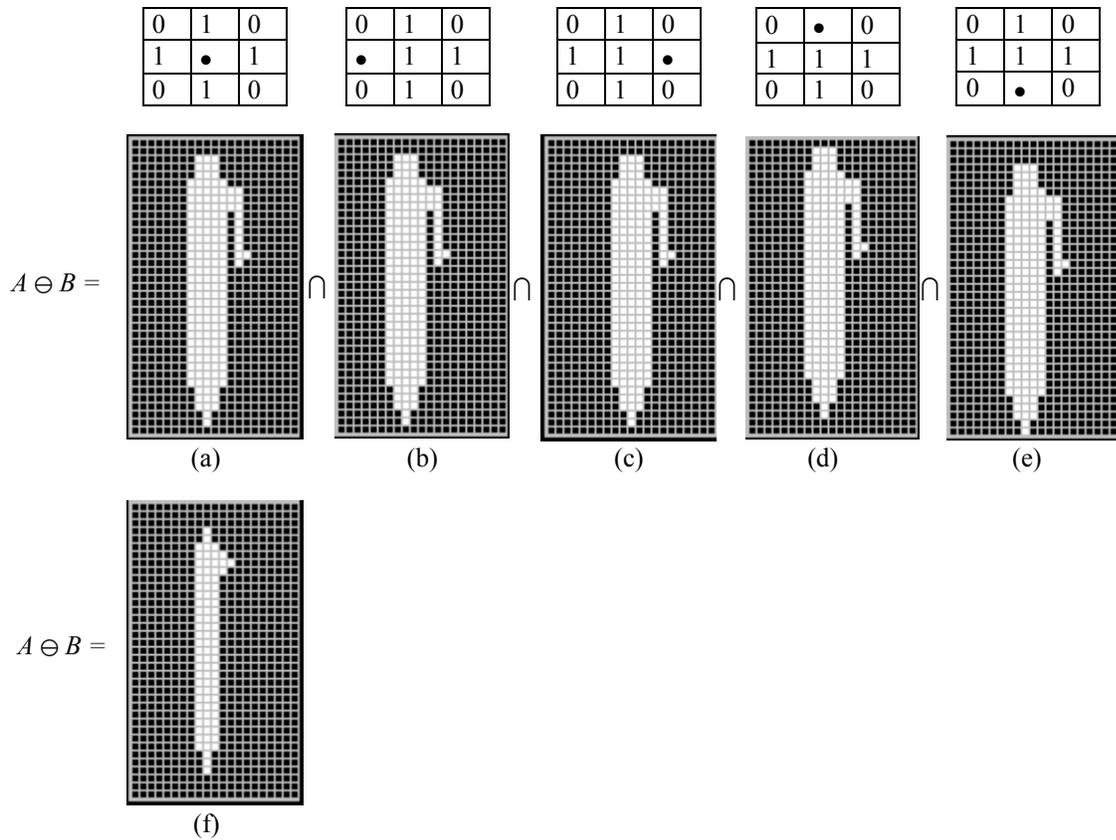


Figura 4.2: (a) imagem A original; (b) imagem A com colunas deslocadas à esquerda; (c) imagem A com colunas deslocadas à direita; (d) imagem A com linhas deslocadas para cima; (e) imagem A com colunas deslocadas para baixo.

Pode-se verificar que um processo de erosão causa uma diminuição do conjunto de resposta, como mostra a imagem resultante da figura 4.2-f. Partes do objeto foram eliminadas pelo processo, porque não houve interseção entre as imagens deslocadas. Sobre o aspecto do

objeto, a perda de informação pode descaracterizar o conjunto inicial, mas por outro lado, este processo é extremamente útil para eliminação de ruídos e separação de objetos conectados. Maiores detalhes sobre este processo, bem como as propriedades da erosão, podem ser observados respectivamente no anexo II ou em Gonzalez (2002) e Facon (1996).

4.3.2. Dilatação binária

A dilatação binária é o processo dual da erosão. A dilatação da imagem A pelo elemento estruturante B , é dado pelo símbolo $A \oplus B$, com a forma:

$$A \oplus B = \{Z | (B)_z \cap A \neq \{\}\} \quad (4.2)$$

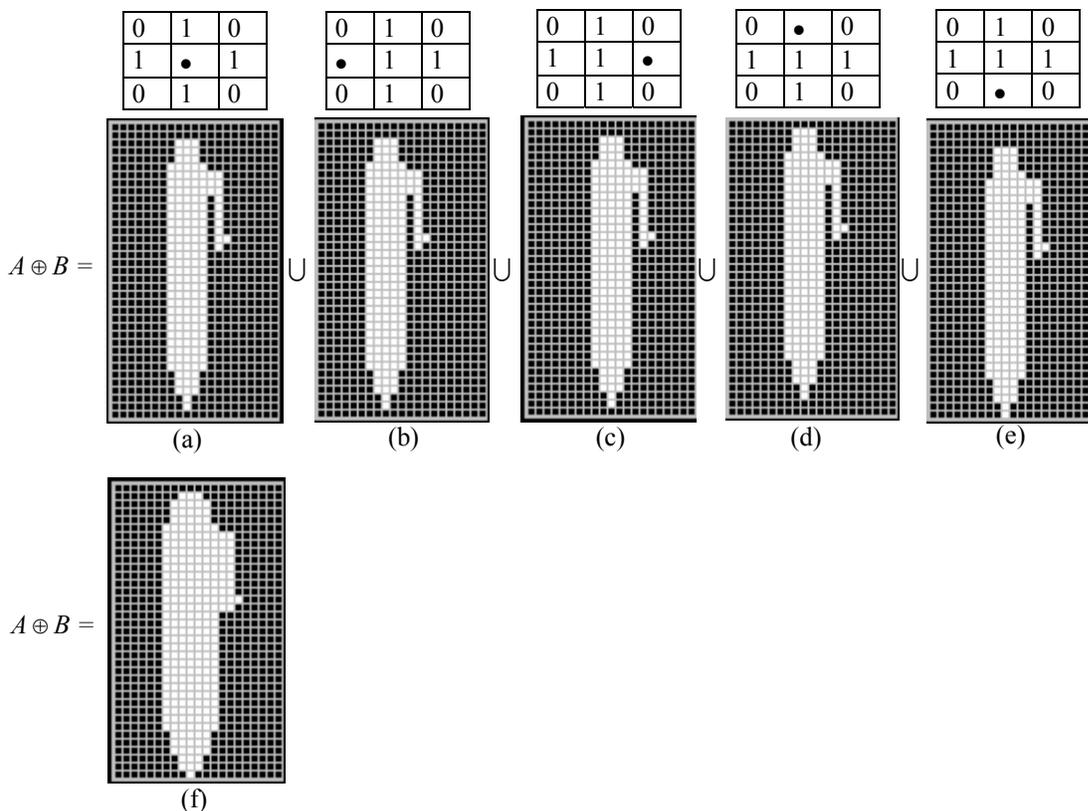


Figura 4.3: (a) imagem A original; (b) imagem A com colunas deslocadas à esquerda; (c) imagem A com colunas deslocadas à direita; (d) imagem A com linhas deslocadas para cima; (e) imagem A com colunas deslocadas para baixo. (f) imagem dilatada

Para a mesma imagem exemplo A e mesmo elemento estruturante B da figura 4.1, o processo de dilatação pode ser detalhado como mostra a figura 4.3.

Observa-se que a imagem 4.3-f resultante das sucessivas uniões, gera o aumento do conjunto de pontos inicial. Isto pode ser útil no preenchimento de furos ou conexão de partículas em uma imagem binária. Gonzalez (2002) e Facon (1996) apresentam às propriedades respectivas a dilatação.

4.3.3. Extração de contorno

O contorno de um conjunto A , denotado por $\beta(A)$, pode ser obtido pela diferença entre o conjunto inicial e o conjunto erodido, da forma:

$$\beta(A) = A - (A \ominus B) \quad (4.3)$$

como também, a partir da dilatação, assim:

$$\beta(A) = (A \oplus B) - A \quad (4.4)$$

A imagem resultante é apresentada na figura 4.4.

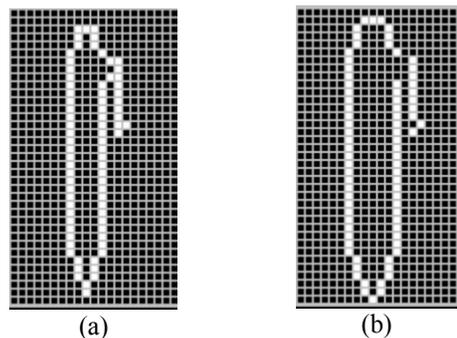


Figura 4.4: (a) Imagem resultante a partir da expressão 4.3. (b) imagem resultante a partir da expressão 4.4.

Pode-se verificar através da figura 4.4-a, que a realização da subtração da erosão a partir da imagem original, conforme a expressão 4.3, preserva a silhueta original da imagem. Já com a utilização do processo pela equação 4.4, a imagem apresenta um aumento de tamanho em função da dilatação, que altera algumas características da borda. A utilização da expressão 4.3 é suficiente para a extração de bordas de uma imagem binária.

4.4. Resumo do Capítulo

Este capítulo apresentou a descrição de algumas técnicas de morfologia matemática aplicadas a imagens binárias, que são essenciais para o entendimento dos processos descritos nos capítulos que se seguem. Tanto o procedimento de extração de bordas, quanto o processo de afinamento são utilizados na descrição do processo de criação da caixa de contorno no capítulo 6.

É importante ressaltar que a extração de atributos da imagem, poderia ser feita diretamente da imagem em níveis de cinza ou dos índices da imagem indexada. A extração de bordas, por exemplo, pode ser resultado da aplicação de filtros como *sobel*, *prewit* e *canny*, também descritos em Gonzalez (2002). No entanto, optou-se pela utilização de imagens binárias, porque além da extração de contorno, é possível representar o eixo de simetria do objeto usando uma descrição por afinamento.

Capítulo 5

Caixas de contorno (*Bounding Box*)

Reconstruções tridimensionais são utilizadas com diversos propósitos nos processos onde a informação de profundidade é essencial para o entendimento de uma cena. Uma cena é formada por uma estrutura complexa de elementos variados, que nem sempre são necessários num procedimento de reconstrução. Alguns pontos de referência do ambiente podem ser suficientes para determinar a movimentação de um robô, identificar a presença de um objeto e recuperar suas características geométricas básicas. A simplificação do ambiente pode reduzir o esforço computacional pela supressão de informações que não serão necessárias na análise de uma cena. A diminuição da quantidade de informações pode melhorar a eficiência do processo sem comprometer o resultado final.

Uma parte essencial deste trabalho trata da obtenção do espaço ocupado por um objeto, sem a necessidade da completa reconstrução de sua superfície, como apresentado em Kurka (2005). Para realizar isto, está sendo proposta a criação de uma caixa de contorno (*bounding box*), que se adapte ao contorno de um objeto e permita extrair informações sobre a sua posição e sobre o volume ocupado. A idéia da utilização das caixas de contorno foi primeiramente apresentada por Gottschalk(2000), para a identificação de colisão de objetos que estão em movimento. Alguns exemplos de aplicação das caixas de contorno podem ser vistos em Barequet (2001) que usa este conceito para indexação e recuperação de imagens, em Chan (2003) para processos de empacotamento, em Shin (2003) para representação de superfícies, em Coma (2003) em projeto de montagem (*Design for Assembly – DFA*), e em Majchrzak (2004) para realizar análise do movimento de objetos em cenas bidimensionais. Em Chan (2005) é apresentada a definição de uma caixa de contorno, para simplificar a representação de um objeto 3D, a partir de um modelo

CAD com a utilização de imagens sintéticas. A proposta do trabalho é ir além da criação de um enquadramento para objetos.

Está sendo apresentado na figura 5.1 o exemplo de uma caixa de contorno para ilustrar o ajuste a um objeto.

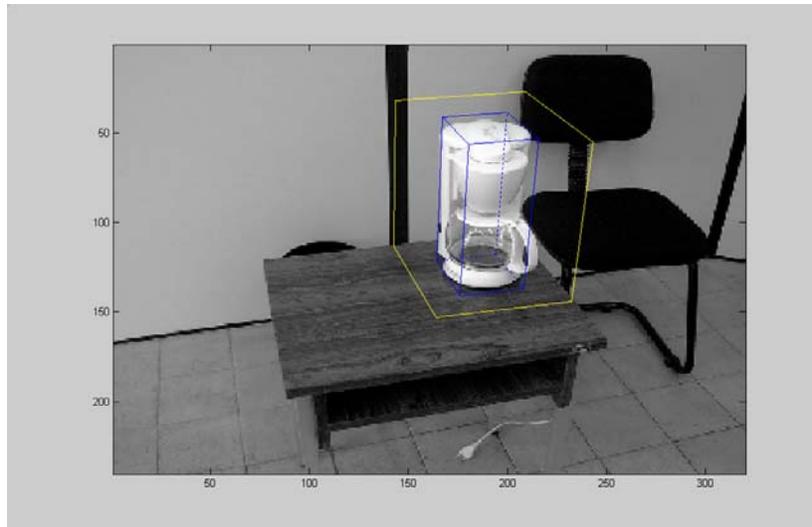


Figura 5.1: Exemplo de uma caixa de contorno ajustado a um objeto.

Através da figura 5.1 pode-se verificar a existência da moldura da caixa de contorno inicial, formada pelas linhas mais externas ao objeto. Também, na mesma figura pode ser vista a caixa de contorno ajustada ao objeto, representado pelas linhas mais internas próximas ao objeto.

Alguns métodos heurísticos têm sido propostos para a determinação de uma caixa de contorno que circunscreve um conjunto de pontos. Uma forma é se basear nas coordenadas de uma nuvem de pontos como em Barequet (2001). Em Shin (2003), além de se usar nuvens de pontos, cria-se vetores de gradientes baseados na distribuição do histograma dos pontos pertencentes às faces do objeto. O pico de nível mais elevado do histograma é o que se aproxima mais da face da caixa de referência. Assim é possível determinar a posição das arestas para cada face da caixa de contorno sobre o modelo. Outra maneira é apresentada em Chan (2003) onde a geração da caixa de contorno é baseada na projeção da imagem sintética de um sólido sobre três planos principais. Assim as orientações dos contornos projetados, são usadas para aproximação da posição do sólido.

De forma diferente aos métodos apresentados, este trabalho descreve no capítulo 6, o processo de construção de uma caixa de contorno baseado na reconstrução tridimensional das imagens projetadas de uma caixa de calibração usada como referência inicial.

Dois tipos de caixas de contorno são descritos por Gottschalk (2000). Um deles trata do enquadramento pelo método dos eixos alinhados, chamado AABB (*Axis Aligned Bounding Box*). Outra maneira é calcular o enquadramento através de eixos orientados, chamado OBB (*Oriented Bounding Box*). Os dois tipos serão usados como referência para a criação de uma caixa de contorno pelo método sugerido neste trabalho. A diferença entre eles é sintetizada nos itens 5.1 e 5.2 a seguir.

5.1. Caixa de contorno alinhada (AABB)

Um AABB é um paralelogramo cujas faces podem ser representadas como a extensão dos eixos ao longo de um *frame* de referência. Neste caso o ajuste requer que o objeto esteja geometricamente posicionado de acordo com uma caixa de contorno de aproximação inicial (*outer box*). A definição para um AABB, pode ser dada como em Gottschalk (2000). Denominando de B a região mapeada pela caixa de contorno, se tem a seguinte definição:

$$B = \{(x, y, z) \mid |c_x - x| \leq r_x, |c_y - y| \leq r_y, |c_z - z| \leq r_z\} \quad (5.1)$$

Onde \mathbf{c} é o centro de massa e os eixos das faces são respectivamente r_x , r_y e r_z . A figura 5.2 apresenta um exemplo, para $z = 0$, com os eixos r_x e r_y , ajustados em função das dimensões do objeto.

Diz-se que a caixa de contorno é mínima se as suas arestas estão na menor distância possível das bordas do objeto. O esboço da figura 5.2 representa através das linhas pontilhadas um bom ajuste para um AABB, onde o enquadramento ocorre muito próximo ao contorno do objeto.

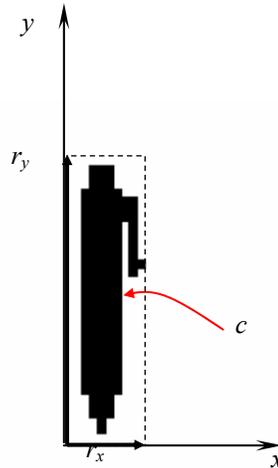


Figura 5.2: Esboço para um AABB com ajuste mínimo.

A figura 5.3 apresenta o ajuste do mesmo AABB para um objeto com alinhamento diferente da posição dos eixos de referência do sistema.

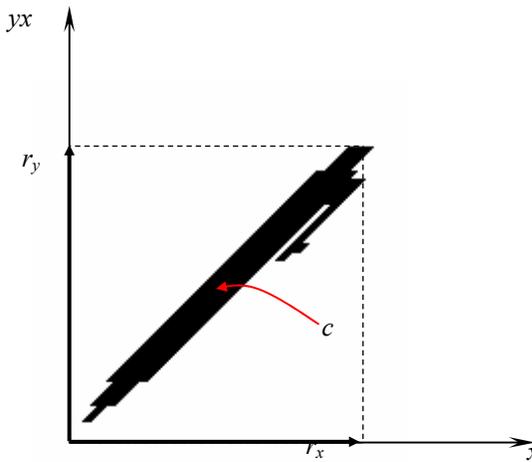


Figura 5.3: Esboço para um AABB sem ajuste mínimo.

Para o caso da figura 5.3, os eixos de orientação do objeto não correspondem à posição dos eixos de referência. As arestas da caixa de contorno não se ajustam à silhueta do objeto. Isto implica que o ajuste do AABB é comprometido, pois não representa uma caixa de contorno mínima para o objeto.

Em Kurka (2005) é introduzido o conceito de *range cube* para denominar um AABB

tridimensional, que é construído a partir do próprio gabarito de calibração de câmeras. Em Rudek (2005) é descrito o método baseado nos pontos 2D da própria imagem, que permite criar uma caixa de contorno através da extração de pontos de fuga. Todos os procedimentos de enquadramento da caixa de contorno usadas em Rudek (2005) são completamente apresentados no capítulo 6.

Para resolver o problema do ajuste mínimo demonstrado na figura 5.2, pode ser criada uma caixa de contorno do tipo OBB orientado pelas direções principais do objeto. O item 5.2 a seguir descreve a obtenção dos parâmetros de orientação.

5.2. Caixa de contorno orientada (OBB)

Um OBB é um paralelogramo orientado de forma arbitrária, que traz consigo a informação sobre a sua orientação. Ele pode ser representado por um ponto central \mathbf{c} , com comprimentos de bordas orientadas r_x , r_y e r_z , e de orientação fornecida por três vetores v_1 , v_2 e v_3 mutuamente ortogonais. Sua representação é dada pela expressão:

$$B = \{ \mathbf{c} + ar_1v^1 + br_2v^2 + cr_3v^3 \mid a, b, c \in [-1,1] \} \quad (5.2)$$

Os vetores v_1 , v_2 e v_3 são as três colunas da matriz de rotação R . A figura 5.4 apresenta um esboço para uma caixa de contorno orientada, com $z = 0$.

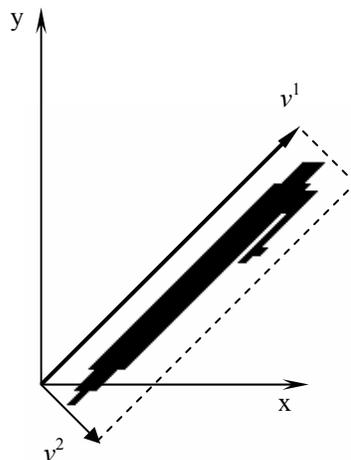


Figura 5.4: Esboço de um OBB orientado segundo os eixos de simetria do objeto.

Para a construção do OBB, o primeiro passo é descobrir uma orientação principal do objeto e em seguida, encontrar a orientação do sistema de eixos. O texto sugere duas estratégias. Uma se refere à análise dos componentes principais e outra pelos momentos e eixos principais de inércia.

5.2.1. Análise dos Componentes Principais

A análise de componentes principais é uma técnica estatística utilizada para redução do número de variáveis, através da transformação das variáveis originais em novas variáveis denominadas componentes principais. A determinação destas componentes é dada pelo cálculo dos autovalores e seus respectivos autovetores, a partir uma matriz de covariâncias ou de uma matriz de coeficientes de correlação.

Um autovalor é um escalar λ_k que satisfaz a relação,

$$|\mathbf{A} - \lambda_k \mathbf{I}| = 0 \quad (5.3)$$

Onde \mathbf{A} é uma matriz $n \times n$, \mathbf{I} é a matriz identidade e $k=1, \dots, n$. Em termos geométricos o primeiro autovalor representa o eixo de maior comprimento e o segundo autovalor um eixo ortogonal ao primeiro. Dada uma matriz \mathbf{A} que contém as coordenadas de dois pontos no plano da seguinte forma,

$$\mathbf{A} = \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \end{bmatrix} \quad (5.4)$$

Os seus respectivos autovalores podem ser obtidos da relação (5.3). Os autovalores λ_1 e λ_2 contêm a informação sobre o tamanho do maior e menor eixo de uma elipse que passa pelos pontos definidos pela matriz \mathbf{A} . Os autovalores podem ser representados graficamente como ilustra a figura 5.5. Cada autovalor possui um autovetor correspondente de forma que,

$$\mathbf{A} \mathbf{v}_k = \lambda_k \mathbf{v}_k \quad (5.5)$$

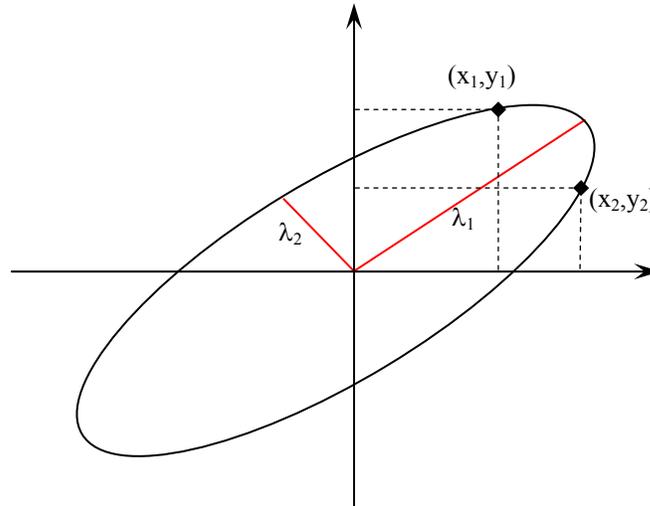


Figura 5.5: Representação gráfica das distâncias máxima e mínima dada pelos autovalores.

De acordo com a figura 5.5 os autovetores geram as direções principais e o comprimento de cada eixo é dado pelos seus autovalores correspondentes. Em uma matriz simétrica existirão tantos autovalores quanto a quantidade de linhas da matriz. Um agrupamento de pontos obtidos da imagem pode gerar uma matriz de covariância ou de correlação. Estas matrizes são sempre simétricas. A obtenção dos autovalores e autovetores destas matrizes permitem obter os eixos principais do conjunto de pontos.

Como apresentado por Castleman (1996) e aplicado por Gottschalk (2000), os *pixels* do objeto podem ser descritos através de uma matriz de covariância C , e sua localização pelas coordenadas de um centro de massa c . A variância da projeção dos pontos é dada por $v^T C v$. As direções que maximizam ou minimizam estas quantidades são os autovetores de C expresso da forma,

$$C_x = \frac{1}{L} \sum_{i=1}^L x_i x_i^t - m_x m_x^t \quad (5.6)$$

Onde ,

$$m_x = \frac{1}{L} \sum_{i=1}^L x_i \quad (5.7)$$

O i -ésimo segmento de linha tem comprimento denotado por L_i , e seu ponto médio por \mathbf{m}_i . Os elementos da diagonal da matriz C são as variâncias e os elementos que estão fora da diagonal são as covariâncias. Gottschalk (2000) faz a dedução completa da matriz C . Uma propriedade fundamental da matriz de covariância, é que a extensão dos eixos, na maior e na menor direção é sempre perpendicular em qualquer dimensão.

A extração dos componentes principais pode ser obtida usando a decomposição em valores singulares (SVD). Os autovetores associados aos valores singulares fornecem a orientação do conjunto de pontos. A figura 5.6, apresenta um agrupamento de pontos com orientação desconhecida. Cada ponto do agrupamento gera um vetor de coordenadas da forma $\mathbf{p} = [x_i \ y_i]^T$.

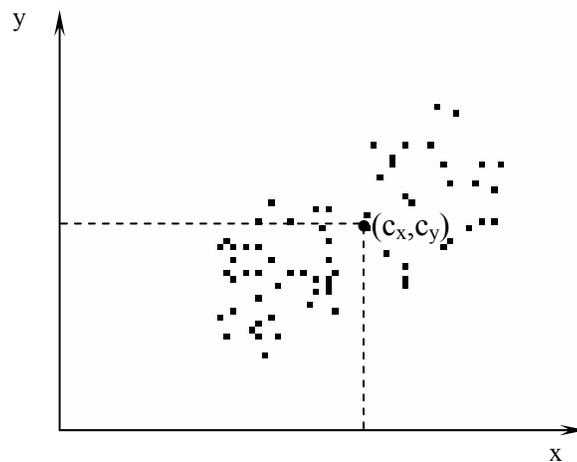


Figura 5.6: Representação do centro de massa.

Se a nuvem de pontos for transladada de forma que seu centro de massa se posicione sobre a origem, obtém-se $\mathbf{p}' = [(x_i - c_x), (y_i - c_y)]^T$, ou seja,

$$\mathbf{p}'_i = \mathbf{p}_i - \mathbf{c} \quad (5.8)$$

Onde \mathbf{c} é o centro de massa da nuvem de pontos. Uma matriz X do conjunto de pontos pode ser formada como,

$$X = \begin{bmatrix} | & | & \cdots & | \\ \mathbf{p}'_1 & \mathbf{p}'_2 & \cdots & \mathbf{p}'_n \\ | & | & \cdots & | \end{bmatrix} \quad (5.9)$$

O cálculo de C_x pode ser obtido pelas expressões 5.6 e 5.7. Os elementos desta matriz são os autovalores de X . Os componentes principais podem ser obtidos através da SVD da matriz X . Uma matriz A pode ser formada de maneira que os seus elementos sejam os autovetores de C_x . Assim, os eixos principais são os autovetores de S , da forma:

$$S = A C_x A^T \quad (5.10)$$

Onde os valores da diagonal de S são os autovalores da matriz de covariância, como:

$$C_x = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_d \end{bmatrix} \quad (5.11)$$

Onde λ_i é o i -ésimo autovalor. O maior valor de S entre todos os vetores unitários de A , está associado a seu máximo autovalor λ_1 . Se v_1 é seu autovetor correspondente então $x' = v_1$. O segundo autovalor λ_2 corresponde ao segundo autovetor $y' = v_2$. O eixo x' e o eixo y' são sempre ortogonais e representam os eixos principais que definem a orientação da nuvem de pontos. A figura 5.7 apresenta o arranjo destes vetores.

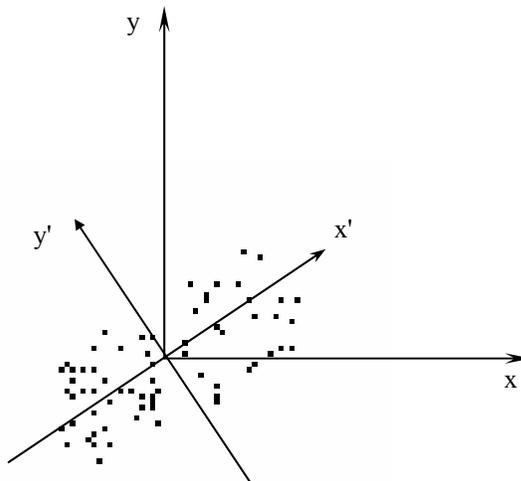


Figura 5.7: Representação dos componentes principais ajustada à nuvem de pontos.

Como apresentado na figura 5.7, a melhor linha de ajuste à nuvem de pontos atravessa o centro de massa até a origem. Invertendo a translação que levou o centro de massa para a origem, se produz o melhor ajuste para o conjunto de pontos.

O algoritmo sugerido para a obtenção dos eixos é dado pelo fluxograma da figura 5.8.

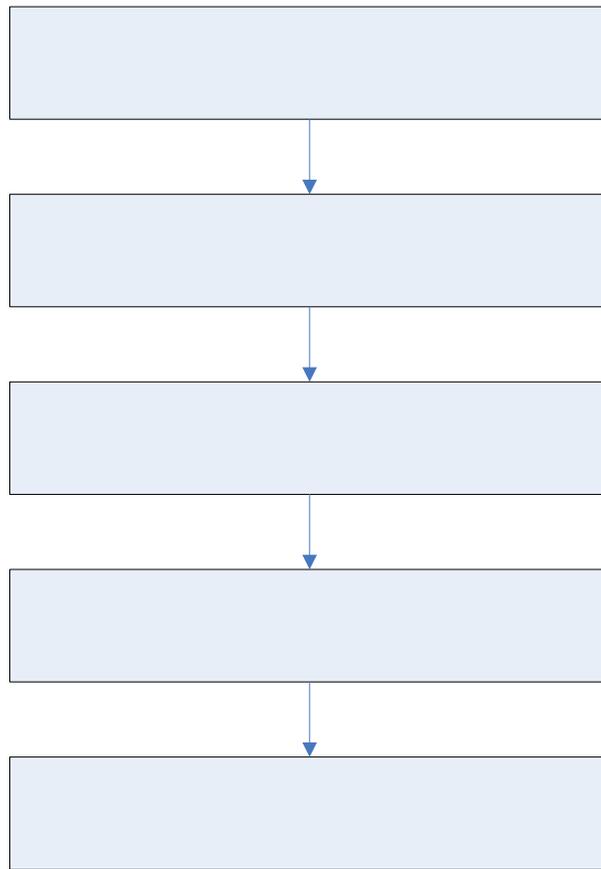
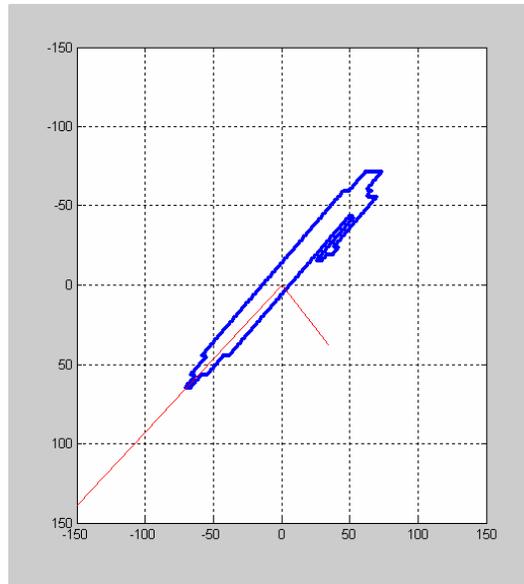
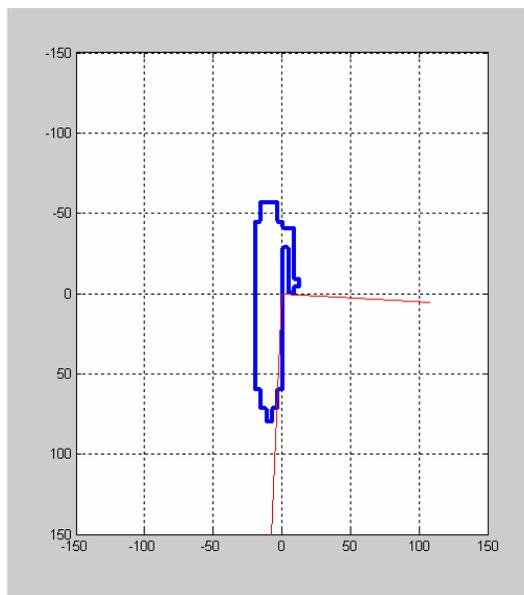


Figura 5.8: Seqüência de operações para obtenção dos eixos principais.

A figura 5.8 apresenta graficamente a seqüência de operações para a obtenção dos eixos principais de uma imagem. Este algoritmo deve ser aplicado sobre uma imagem segmentada onde o agrupamento de pontos se refere ao contorno do objeto. A figura 5.9 apresenta o resultado da localização de eixos para uma imagem sintética.



(a)



(b)

Figura 5.9: Geração dos eixos principais para uma imagem sintética.

A figura 5.9 apresenta os eixos principais do objeto obtidos pelo método da análise dos componentes principais. Este sistema de eixos define a orientação para a construção de uma caixa de contorno tipo OBB. O exemplo de ajuste do OBB será apresentado mais a frente no capítulo 6. Outra maneira de se obter os eixos principais de um conjunto de pontos é através dos momentos de inércia. O item 5.2.2 a seguir apresenta o método.

5.2.2. Obtenção dos eixos principais planos pelos momentos de inércia

De forma diferente da apresentada anteriormente, os eixos principais de um objeto podem ser entendidos como aqueles que possuem o momento de inércia máximo e mínimo um em relação ao outro, e, portanto, são perpendiculares entre si. As definições de momento de inércia, produto de inércia e eixos principais no espaço tridimensional são descritos por Hahn (2004) e Kwon (2006). Em Soares (2006) é descrito que um sistema de coordenadas no plano x_1y_1 de centro **cg**, está deslocado de uma distância a e b em relação ao eixo de referência xy da imagem. A relação entre os eixos e os momentos pode ser vista na figura 5.10, e é dada como:

$$x = x_1 + a \quad \text{e} \quad y = y_1 + b \quad (5.12)$$

As relações apresentadas em 5.12 podem ser visualizadas através da figura 5.10. Assim, chega-se a relação:

$$M_x = M_{x_1} + bS \quad \text{e} \quad M_y = M_{y_1} + aS \quad (5.13)$$

Onde M_x é o momento estático em relação ao eixo X e M_y é o momento estático em relação ao eixo y , e S é a área total da seção.

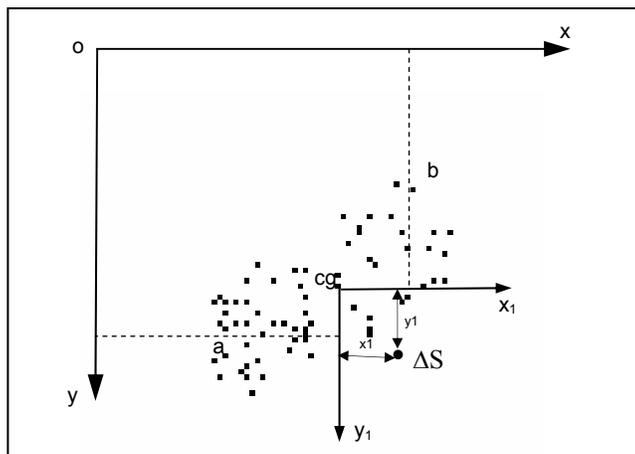


Figura 5.10: Translação de eixos e localização do centróide cg do objeto.

Os eixos x_1y_1 são chamados eixos centrais e a interseção entre eles é o centro de gravidade de centro **cg** do objeto plano. A partir destes eixos de referência é possível encontrar os eixos principais do objeto no plano. Será denotado de x' o eixo de momento mínimo e y' o de momento máximo, como apresenta a figura 5.11.

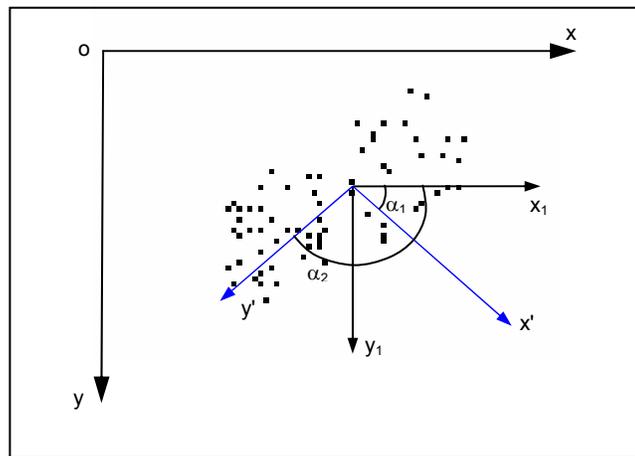


Figura 5.11: Eixos principais de inércia.

Os eixos principais de inércia x' e y' no plano podem ser obtidos pelas seguintes relações, descrito em Soares (2006), onde, os elementos I_{xx} e I_{yy} são os momentos de inércia e I_{xy} é o produto de inércia.

$$I_{x'} = \frac{(I_{xx} + I_{yy})}{2} + \sqrt{\left(\frac{I_{xx} - I_{yy}}{2}\right)^2 + I_{xy}^2} \quad (5.14)$$

$$I_{y'} = \frac{(I_{xx} + I_{yy})}{2} - \sqrt{\left(\frac{I_{xx} - I_{yy}}{2}\right)^2 + I_{xy}^2} \quad (5.15)$$

$$\tan \alpha_{x'} = \frac{(I_{xx} - I_{x'})}{I_{xy}} \quad (5.16)$$

$$\tan \alpha_{y'} = \frac{(I_{xx} - I_{y'})}{I_{xy}} \quad (5.17)$$

O momento de inércia I_{xx} é obtido pela correlação dos pontos $y \cdot y^T$ das coordenadas $\mathbf{p}=[x,y]^T$ do contorno da imagem. O momento de inércia I_{yy} é dado pela correlação de $x \cdot x^T$ das coordenadas $\mathbf{p}=[x,y]^T$ do contorno. O produto de inércia I_{xy} é dado pela correlação $x \cdot y^T$ das coordenadas da borda da imagem.

Os ângulos α_1 e α_2 definem a nova orientação para os eixos. Seus comprimentos são dados pelos pontos de mínimo e máximo das coordenadas de borda da imagem em função de cada ângulo obtido. A figura 5.12 apresenta graficamente os eixos obtidos a partir de uma imagem sintética.

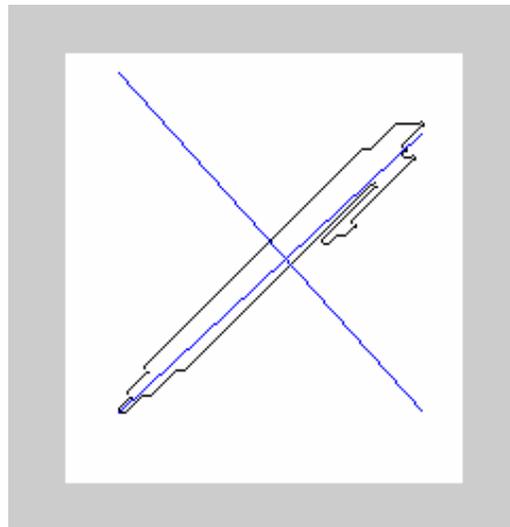


Figura 5.12: Eixos principais do objeto.

A figura 5.12 apresenta graficamente o posicionamento dos eixos principais do objeto, obtidos a partir do cálculo dos seus momentos e produtos de inércia. Um exemplo aplicado a imagens reais é apresentado no capítulo 6 com a construção de uma caixa de contorno OBB.

5.3. Resumo do capítulo

O capítulo apresentou o processo de enquadramento de um objeto através de uma caixa de contorno (*bounding box*). Existem duas formas de construção. Uma AABB é de construção mais direta, pois usa o *frame* de referência da imagem como base para construção das arestas. Este método tem aplicação quando o objeto é simétrico e possui a mesma orientação do *frame* de referência. Pode-se dizer que um AABB pode ser usado quando o objeto em análise não muda durante o processo. Se o objeto estiver completamente enquadrado pela caixa de contorno AABB seu volume será muito próximo do volume original. Se houver uma mudança de objeto ou uma alteração de posição, o volume não corresponde mais ao esperado. Assim se todas as suas características geométricas são sempre conhecidas, o ajuste da caixa de contorno pode indicar a diferença de posição pela diferença no volume ocupado.

O texto apresentou também a definição de um OBB. Para a construção da caixa de contorno orientada, deve-se obter a orientação do objeto para o correto ajuste da caixa de contorno. Foram apresentadas duas estratégias para obtenção das direções principais. A primeira usa a definição da análise dos componentes principais (PCA) para estabelecer os eixos associados a um conjunto de pontos. A segunda forma é a obtenção dos eixos principais pelos momentos e produtos de inércia da nuvem de pontos.

A determinação de uma caixa de contorno para o enquadramento de um sólido, simplifica a complexidade da representação tridimensional do objeto. Assim, apenas a projeção 3D dos eixos principais e do contorno são necessários para análise de posição e volume. Para posições aleatórias do objeto, com orientação diferente do *frame* de referência um enquadramento por OBB é mais preciso, pois se chega muito próximo ao valor de volume mínimo. A escolha do método de enquadramento, por AABB ou por OBB, depende exclusivamente da aplicação onde serão utilizados.

Capítulo 6

Método de Reconstrução Proposto

Este capítulo propõe um método para realizar a reconstrução de uma caixa de contorno baseada em um par de imagens do objeto. O método proposto se baseia na criação de uma caixa de contorno em torno do objeto, de forma que sua posição e o seu volume correspondam a uma estimativa de valores referentes aos do objeto.

Os métodos de ajuste da caixa de contorno apresentados no capítulo 5 operam basicamente de duas maneiras; - ou realizam o ajuste de uma caixa de contorno 2D para objetos no plano sem reconstrução espacial, ou criam caixas 3D sobre imagens sintéticas tridimensionais. O método de criação da caixa de contorno proposta neste capítulo, difere dos métodos citados no capítulo anterior. É apresentada uma maneira de ajustar uma caixa de contorno 3D baseado nas projeções 2D das imagens da caixa de calibração. O processo envolve duas etapas. Primeiro, as projeções dos vértices e arestas da caixa de calibração inicial servem como referência para a obtenção dos seus pontos de fuga (*vanishing points*). A partir dos pontos de fuga conhecidos é feita a aproximação das faces ao objeto, criando uma caixa de contorno ajustada, que tangencia as projeções das bordas do objeto. A aproximação das faces da caixa é baseada na obtenção de novas arestas a partir dos pontos de fuga calculados anteriormente. A segunda etapa ocorre após o ajuste das arestas, e trata da reconstrução espacial da caixa de contorno ajustada. Assim, para se construir um AABB, apenas os vértices principais da caixa de referência precisam ser conhecidos. A caixa de referência usada para criar o AABB inicial pode ser usada também para calibração dos parâmetros intrínsecos do sistema.

O texto apresenta inicialmente os passos necessários para realizar a calibração da câmera, conforme a teoria apresentada no capítulo 3. Em seguida descreve o processo para a identificação da região de interesse que contém o objeto, com o uso das ferramentas de segmentação vistas no capítulo 4. O foco principal do texto é apresentar o ajuste de arestas e a reconstrução espacial de uma caixa de contorno do tipo AABB baseada na caixa de calibração inicialmente conhecida. De forma complementar, também é sugerido o método para a criação de um OBB.

6.1. Procedimento de Calibração da Câmera

Para calibração da câmera e obtenção dos parâmetros de *pose*, foi construído um gabarito tridimensional conforme o modelo mostrado na figura 3.3 (capítulo 3). O gabarito possui a forma de um cubo de lados $L_1=L_2=L_3\cong 200$ mm. A dimensão de cada quadrado do padrão xadrez possui valores $l_1=l_2= 40$ mm. A obtenção dos parâmetros intrínsecos é realizada pelo método de Zhang (2000), conforme descrito no capítulo 3. O método é baseado na homografia planar. Para realizar o processo de calibração da câmera, os pontos de correlação devem estar no mesmo plano. Então deve ser utilizada somente uma das faces do cubo. Em Zhang (2000) é fornecido um *toolbox* com as ferramentas desenvolvidas em *MatLab* para a configuração dos parâmetros de calibração.

A figura 6.1 apresenta a face selecionada do gabarito utilizada na calibração, onde estão representados os dois eixos do plano.

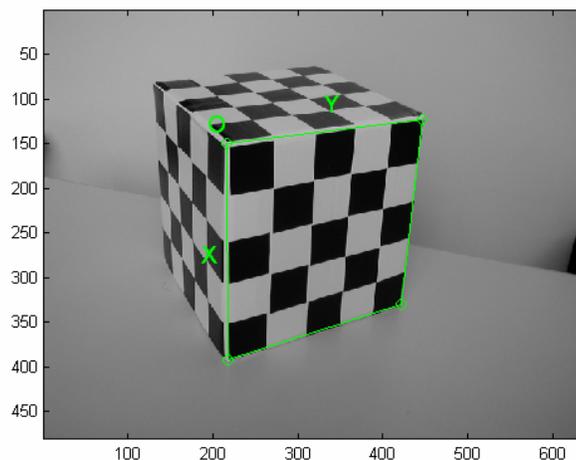


Figura 6.1: Marcação dos eixos do plano de calibração.

Após a criação dos eixos de orientação da face, todos os cantos da face são detectados automaticamente. A figura 6.2 apresenta uma das vistas do gabarito de calibração onde são exibidos os cantos encontrados. Se o ajuste não for bom, o processo deve ser executado novamente. O programa desenvolvido por Zhang (2000) permite identificar todos os cantos do padrão quadriculado, gerando uma correspondência de pontos entre o par de imagens do gabarito.

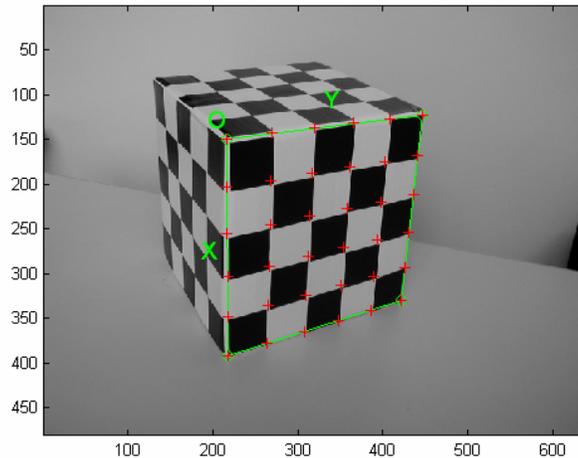
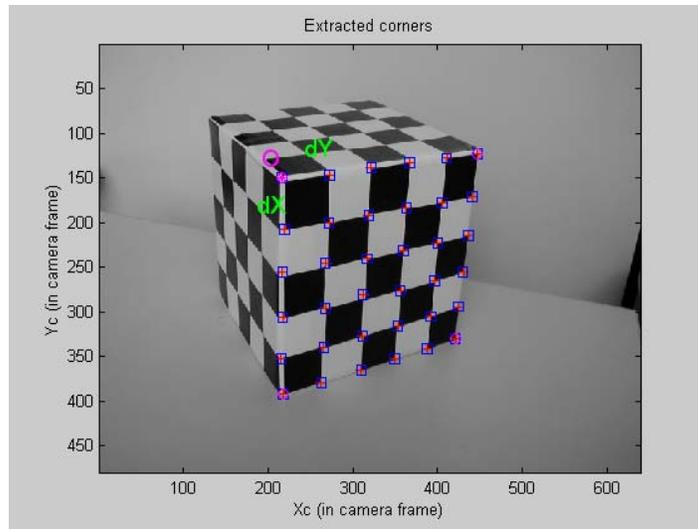


Figura 6.2: Marcação dos cantos do *grid* de calibração.

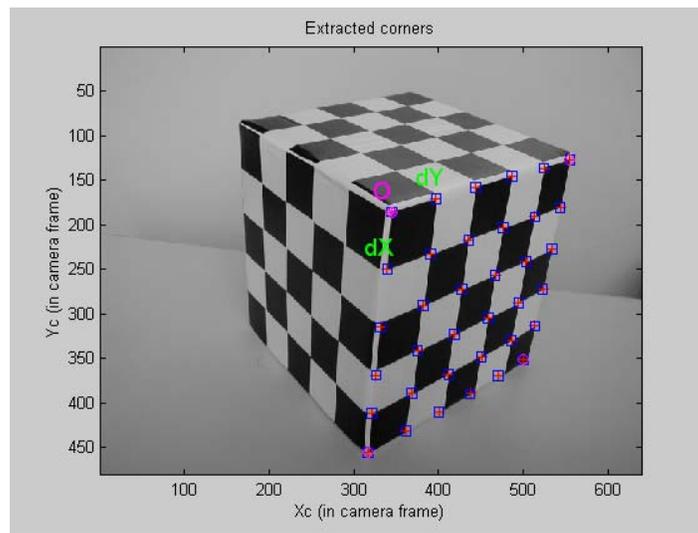
A correspondência entre pontos é um passo preliminar para se estimar os parâmetros da câmera. Os métodos automáticos de correspondência (*matching*) se baseiam na busca de atributos geométricos mais facilmente identificáveis, como interseções de arestas que formam os cantos de um objeto na imagem.

O detector de cantos de Harris e Stephens proposto por Harris (1988) e implementado no *toolbox* de Zhang (2000), permite encontrar as correspondências entre duas imagens de forma automática, gerando um arquivo de associações entre os pontos das duas imagens, que é usado posteriormente na calibração.

A figura 6.3 apresenta um exemplo da correspondência entre a imagem da esquerda e da direita do gabarito de calibração. Pode-se observar que todos os cantos do padrão xadrez foram localizados.



(a)



(b)

Figura 6.3: Exemplo da detecção de cantos pelo método de Zhang (2000) para as duas vistas.

Para Oisel (2003), a recuperação da informação 3D pela estimação dos parâmetros da câmera consiste em associar alguns valores arbitrários para parâmetros intrínsecos e então estimar os extrínsecos. De maneira mais eficiente, o processo de estimação dos parâmetros intrínsecos pode ser obtido pelo método de Zhang (2000) como descrito no capítulo 3. Os resultados obtidos através deste método são mostrados na figura 6.4, que apresenta um exemplo dos valores encontrados para a matriz de calibração, baseado nas imagens da figura 6.3-a e 6.3-b.

Definição dos parâmetros intrínsecos	
<i>Focal Length (fc):</i>	[1697.31000 1697.31000]
<i>Principal point (cc):</i>	[799.50000 599.50000]
<i>Skew (alpha_c):</i>	[0.00000]
<i>Distortion (kc):</i>	[0.00000 0.00000 0.00000 0.00000 0.00000]

Tabela 6.1: Exemplo da calibração gerado pelo Toolbox em Matlab, de Zhang (2000).

Os parâmetros apresentados na tabela 6.1 compõem a matriz de calibração K , da forma definida pela equação (2.35), como o exemplo:

$$k = \begin{bmatrix} fs_x & fs_\theta & O_x \\ 0 & fs_y & O_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$k = \begin{bmatrix} 1697.31000 & 0 & 799.50000 \\ 0 & 1697.31000 & 599.50000 \\ 0 & 0 & 1 \end{bmatrix}$$

As informações geradas pelo programa, conforme apresentado na figura 6.4, são utilizadas no processo de obtenção dos parâmetros extrínsecos, referente aos parâmetros de *pose* da câmera. A obtenção destes parâmetros é descrita a seguir no item 6.2.

6.2. Estimação dos parâmetros de pose da câmera

A estimativa da matriz de rotação R e do vetor T que representam a posição relativa entre as câmeras, dependem de uma correspondência inicial de $n \geq 8$ pontos, segundo o algoritmo dos oito pontos apresentados no capítulo 3. As imagens de duas vistas de um gabarito de calibração podem ser capturadas para fornecer a correspondência entre pontos. A figura 6.4, ilustra os pontos numerados de $n = 1, 2, \dots, 10$ considerados para correlação entre duas imagens. Estes pontos podem ser correlacionados via o processo automático proposto por Harris (1998). Mas por

conveniência prática, de modo a permitir diversas simulações distintas, a estimação de pose foi baseada num processo de correlação manual.

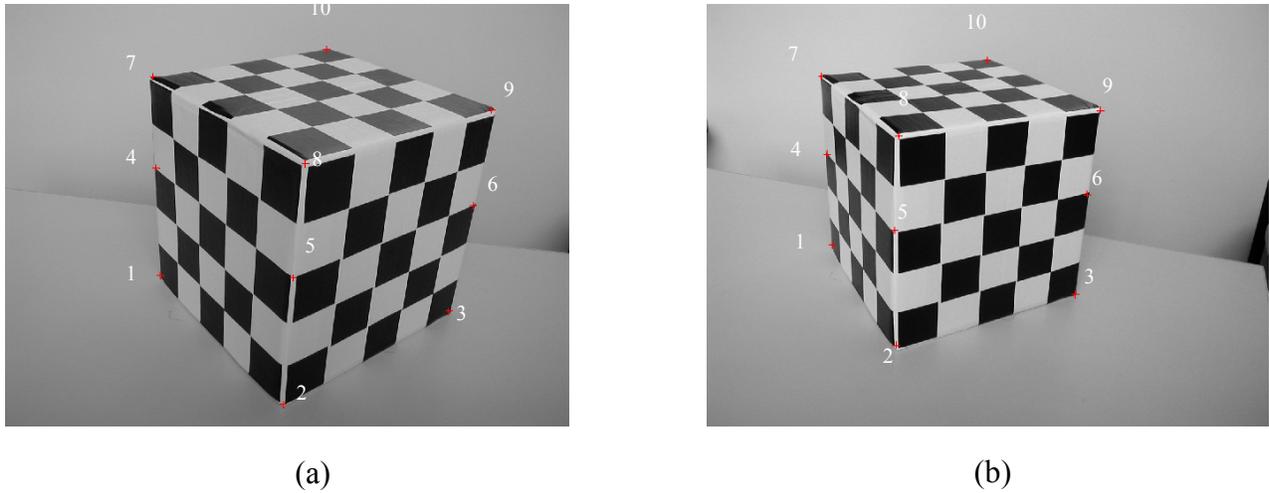


Figura 6.4: Caixa de calibração. a) imagem esquerda. b) imagem direita

Conforme mostra a figura 6.4, os pontos selecionados estão em todas as faces visíveis do cubo, com a respectiva seqüência numérica identificando cada uma das correspondências. Ou seja, o ponto $n=1$ da imagem 6.4-a, corresponde ao ponto $n=1$ da imagem 6.4-b, e assim sucessivamente para todos os demais pontos. A tabela 6.2 apresenta as coordenadas $\mathbf{p}_n=[x,y,z]^T$ dos $n=1,2,\dots,10$ pontos assinalados da caixa de calibração da figura 6.4, para ambas as projeções perspectivas da imagem esquerda e direita. As coordenadas \mathbf{p}_n estão em *pixel* no sistema de coordenadas de imagem.

n	<i>Imagem Esquerda</i>			<i>Imagem Direita</i>		
	X_0	Y_0	Z_0	X_1	Y_1	Z_1
1	177.4256	313.5000	1.0000	142.4074	274.5000	1.0000
2	315.4977	456.5000	1.0000	216.4460	392.5000	1.0000
3	505.5968	351.5000	1.0000	419.5519	331.5000	1.0000
4	172.4230	191.5000	1.0000	137.4047	173.5000	1.0000
5	327.5039	309.5000	1.0000	215.4455	263.5000	1.0000
6	531.6104	229.5000	1.0000	433.5592	217.5000	1.0000
7	164.4188	84.5000	1.0000	130.4011	84.5000	1.0000
8	340.5107	182.5000	1.0000	217.4465	152.5000	1.0000
9	553.6219	121.5000	1.0000	446.5660	20.5000	1.0000
10	366.5243	53.5000	1.0000	319.4997	65.5000	1.0000

Tabela 6.2: Coordenadas (x,y,z) para 10 pontos de projeção da caixa de calibração.

A figura 6.5 apresenta um exemplo de resultado da configuração de pose da câmera, através dos valores de R e T , obtidos da forma descrita na seção 4.3.

$$R = \begin{bmatrix} 0.9775 & -0.0962 & 0.1878 \\ 0.0886 & 0.9949 & 0.0489 \\ -0.1915 & -0.0311 & 0.9810 \end{bmatrix}$$
$$T = \begin{bmatrix} -0.8987 \\ -0.3812 \\ 0.2170 \end{bmatrix}$$

Figura 6.5: Exemplo dos parâmetros de pose estimados para as imagens da figuras 6.4.

6.3. Enquadramento da caixa de contorno por eixos alinhados (AABB)

O passo seguinte, após a obtenção dos parâmetros intrínsecos e a estimação da configuração de pose das câmeras, é a realização do enquadramento do objeto através da construção da caixa de contorno. Este tópico apresenta a construção de uma caixa de contorno alinhada aos eixos da caixa de calibração conforme a definição de um AABB apresentada no capítulo 5. O método descrito a seguir se baseia na utilização de pontos de fuga das imagens da caixa de calibração, para ajustar as arestas do cubo ao objeto.

6.3.1. Determinação dos Pontos de Fuga

Rogers (1976) define os pontos de fuga (*vanishing points*) como as interseções das linhas que formam os vértices do objeto, sobre uma linha que representa o horizonte (ou nível do olho). Esta interpretação pode ser usada para representar a projeção perspectiva no plano de imagem da câmera. A figura 6.6 apresenta a projeção perspectiva de uma caixa e a respectiva posição de um de seus pontos de fuga. A linha do horizonte é formada por pontos infinitamente distantes, ou “*vanishing directions*”, do plano de referência como apresentado por Rogers (1976).

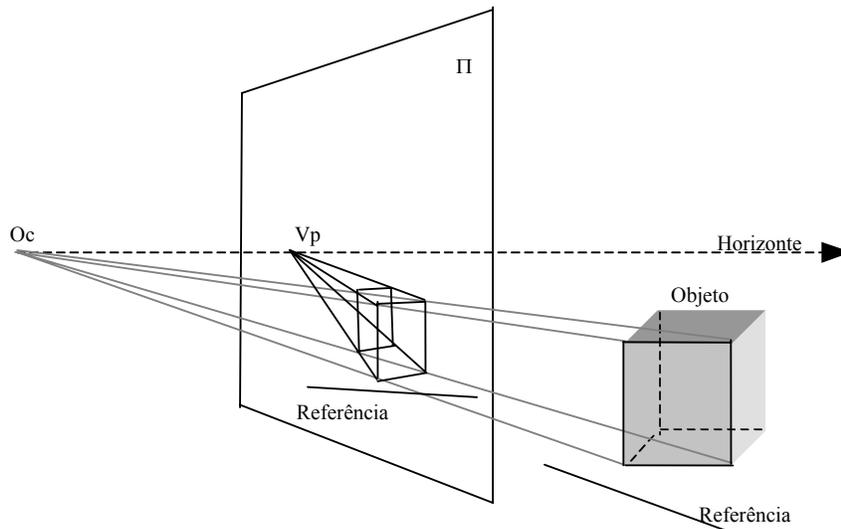


Figura 6.6: Transformação perspectiva do objeto e a representação do ponto de fuga V_p (vanishing point) sobre o plano da imagem. Ilustração adaptada de Rogers (1976).

Como apresentado por Mohr (1996), qualquer par de linhas paralelas parecem se encontrar num ponto no horizonte, que corresponde às suas direções. Todas as interseções no infinito permanecem constantes quando o observador se move. Isto pode ser modelado matematicamente usando coordenadas homogêneas, como apresentado no capítulo 2.

A projeção do objeto vista pela câmera de centro O_c pode assumir as características de projeção dos pontos de fuga, como em Rogers (1976). Como os pontos de fuga de todas as faces do objeto estão sobre o plano da imagem, é possível determiná-los pela interseção das arestas que formam a caixa. O ponto de interseção estará sobre o plano da imagem e possuirá coordenadas no plano $\mathbf{Vp}=[x',y']^T$, onde x' e y' são as coordenadas em *pixels*. Os pontos de fuga podem ocorrer fora da linha do horizonte, e neste caso, estes pontos são denominados por Rogers (1976) como “*trace points*”. No contexto deste trabalho o termo ponto de fuga é usado para ambos os casos.

Para o método proposto, os pontos de fuga são usados com dois propósitos. Primeiro na construção da caixa de contorno inicial, para realizar a determinação dos vértices escondidos da

caixa de referência. E segundo, para permitir o ajuste das arestas da caixa de contorno realizando o enquadramento sobre o objeto de interesse.

A partir da caixa de calibração capturada pelas câmeras, pode-se construir uma caixa virtual, baseada na posição dos vértices. A extração dos vértices de contorno da caixa pode ser feita manualmente pela intervenção do usuário. As câmeras sempre deverão enxergar pelo menos seis arestas que serão necessárias para a identificação dos pontos de fuga.

A figura 6.7 mostra um exemplo de caixa virtual com a identificação dos vértices v e arestas k , obtida a partir da caixa de calibração da figura 6.4.a.

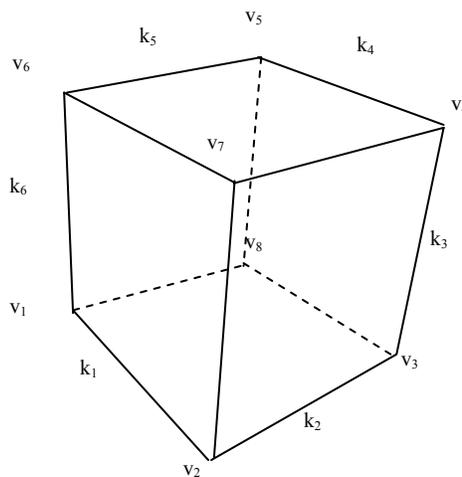


Figura 6.7: Identificação das arestas e vértices da caixa virtual.

A caixa virtual é usada como o modelo de referência inicial do processo de ajuste. É formada por um total de oito vértices chamados v_n , onde $n=1, 2, \dots, 8$. Deste total, os vértices numerados do 1 ao 6 são encontrados diretamente de forma interativa. Os vértices v_7 e v_8 são desconhecidos por não fazerem parte do contorno, mas podem ser calculados baseados nos pontos de fuga das arestas conhecidas.

Em função da posição das arestas, os pontos de fuga F_1 , F_2 e F_3 indicados na figura 6.8, podem ser calculados. A interseção da reta da aresta k_1 com a reta da aresta k_4 , determina o ponto de fuga F_1 . Usando o mesmo princípio, todos F_i com $i=1,2,3$ podem ser encontrados usando as

arestas principais. De forma genérica, os três pontos de fuga podem ser obtidos através da equação da reta para cada uma das arestas. Assim cada aresta pode ser descrita como,

$$a_k x + b_k y + c_k = 0, \quad k=1,2,3,\dots,6 \quad (6.1)$$

E, para encontrar o ponto de fuga F_i , usar as suas arestas principais.

$$F_i(x, y) = \begin{cases} a_i x + b_i y + c_i = 0 \\ a_{(i+3)} x + b_{(i+3)} y + c_{(i+3)} = 0 \end{cases} \quad i = 1,2,3 \quad (6.2)$$

A partir dos pontos de fuga e da interseção com as arestas conhecidas, pode-se determinar a posição dos vértices desconhecidos v_7 e v_8 .

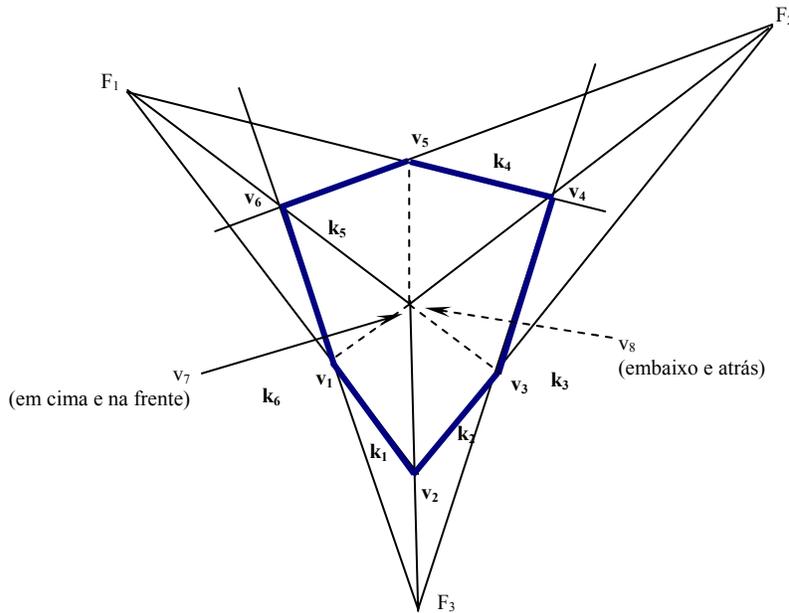


Figura 6.8: Posicionamento dos pontos de fuga F , dos vértices v , e das arestas k . As linhas escuras são as arestas conhecidas. Os pontos de fuga são obtidos a partir destas arestas.

O vértice v_7 pode ser obtido da interseção das retas $F_1 v_6 \cap F_2 v_4$, $F_1 v_6 \cap F_3 v_2$ e $F_2 v_4 \cap F_3 v_2$. Então se calcula o ponto médio dentre estas três possibilidades, que corresponde ao sétimo

vértice da caixa de referência. Procede-se da mesma maneira para o ponto v_8 , através da interseção das retas $F_1v_3 \cap F_2v_1$, $F_1v_3 \cap F_3v_5$ e $F_2v_1 \cap F_3v_5$. Assim, a caixa de referência $C_r = [v_1 v_2 v_3 v_4 v_5 v_6 v_7 v_8]^T$ possui os oito vértices conhecidos.

Como apresenta Chu (2001), a quantidade de pontos de fuga que podem ser localizados depende da posição e direção da câmera. O número de pontos de fuga é igual ao número de grupos de linhas paralelas. Pontos de fuga no infinito podem ocorrer quando o plano de imagem é fisicamente paralelo com uma das faces do cubo. O trabalho de Jelinek (2001) se refere à situação onde somente dois, dos três pontos de fuga são conhecidos. Existe uma solução para o problema da reconstrução usando um procedimento de otimização para encontrar possíveis valores de pontos de fuga que estejam faltando. No caso de não haver informação sobre nenhum dos pontos de fuga a reconstrução se baseia na correspondência entre vértices do modelo e pontos da imagem. Apesar de isto parecer uma restrição, ela pode ser útil na definição da posição da câmera. Se as linhas do ponto de fuga não se interceptam, é porque a posição da câmera não está adequada.

O processo de enquadramento propõe o ajuste do cubo de referência sobre as bordas do objeto usando pontos de fuga, sem reconstruir a superfície do sólido. Após a identificação dos pontos de fuga, procede-se a aproximação das arestas do cubo sobre o objeto da imagem.

6.3.2. Identificação da região de interesse

Segundo Chan (2005), em uma visualização tridimensional, a construção de faces com grande quantidade de pontos (obtidas por nuvens de pontos) resulta em um modelo com um número muito elevado de facetas e vértices, o que torna enorme o esforço computacional para a realização de operações como translação e rotação do objeto, em função de todas as informações que devem ser recalculadas.

Em virtude disto, normalmente pode-se reduzir a quantidade de informações de uma imagem para um conjunto mínimo de atributos, usando um processo de segmentação como

descrito no capítulo 4 e no anexo II. A figura 6.9 apresenta duas vistas de um objeto usado como exemplo para o ajuste de uma caixa de contorno do tipo AABB.



(a)



(b)

Figura 6.9: Imagens de exemplo para determinação da posição e do volume do objeto. a) imagem esquerda. b) imagem direita

Um passo inicial necessário é a identificação do objeto sobre o qual será realizado o enquadramento. Este processo é chamado de identificação da região de interesse (*ROI – Region Of Interest*). Um processo de segmentação da imagem permite a identificação de regiões e componentes de uma cena que serão utilizados no processamento. A obtenção da região de

interesse também permite limitar a quantidade de informações que devem ser analisadas de uma imagem. Para o presente trabalho a região de interesse deve conter o objeto a ser medido, eliminando as informações de fundo da imagem (*background*) a fim de separar o objeto (*foreground*). A figura 6.10 apresenta as duas imagens que serão subtraídas a fim de se obter somente a região de interesse.



(a)



(b)

Figura 6.10: Imagem de teste (a) cena sem objeto a ser medido e (b) com o objeto a ser medido

Pela subtração das duas imagens, consegue-se obter a imagem resultante apresentada na figura 6.11. Além do objeto em evidência na imagem, existem ruídos periféricos originados a partir de diferenças de intensidade de iluminação entre as duas imagens.

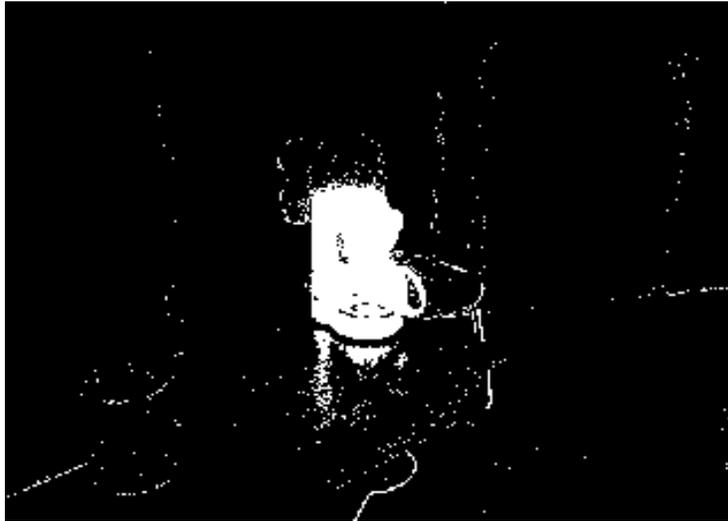


Figura 6.11: Imagem resultante da subtração das imagens 6.10 (a) e (b).

Toda a informação que não corresponde à imagem resultante precisa ser retirada para que o processo de ajuste da caixa de contorno funcione. Para realizar isto, utilizou-se neste caso, um processo morfológico para imagens binárias, chamado erosão binária, para eliminar os pequenos pontos. Qualquer informação da imagem, com dimensões menores do que o elemento estruturante, é eliminado (ver capítulo 4 e anexo II).

Como a propriedade fundamental da erosão faz que objetos da imagem tenham seu tamanho diminuído, isto pode influenciar na obtenção da medição final do objeto. Para evitar este problema, em seguida da erosão é realizado um processo de dilatação binária, usando o mesmo elemento estruturante da erosão. Este procedimento se denomina fechamento binário. Assim consegue-se retornar o objeto para as dimensões originais. O resultado após o fechamento é apresentado na figura 6.12.

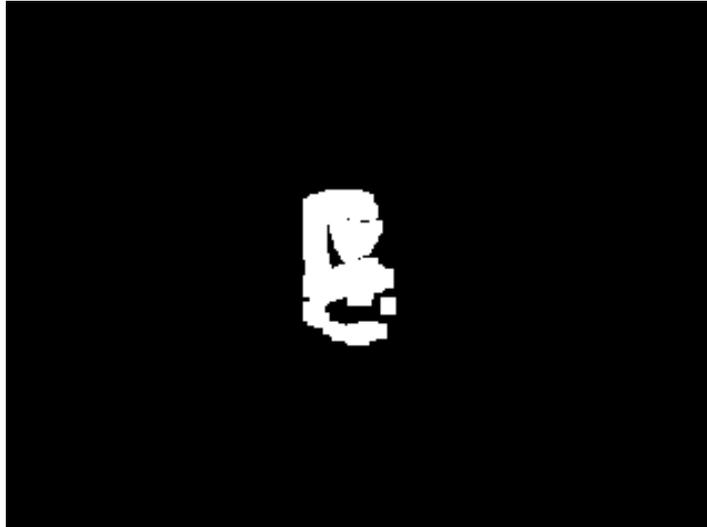


Figura 6.12: Imagem resultante após aplicação de operações de erosão e dilatação binárias.

O processo de ajuste da caixa de contorno ao objeto se baseia nos *pixels* mais externos do objeto, que são os *pixels* referentes aos seu contorno. Então, ainda é necessário extrair a borda do objeto. Um método para a detecção de borda foi apresentado no capítulo 4. Detalhes sobre estes métodos podem ser vistos em Gonzalez (2002) e Facon (1996) entre outros. A imagem resultante após a extração de borda, é dada pela figura 6.13.

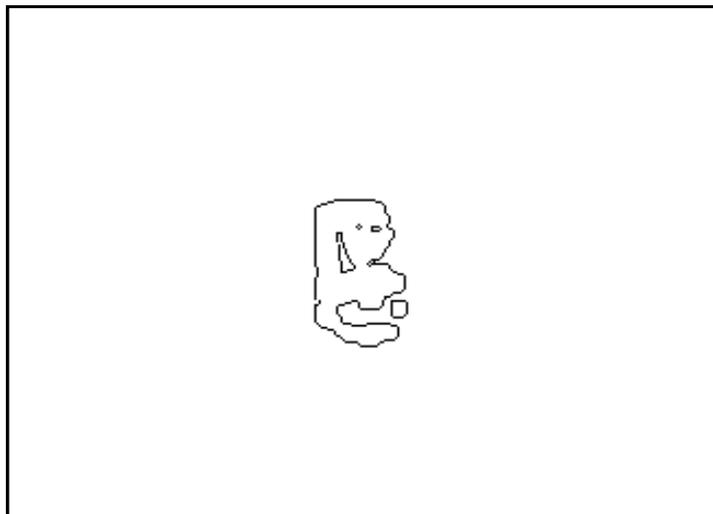


Figura 6.13: Imagem resultante final após a detecção de bordas.

6.3.3. Aproximação das Faces

As coordenadas iniciais das projeções dos vértices $v_n (x'_v, y'_v)$ da caixa de referência já são conhecidas como descrito na seção 6.3.1. As coordenadas $\mathbf{p}' = [x' \ y']^T$ dos *pixels* do contorno do objeto foram determinadas como descrito previamente na seção 6.3.2. A partir destas informações, procede-se a aproximação das faces da caixa de referência. A aproximação ocorre até que a aresta da caixa de referência se encontre com algum elemento da borda do objeto. Ao final do processo de translação das faces da caixa de referência, estará definida a caixa de contorno ajustada aos contornos do objeto.

A figura 6.14 ilustra a seleção de uma das faces da caixa de referência. Os vértices desta face formam a primeira superfície de circunscrição. A face que contém os vértices v_1, v_8, v_5 e v_6 da caixa projetada é inicialmente escolhida para a aproximação.

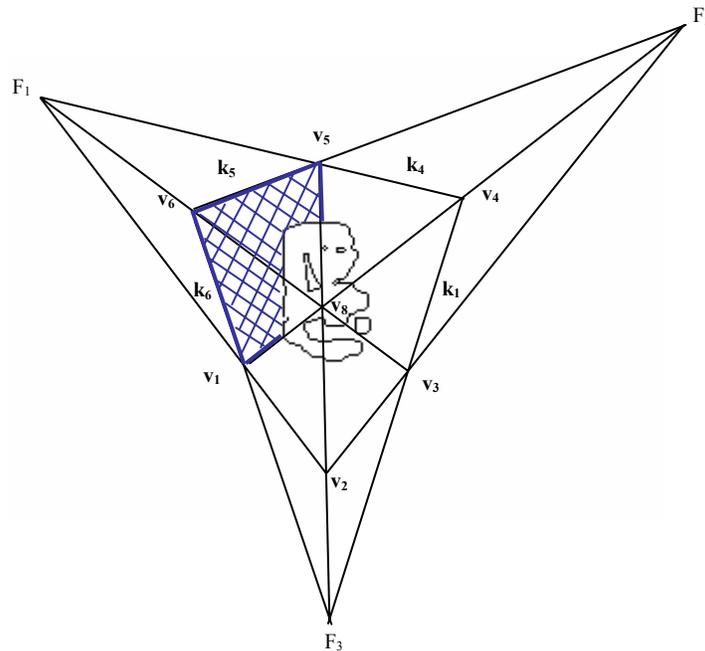


Figura 6.14: Seleção da face do cubo a ser movida.

Os *pixels* do contorno do objeto são utilizados na formação de um segmento de reta entre sua posição na imagem e as coordenadas dos pontos de fuga calculados. A reta que possuir o coeficiente angular mais próximo da aresta principal da caixa de referência será eleita como a nova aresta. Este processo é feito sempre nas direções de dois pontos de fuga. As novas arestas formarão uma nova face que estará tangenciando o objeto.

A figura 6.15 demonstra o processo. A aproximação é conseguida pela rotação das bordas k_5 e k_6 , com pivôs em F_2 e F_3 respectivamente, até a primeira das duas tocar um ponto (*pixel*) dentro da ROI. Nesta situação os novos vértices v'_1 , v'_6 e v'_5 são determinados pela interseção das bordas rotacionadas com a borda original remanescente (não rotacionada). Cada procedimento de aproximação de face é repetido até se alcançar as cinco superfícies de circunscrição da caixa naquela vista projetada.

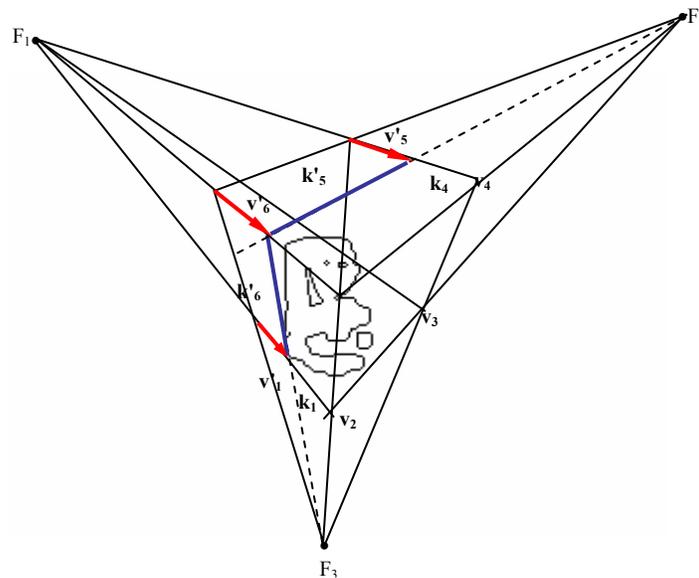
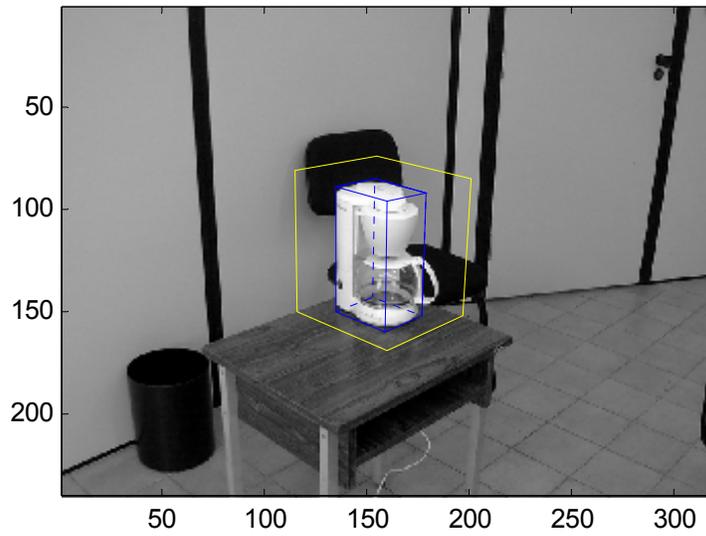
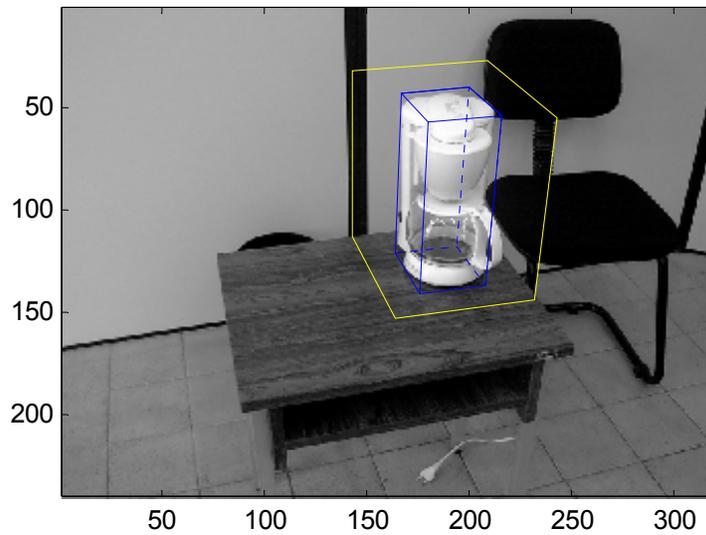


Figura 6.15: Ilustração do processo de aproximação da face às bordas do objeto.

A implementação de um *software* deste procedimento em *MatLab*, gerou os ajustes da caixa de contorno sobre cada uma das imagens do objeto, como apresentado pela figura 6.16. As arestas mais externas ao objeto representam a posição da caixa virtual, obtida a partir das coordenadas do gabarito de calibração. As arestas que formam a caixa interna correspondem ao AABB ajustado ao objeto.



(a)



(b)

Figura 6.16: Exemplo de ajuste da caixa de contorno ao objeto para o par de imagens (a) e (b).

Sempre será possível usar um AABB quando o objeto for simétrico e estiver alinhado a uma *frame* de referência conhecido. Mas para isto é necessário que a caixa de calibração inicial seja maior que o objeto. O ajuste pode não ser correto se o objeto estiver numa posição diferente à da caixa de referência.

6.3.4. Resumo do capítulo

O capítulo apresentou o método para ajuste de uma caixa de contorno sobre as imagens de um objeto. Uma caixa de contorno ajustada sobre os limites de um objeto pode definir a posição espacial e o volume ocupado por ele.

Uma caixa de contorno virtual pode ser obtida a partir dos vértices do gabarito de calibração. Os vértices são reconstruídos espacialmente formando um cubo virtual. O cubo virtual é projetado sobre cada uma das imagens formando a caixa de referência inicial. A partir de uma caixa de referência inicial as arestas podem ser deslocadas de forma a tangenciarem os *pixels* de contorno do objeto. O deslocamento das arestas se baseia na rotação a partir de pontos pivôs. Estes pontos correspondem aos pontos de fuga das arestas.

O capítulo 7 a seguir, irá apresentar de forma detalhada a construção e uma caixa de contorno tipo AABB a partir da caixa de calibração quando os objetos são alinhados com o gabarito, e também fará a demonstração da criação de uma caixa de contorno do tipo OBB a partir da análise dos componentes principais descrita no capítulo 5.

Capítulo 7

Aplicação do Método

O método de reconstrução proposto é baseado na construção de uma caixa de contorno tridimensional sobre a imagem de um objeto. Este capítulo oferece uma visão mais ilustrativa do método desenvolvido no capítulo 6, através da aplicação do método para um conjunto diferenciado de imagens. O texto aborda a criação de um AABB baseado no alinhamento do gabarito de calibração. Também é apresentado um ajuste OBB onde as orientações da posição da caixa de contorno são obtidas diretamente da imagem. A idéia do capítulo é apresentar a eficiência do método para objetos de formas geométricas diferenciadas. Ao final do capítulo são apresentadas as discussões sobre os pontos fortes e fracos do método, bem como são propostas algumas alternativas de solução para os problemas apresentados.

7.1. Ajuste de uma caixa de contorno tipo AABB

Para demonstração do método de ajuste por eixos alinhados, foram adquiridas duas imagens do gabarito de calibração, onde uma delas é ilustrada pela figura 7.1(a) que contém a imagem da esquerda e a outra pela figura 7.1(b) que contém a imagem da direita. Esta figura apresenta em evidência os vértices e as arestas dos gabaritos de calibração que permitirão definir a caixa virtual de contorno. Como apresentado no capítulo 6, esta caixa virtual é usada como ponto de partida para a movimentação das arestas durante o processo de formação da caixa de contorno. A partir das imagens dos gabaritos deve ser realizada a calibração do sistema, que é necessária para a estimação dos parâmetros intrínsecos e para a obtenção dos parâmetros de pose, conforme o método da matriz essencial apresentado no capítulo 3.

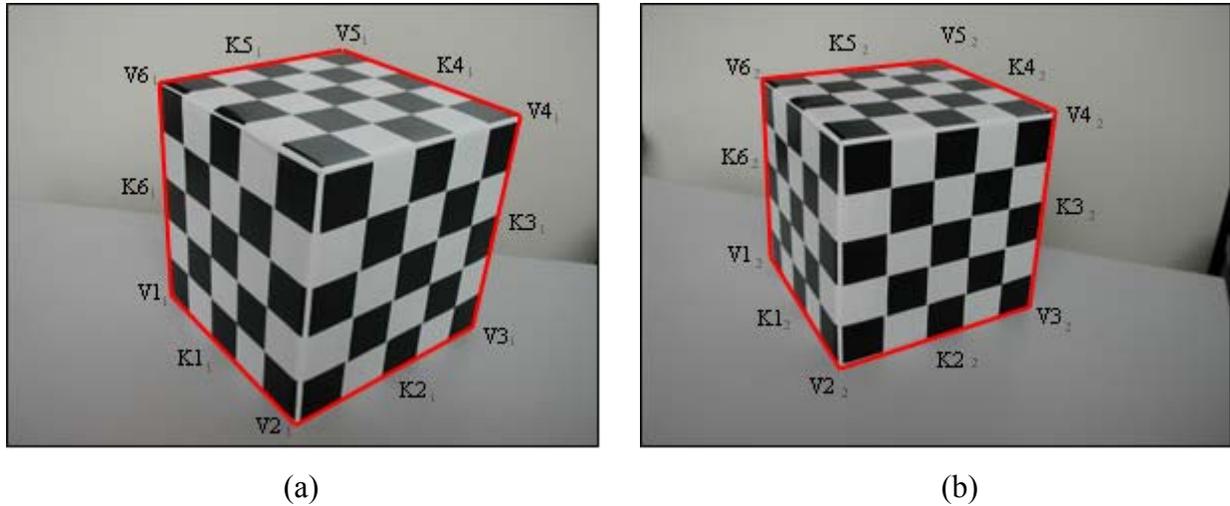


Figura 7.1: Imagens do gabarito de calibração com as arestas e vértices em evidência.

Da figura 7.1 são extraídos os conjuntos de coordenadas dos vértices $v_{i,j} = [x_{i,j} \ y_{i,j} \ z_{i,j}]^T$ onde $i=1, \dots, 6$ é a posição do vértice e $j=1, 2$ é cada uma das imagens. Para as projeções das imagens os valores de $z_{i,j}$ são sempre 1. Cada agrupamento destas coordenadas gera duas matrizes da forma,

$$\mathbf{X}_1 = [v_{1,1} \ v_{2,1} \ v_{3,1} \ v_{4,1} \ v_{5,1} \ v_{6,1}]^T \quad (7.1)$$

$$\mathbf{X}_2 = [v_{1,2} \ v_{2,2} \ v_{3,2} \ v_{4,2} \ v_{5,2} \ v_{6,2}]^T \quad (7.2)$$

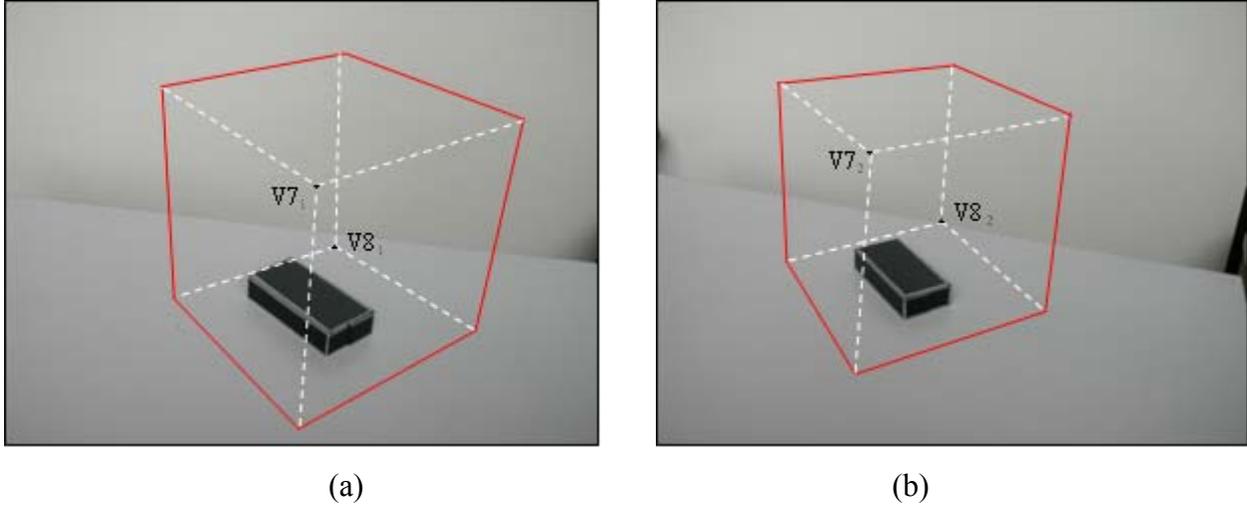
	V _{1,1}	V _{2,1}	V _{3,1}	V _{4,1}	V _{5,1}	V _{6,1}
x ₁	109.4243	197.4979	314.5957	348.6242	227.5230	105.4210
y ₁	192.5000	284.5000	220.5000	79.5000	34.5000	54.5000
z ₁	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Tabela 7.1: Coordenadas dos vértices do gabarito para a imagem 1.

	V _{1,2}	V _{2,2}	V _{3,2}	V _{4,2}	V _{5,2}	V _{6,2}
x ₂	88.4068	136.4469	262.5523	279.5665	198.4987	82.4018
y ₂	173.5000	247.5000	206.5000	75.5000	40.5000	51.5000
z ₂	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Tabela 7.2: Coordenadas dos vértices do gabarito para a imagem 2.

Para realizar a simulação do ajuste foram considerados dois tipos de objetos de características distintas. O primeiro tipo de objeto é apresentado pelas figuras 7.2(a) e 7.2(b), que mostram as imagens das duas projeções de uma caixa metálica na forma de um polígono regular.



(a) (b)
 Figura 7.2: Imagens do objeto e das arestas da caixa virtual.

A figura 7.2 apresenta também a projeção da caixa de contorno virtual sobre cada uma das imagens. As linhas cheias representam as arestas e vértices conhecidos, e as linhas pontilhadas representam as arestas e os vértices $v_{7,i}$ e $v_{8,i}$ que devem ser encontrados. Os parâmetros desconhecidos podem ser obtidos pelo método de ajuste proposto no capítulo 6, que utiliza os pontos de fuga do conjunto de vértices do gabarito para definir a posição das arestas. Os coeficientes angulares das arestas da caixa de contorno servem como ponto de partida para a obtenção dos pontos de fuga da imagem. Um vetor de coeficientes C_i para cada imagem $i=1,2$ pode ser formado como:

$$C_1 = [c_{1,1} \quad c_{2,1} \quad c_{3,1} \quad c_{4,1} \quad c_{5,1} \quad c_{6,1}]^T \quad (7.3)$$

$$C_2 = [c_{1,2} \quad c_{2,2} \quad c_{3,2} \quad c_{4,2} \quad c_{5,2} \quad c_{6,2}]^T \quad (7.4)$$

Onde cada elemento de C_i é dado por:

$$c_{k,i} = \frac{Yv_{(k+1,i)} - Yv_{k,i}}{Xv_{(k+1,i)} - Xv_{k,i}} \quad (7.5)$$

Na expressão 7.5, a coordenada x de cada vértice $v_{k,i}$ é denominada por Xv e a coordenada y de cada vértice é denominada Yv para cada aresta $k = 1...5$. Assim, os coeficientes angulares de cada aresta são obtidos em seqüência da primeira à penúltima aresta. Para fechar o polígono, ângulo $c_{6,i}$ da última aresta depende exclusivamente do valor do primeiro vértice $v_{1,i}$ e do último $v_{6,i}$. Neste caso particular a expressão 7.5 pode ser modificada para:

$$c_{6,i} = \frac{Yv_{6,i} - Yv_{1,i}}{Xv_{6,i} - Xv_{1,i}} \quad (7.6)$$

A partir dos ângulos conhecidos de cada aresta, podem ser encontrados os pontos de fuga que formam as linhas de contorno da caixa virtual. Os pontos de fuga F_i para cada imagem $i=1,2$, podem ser obtidos a partir das arestas que formam cada um dos vértices. O método foi apresentado no capítulo 6, e se baseia na interseção das arestas de acordo com a expressão:

$$\begin{bmatrix} a_k & b_k \\ a_{(k+3)} & b_{(k+3)} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} c_k \\ c_{(k+3)} \end{bmatrix} \quad (7.7)$$

Para $k=1,2,3$ encontra-se os pontos de fuga pela solução do sistema (7.7). São formados três pontos de fuga para cada imagem.

$$F_i = [F_{1,i} \quad F_{2,i} \quad F_{3,i}]^T \quad (7.8)$$

Os vértices $v_{7,i}$ e $v_{8,i}$ podem ser localizados a partir dos pontos de fuga. A figura 7.2 mostra a posição destes vértices, na interseção das linhas pontilhadas. Os vértices v_7 e v_8 de cada imagem são obtidos encontrando-se os coeficientes angulares da forma:

$$C_7 = [c_{7,1} \quad c_{7,2} \quad c_{7,3}]^T \quad (7.9)$$

$$C_8 = [c_{8,1} \quad c_{8,2} \quad c_{8,3}]^T$$

Onde C_7 é o vetor dos coeficientes angulares de cada um dos vértices v_{2i} , v_{4i} e v_{6i} em relação aos pontos de fuga F_{1i} , F_{2i} e F_{3i} , e C_8 é o vetor dos coeficientes angulares que relacionam as coordenadas dos vértices v_{1i} , v_{3i} e v_{5i} com os pontos de fuga F_{1i} , F_{2i} e F_{3i} , e são calculados pelas expressões:

$$c_{7,1} = \frac{y_{F_1} - y_{v_2}}{x_{F_1} - x_{v_2}} \quad c_{7,2} = \frac{y_{F_3} - y_{v_4}}{x_{F_3} - x_{v_4}} \quad c_{7,3} = \frac{y_{F_2} - y_{v_6}}{x_{F_2} - x_{v_6}} \quad (7.10)$$

$$c_{8,1} = \frac{y_{F_3} - y_{v_1}}{x_{F_3} - x_{v_1}} \quad c_{8,2} = \frac{y_{F_2} - y_{v_3}}{x_{F_2} - x_{v_3}} \quad c_{8,3} = \frac{y_{F_1} - y_{v_5}}{x_{F_1} - x_{v_5}}$$

Na expressão 7.10 (x_F, y_F) são as coordenadas do ponto de fuga e (x_v, y_v) são as coordenadas dos vértices. As interseções das retas formadas pelos coeficientes C_7 geram três soluções possíveis, dadas como:

$$v_7 = \text{média}(v_{7a}, v_{7b}, v_{7c}) \quad (7.11)$$

Onde,

$$v_{7a} = \begin{bmatrix} -x_{c_1} & 1 \\ -x_{c_2} & 1 \end{bmatrix}^{-1} \begin{bmatrix} y_{c_1} \\ y_{c_2} \end{bmatrix}$$

$$v_{7b} = \begin{bmatrix} -x_{c_1} & 1 \\ -x_{c_3} & 1 \end{bmatrix}^{-1} \begin{bmatrix} y_{c_1} \\ y_{c_3} \end{bmatrix}$$

$$v_{7c} = \begin{bmatrix} -x_{c_2} & 1 \\ -x_{c_3} & 1 \end{bmatrix}^{-1} \begin{bmatrix} y_{c_2} \\ y_{c_3} \end{bmatrix}$$

(7.12)

Também são geradas três soluções possíveis para a localização do oitavo vértice. O procedimento para o cálculo do v_8 é o mesmo usado na obtenção do vértice v_7 . O valor médio dos três pontos de interseção define as coordenadas para este vértice.

$$v_8 = \text{média}(v_{8a}, v_{8b}, v_{8c}) \quad (7.13)$$

Os valores de v_{8a} , v_{8b} e v_{8c} são obtidos da mesma forma apresentada pela expressão 7.12, usando os coeficientes angulares a partir de C_8 (expressão 7.9). Após a obtenção dos vértices v_7 e v_8 para cada imagem, procede-se o ajuste das arestas da caixa virtual sobre o contorno do objeto.

Os segmentos de linha que formam as arestas da caixa virtual devem ser posicionados de forma a tangenciar os *pixels* da borda do objeto. Os pontos de fuga F_i servem como pivôs para a rotação das arestas. Duas matrizes p_1 e p_2 armazenam as coordenadas dos n pontos de contorno de ambas as imagens.

$$p_1 = \begin{bmatrix} x_{1,1} & x_{2,1} & x_{3,1} & \cdots & x_{n,1} \\ y_{1,1} & y_{2,1} & y_{3,1} & \cdots & y_{n,1} \end{bmatrix}^T \quad (7.14)$$

$$p_2 = \begin{bmatrix} x_{1,2} & x_{2,2} & x_{3,2} & \cdots & x_{n,2} \\ y_{1,2} & y_{2,2} & y_{3,2} & \cdots & y_{n,2} \end{bmatrix}^T \quad (7.15)$$

Cada ponto $p=(x,y)$ do contorno forma uma reta r_q , com $q=1,\dots,n$, até cada um dos pontos de fuga F_i . O processo apresentado no capítulo 6, consiste em encontrar a reta r_q que possui a menor distância de cada um dos vértices v_{ki} , com $k=1,\dots,6$ em relação aos pontos de contorno p_i , para cada imagem $i=1,2$. Cada reta r_q tem a seguinte configuração:

$$r_q = [a_q \quad b_q]^T \quad (7.16)$$

O parâmetro a_q da equação 7.16 corresponde ao coeficiente angular da reta r_q . O parâmetro b_q corresponde ao coeficiente linear.

O algoritmo do processo de posicionamento das arestas é descrito a seguir. Para a seqüência dos vértices v_k , $k=1...6$, executar:

- i. Calcular a equação de todas as retas r_q formadas entre os *pixels* do contorno e os pontos de fuga F_i ;
- ii. Encontrar os coeficientes angulares entre cada vértice v_k e os pontos de fuga F_i ;
- iii. Calcular a diferença angular entre as retas r_q e as retas do ponto de fuga; O menor ângulo corresponde a reta tangente mais próxima de v_k .
- iv. Deslocar o vértice v_k para o ponto de interseção e definir as novas arestas baseadas neste vértice.

A figura 7.3 apresenta o resultado final do processo de ajuste das arestas. O resultado do enquadramento proporciona um ajuste das arestas bem próximo ao contorno do objeto. Pode-se notar pelas imagens da esquerda e da direita, das respectivas projeções do objeto, que eles estão posicionados com a mesma orientação do gabarito de calibração. Isto é essencial para um ajuste do tipo AABB.

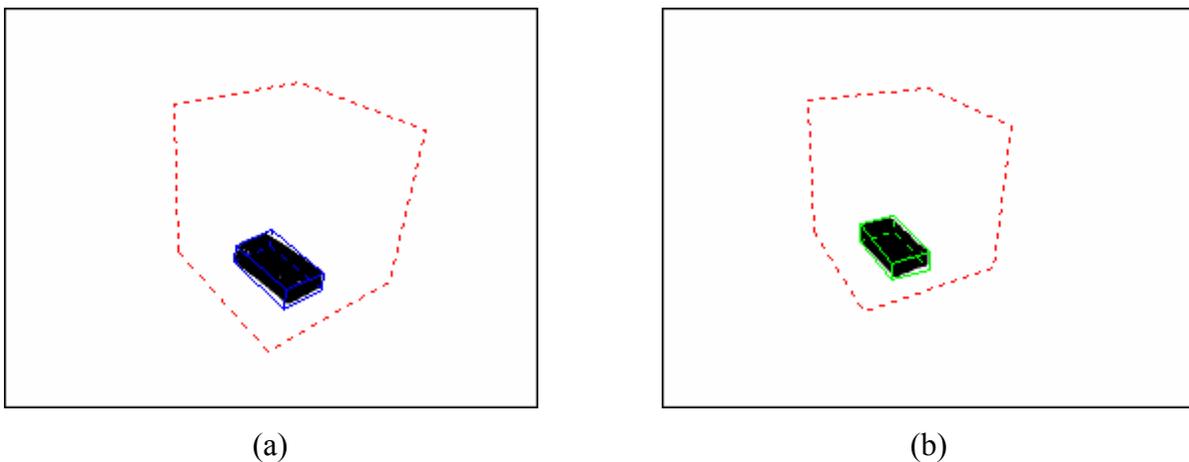


Figura 7.3: Ajuste do AABB sobre o objeto caixa metálica. (a) imagem da esquerda. (b) imagem da direita.

O segundo tipo de objeto utilizado para exemplificar o método é apresentado pelas figuras 7.4(a) e 7.4(b), que mostram as imagens de um perfurador de papéis, com simetria em apenas uma das direções, contendo saliências e elementos vazados.

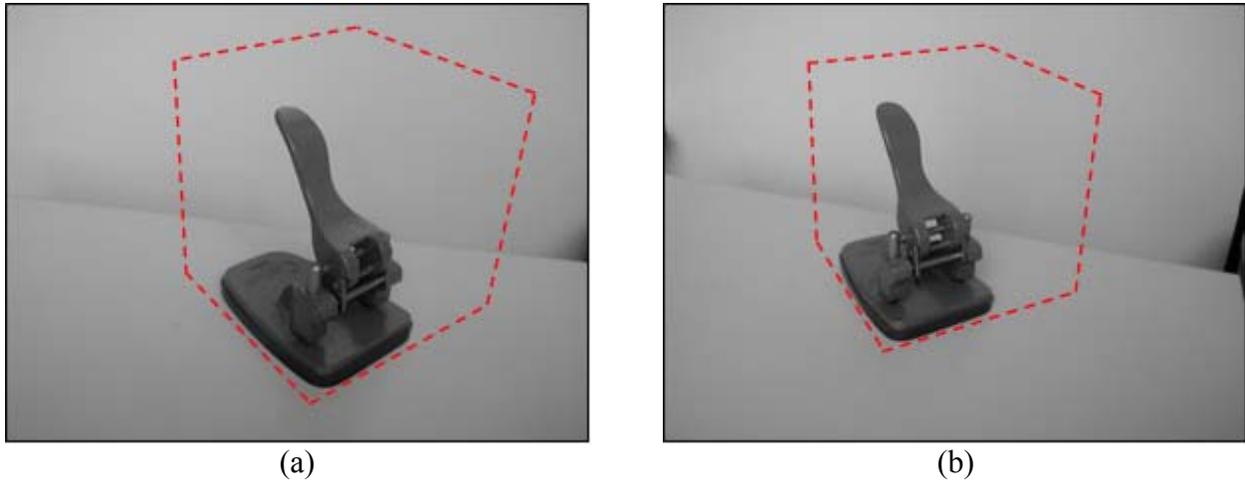


Figura 7.4: Par de imagens do furador de papel.

O objeto é inicialmente alinhado sob a mesma posição do gabarito de calibração. Procedendo-se o método de enquadramento apresentado no capítulo 6, obtém-se como resultado as imagens da figura 7.5.

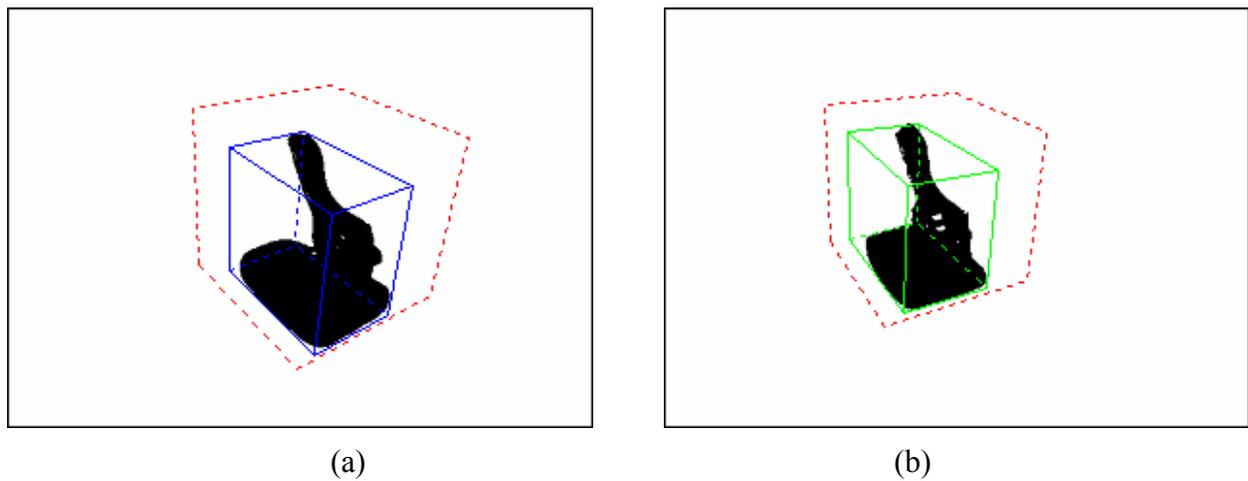


Figura 7.5: Ajuste do AABB sobre o objeto perfurador de papel. (a) imagem da esquerda. (b) imagem da direita.

Conforme apresentado pela figura 7.4 e pela figura 7.5, o ajuste de arestas através de AABB permite um bom enquadramento do objeto em função do alinhamento da caixa de calibração. A figura 7.6 apresenta o problema de orientação, quando o objeto não está alinhado com o gabarito. Para o ajuste da caixa de contorno a condição inicial diz respeito à necessidade

do objeto estar posicionado dentro da região definida pelo gabarito. Entretanto o objeto pode possuir uma orientação aleatória, diferente da esperada.

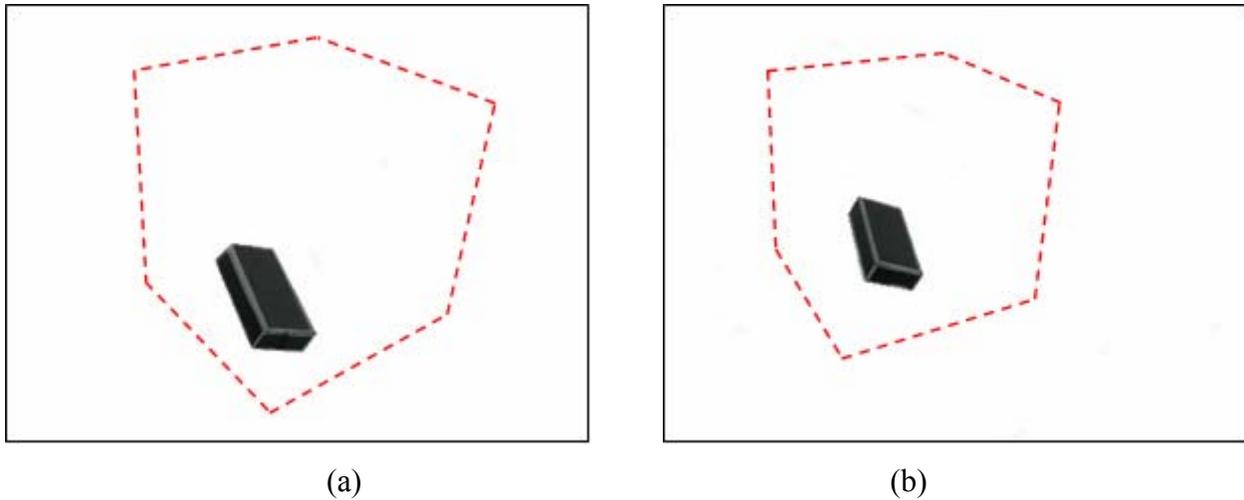


Figura 7.6: Par de imagens desalinhadas em relação ao gabarito.

A construção da caixa virtual segue o mesmo processo apresentado anteriormente. A figura 7.7 apresenta a caixa virtual criada a partir do gabarito de calibração. É facilmente observado que os eixos de simetria do objeto são diferentes da caixa de contorno inicial.

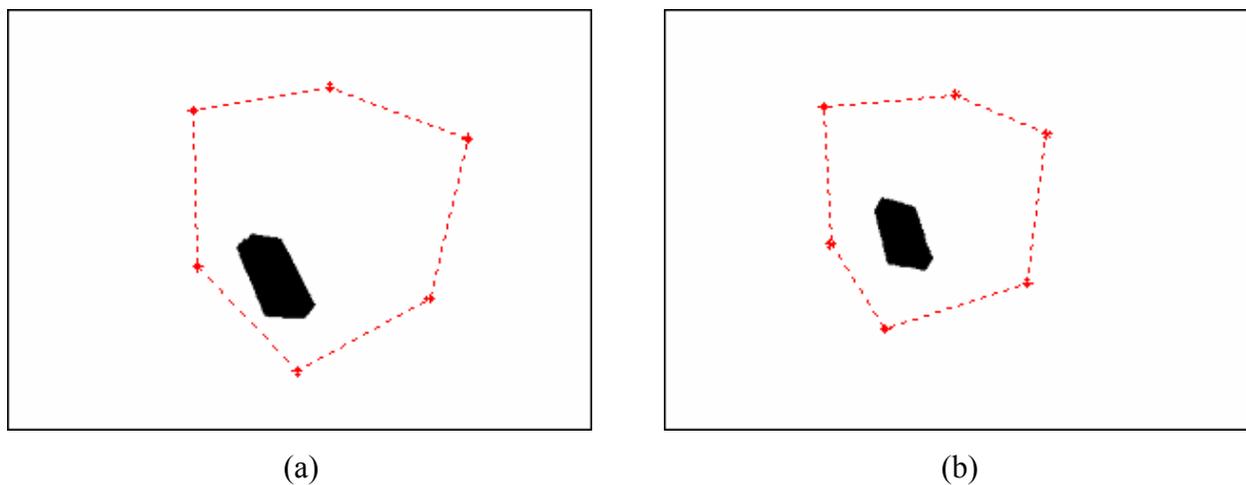


Figura 7.7: Objeto a ser enquadrado sem alinhamento com o gabarito.
(a) imagem da esquerda. (b) imagem da direita.

Ao realizar o ajuste, uma caixa de contorno é gerada ocupando um volume maior do que o esperado para o objeto. A figura 7.8 apresenta o resultado.

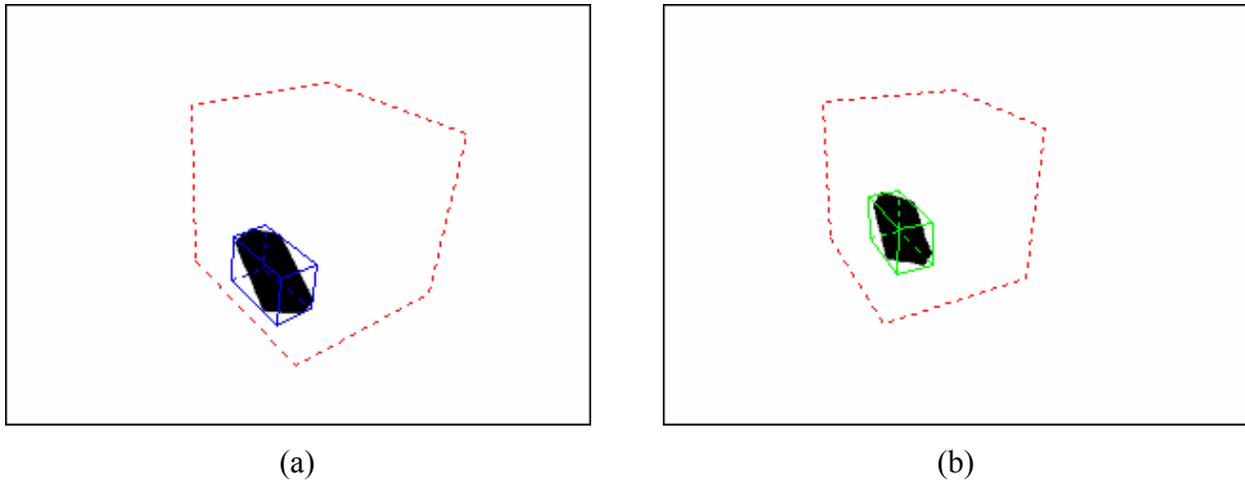


Figura 7.8: Objeto a ser enquadrado. (a) imagem da esquerda. (b) imagem da direita.

Conforme ilustrado pela figura 7.8, o ajuste do AABB não gera um bom enquadramento pelo fato de que os eixos principais do objeto, não estarem alinhados com os eixos principais do gabarito. Assim, a posição e o volume calculado não correspondem ao objeto da figura. A diferença de resultado decorrente de um bom ajuste em relação a um pior ajuste, pode determinar o tipo de estratégia a ser adotada durante um processo automático de reconhecimento. Duas conclusões podem ser obtidas desta situação. Primeiro, se o volume calculado não corresponde ao esperado, pode significar que existe um objeto diferente para ser processado. Uma planta automatizada que contenha este sistema poderá interagir com um banco de dados e reconhecer automaticamente a mudança de objeto e dar seqüência ao processo. A segunda conclusão é a de que o objeto pode estar fora da posição esperada. O sistema pode identificar esta ocorrência e interromper o processo a fim de evitar uma operação indesejada. Objetos que são manipulados por braços robóticos, podem frequentemente apresentar posições aleatórias e a identificação de posição por uma AABB pode determinar a decisão sobre a continuidade do processo. A abrangência de processos que podem utilizar o método proposto de ajuste de uma caixa de contorno pode ser aumentada, pela re-orientação dos eixos da caixa de referência. O item 7.2, a seguir, apresenta uma forma de ajuste ao objeto através de uma caixa de contorno orientada (OBB).

7.2. Ajuste de uma caixa de contorno tipo OBB

A caixa de calibração inicial possui orientação bem definida e sua reconstrução 3D é baseada no emparelhamento de pontos entre as arestas e vértices das duas vistas, como apresentado no item 6.2. A caixa virtual criada a partir desta reconstrução pode ser reorientada espacialmente, de forma que suas arestas coincidam com as direções principais do objeto. Esta nova orientação gera então uma nova caixa de contorno inicial, agora do tipo OBB, que pode ser ajustada pelo mesmo processo apresentado na seção 6.3. O item 7.2.1 apresenta o procedimento de reorientação para a criação do OBB.

7.2.1. Ajuste do OBB pelas rotações do gabarito

Para posicionar a caixa de contorno, inicialmente é necessário definir uma orientação para o objeto e usar depois esta informação para re-posicionar espacialmente a caixa de contorno inicial. As direções principais planares do objeto, podem ser obtidas através da análise dos componentes principais como apresentado no capítulo 5. A figura 7.9 apresenta as duas vistas de um objeto que ocupa o mesmo espaço do gabarito de calibração, porém seus eixos principais não estão alinhados com os eixos do gabarito

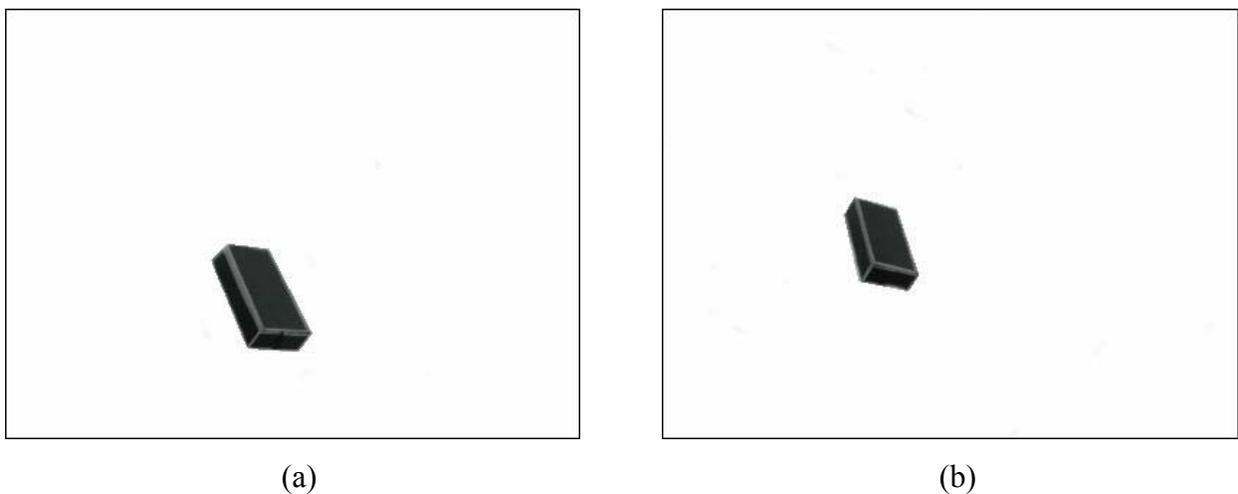
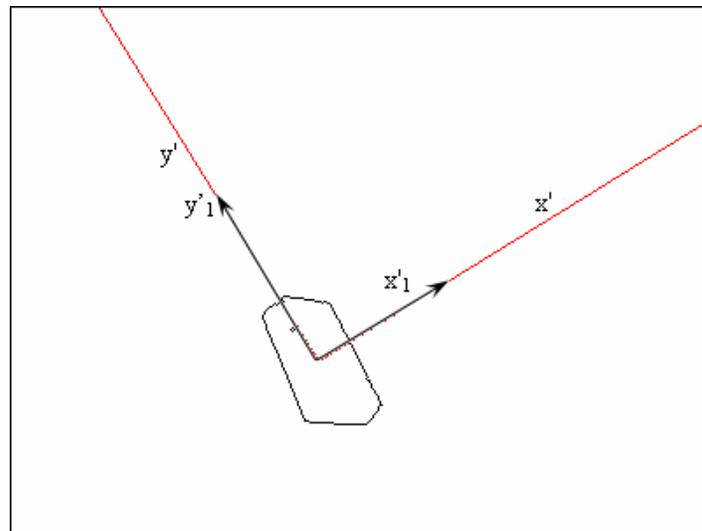
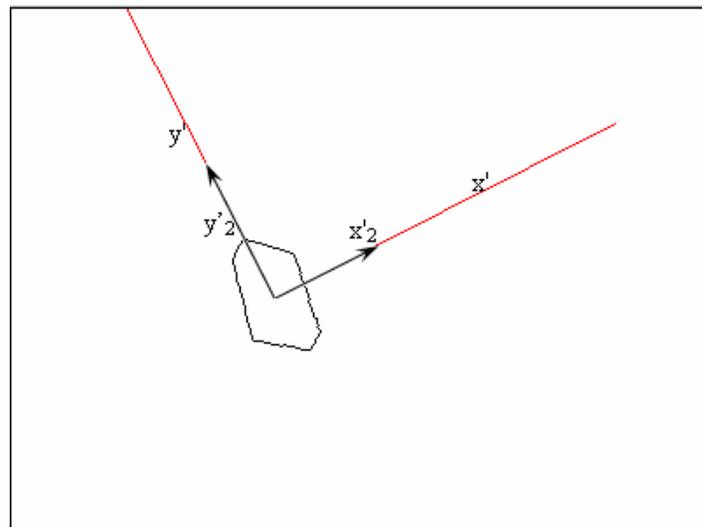


Figura 7.9: Objeto com orientação diferente das direções principais do gabarito de calibração. (a) imagem da esquerda. (b) imagem da direita.

Para determinar a sua posição no plano, é necessário se obter as coordenadas dos *pixels* de borda. O contorno do objeto é extraído conforme o método apresentado no capítulo 4. A figura 7.10 apresenta a composição do sistema de eixos $x'y'$, representando as direções do objeto para cada uma das imagens.



(a)



(b)

Figura 7.10: Eixos principais para o par de imagens do objeto.

Com as coordenadas do contorno pode ser criada uma matriz de pontos $X_i=[p_1, p_2, \dots p_n]^T$, onde $\mathbf{p}_k = [x_k \ y_k]^T$ e $k=1\dots n$ é o número total de *pixels* da borda do objeto. Como descrito no capítulo 5, a partir da matriz X pode ser calculada uma matriz de covariância. A decomposição em valores singulares da matriz de covariância gera os autovalores e os respectivos autovetores que formam o sistema de eixos. A partir da estimação destas direções, podem ser formados para cada imagem, dois vetores colineares \mathbf{x}' e \mathbf{y}' em cada direção, de comprimento igual ao ponto médio de cada distância.

O processo de reconstrução da caixa virtual a partir das imagens do gabarito pode ser feito da maneira apresentada na seção 6.3. Os eixos recém calculados podem ser reconstruídos da mesma forma. Os eixos reconstruídos X' e Y' definem um plano no espaço. A partir dos eixos X' e Y' conhecidos no plano α , é possível encontrar o eixo Z' comum às duas imagens. O produto vetorial entre dois vetores \mathbf{u} e \mathbf{v} não paralelos, descrito por Venturi (88), gera um terceiro vetor perpendicular aos vetores dados, apresentado na figura 7.11.

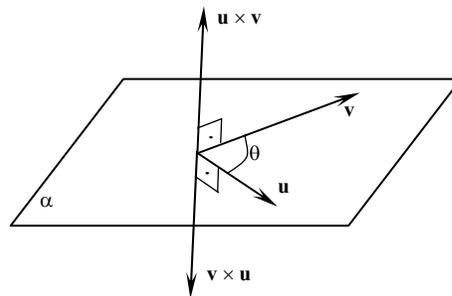


Figura 7.11: Produto vetorial entre dois vetores \mathbf{u} e \mathbf{v} .

A figura 7.12 apresenta a reconstrução espacial dos eixos orientados do objeto X' e Y' , bem como o eixo ortogonal Z' . Pela característica construtiva do objeto, os eixos devem ser mutuamente ortogonais. O produto vetorial entre o novo vetor gerado Z' e o eixo X' gera um novo vetor Y' , forçando a ortogonalidade do sistema.

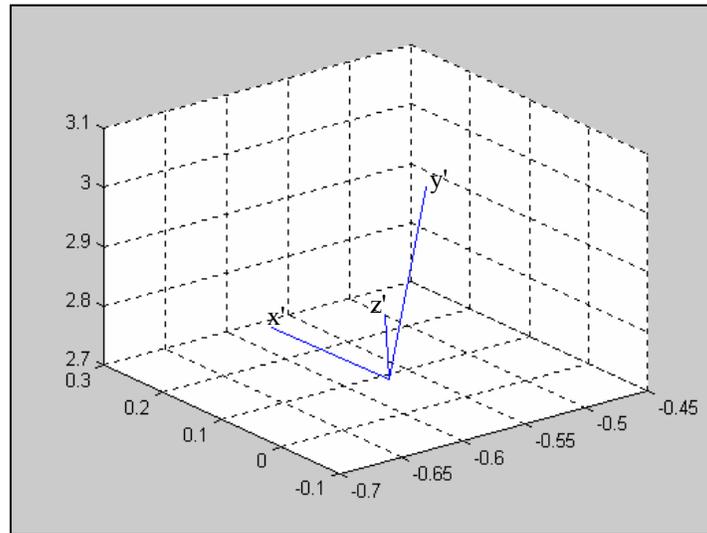


Figura 7.12: Representação 3D dos eixos de orientação do objeto.

A figura 7.13 apresenta as projeções dos eixos em cada plano de imagem. Os vetores $v_{k,i}$ definem a orientação do objeto. Onde $k = 1,2,3$ representa o eixo, e $i=1,2$ identifica a imagem.

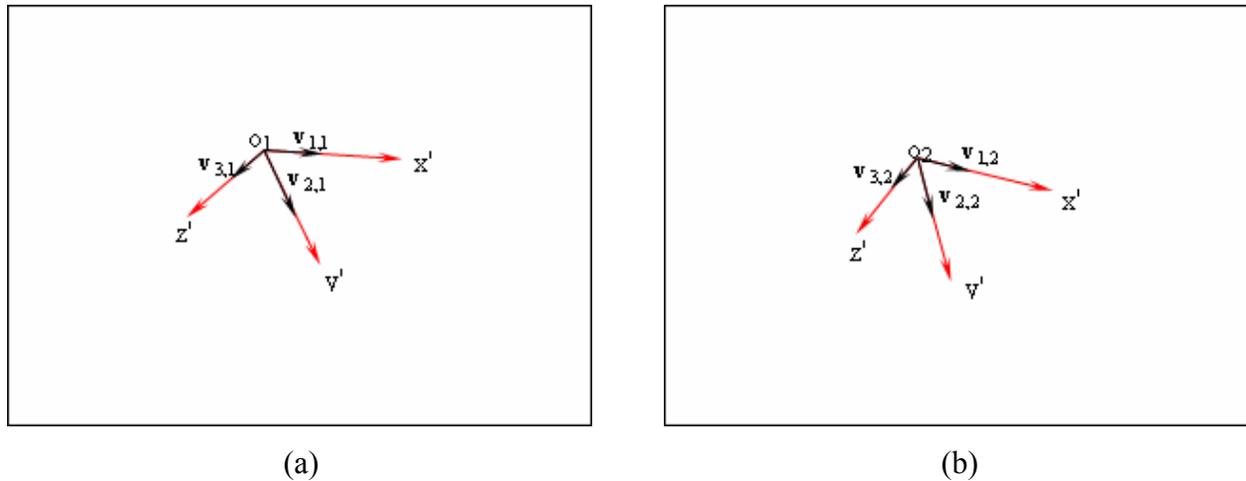
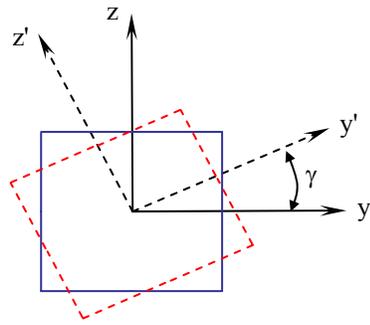


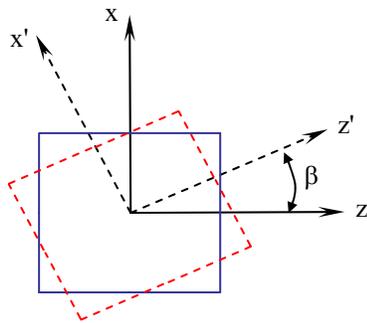
Figura 7.13: Definição dos vetores $v_{k,i}$ em cada uma das direções.

Os vetores $v_{k,i}$ descrevem a posição de um mesmo ponto no espaço. Baseado na orientação destes eixos é possível re-posicionar a caixa virtual pela rotação dos seus eixos principais. Cada um dos eixos descreve um ângulo no plano de projeção, como mostra a figura 7.14.



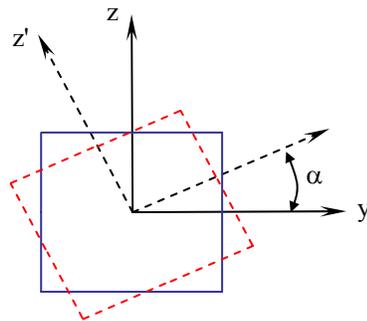
$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \gamma & \text{sen} \gamma \\ 0 & -\text{sen} \gamma & \cos \gamma \end{bmatrix}$$

(a)



$$\begin{bmatrix} \cos \beta & 0 & -\text{sen} \beta \\ 0 & 1 & 0 \\ \text{sen} \beta & 0 & \cos \beta \end{bmatrix}$$

(b)

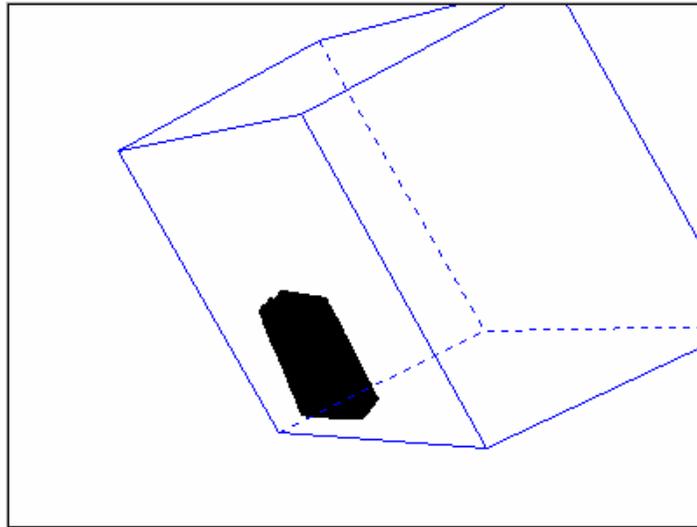


$$\begin{bmatrix} \cos \alpha & \text{sen} \alpha & 0 \\ -\text{sen} \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

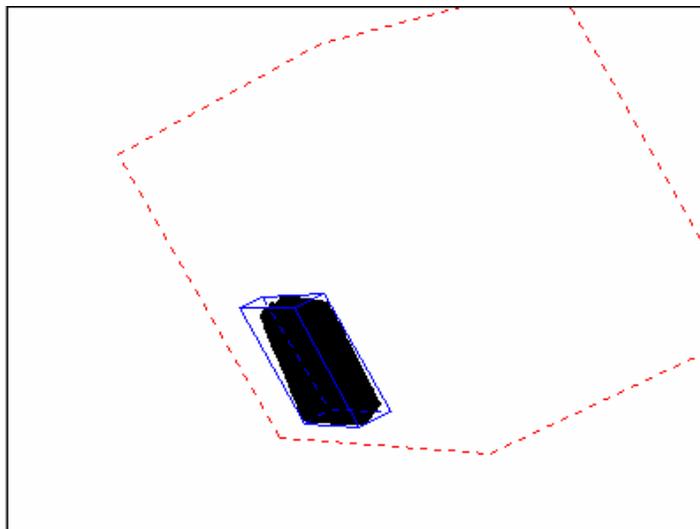
(c)

Figura 7.14: (a) rotação sobre o eixo x. (b) rotação sobre o eixo y. (c) rotação sobre o eixo z.

A partir da projeção dos eixos sobre os planos de imagem é possível encontrar os ângulos que permitirão a rotação do gabarito de calibração. As arestas da caixa virtual podem ser reorientadas através das matrizes de rotação da figura 7.14. A obtenção destas matrizes pode ser vista em Rogers (1976). Uma matriz de rotação pode ser definida pela posição dos eixos e nova orientação do gabarito pode ser definida como mostra a figura 7.15.



(a)



(b)

Figura 7.15: Rotação da caixa de calibração conforme orientação do objeto e ajuste do AABB sobre o objeto.

Conforme ilustra a figura 7.15, o ajuste das arestas sobre o objeto foi obtido pelo método apresentado no item 6.3. A solução desta forma é dependente da utilização de um referencial conhecido, como por exemplo, o gabarito de calibração. O método descrito neste item pode ser utilizado quando for possível usar um gabarito de calibração. A utilização do gabarito de calibração pode não ser prático se ocorrerem muitas variações no ambiente, de forma que o

gabarito deva ser re-posicionado e os parâmetros da câmera recalculados. Para que o método seja mais funcional para aplicações práticas, um OBB deve ser criado diretamente a partir das imagens do objeto. A caixa de referência pode ser obtida diretamente da imagem, considerando as direções principais do objeto. O item 7.2.2 apresenta um método de geração de uma caixa de contorno que pode ser criada automaticamente, sem utilizar a caixa de calibração inicial..

7.2.2. Enquadramento de OBB pelas direções principais

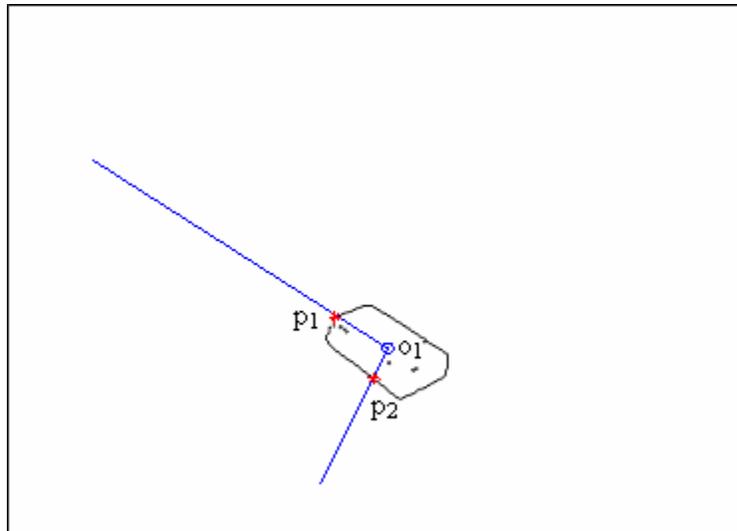
Este tópico apresenta a construção de uma caixa de contorno orientada pelos eixos principais do objeto, conforme a definição de um OBB apresentada no capítulo 5. Como descrito anteriormente os eixos principais podem ser obtidos pela técnica da análise dos componentes principais (PCA) ou pelos momentos de inércia do agrupamento de pontos. A figura 7.16 apresenta duas imagens de um polígono regular, sobre a qual se deseja construir uma caixa de contorno.



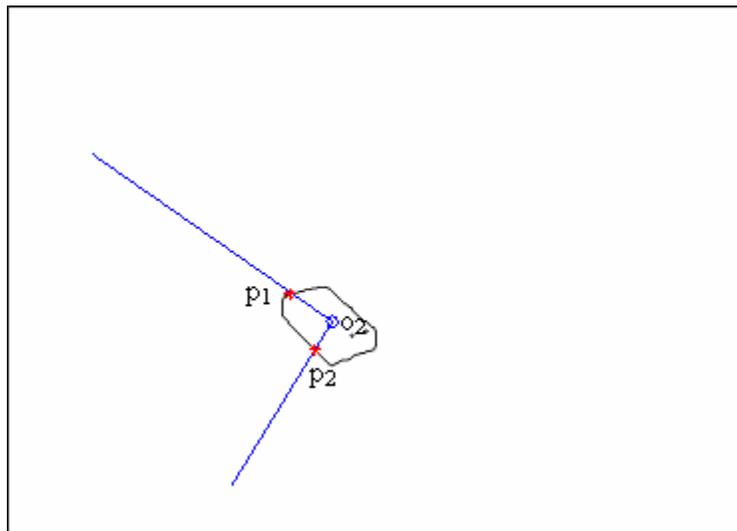
Figura 7.16: Par de imagens para o exemplo do ajuste.

A imagem da figura 7.16 deve ser segmentada de forma a se extrair o seu contorno. Os métodos de segmentação que permitem identificar a região de interesse, bem como as ferramentas para detecção do contorno foram apresentados no capítulo 4. O contorno da figura define o conjunto de pontos usados para a obtenção dos eixos principais. Os pontos de interseção

dos eixos com os elementos de contorno do objeto formam um conjunto de segmentos de reta como apresenta a figura 7.17.



(a)



(b)

Figura 7.17: Direções principais do objeto para as duas vistas da imagem.

Como mostra a figura 7.17, as interseções dos eixos principais com os *pixels* de borda formam dois segmentos de reta bem definidos. Os pontos p_1 e p_2 , marcados na figura definem um conjunto $\mathbf{X}_i = [p1_i \ o_i]$ e um $\mathbf{Y}_i = [p2_i \ o_i]$ para cada imagem i , onde $i=1,2$. A figura 7.18 ilustra

o processo, onde os *pixels* de borda foram propositalmente espaçados para uma melhor visualização.

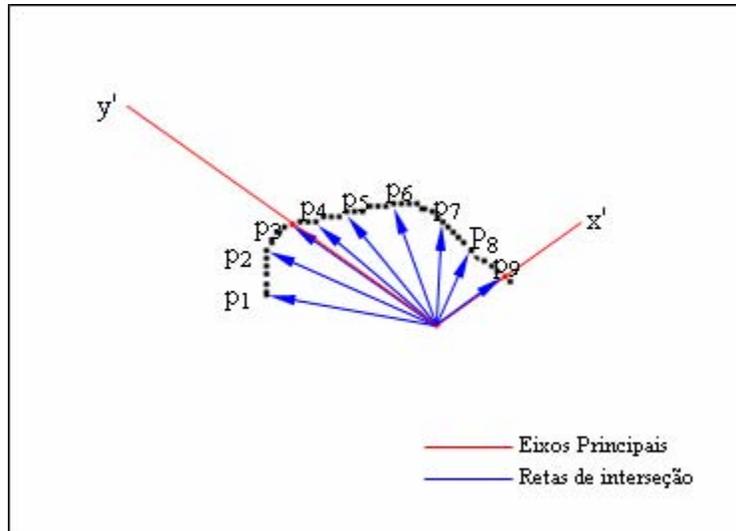


Figura 7.18: Obtenção dos pontos de interseção.

O cálculo das interseções pode ser dado pela comparação dos coeficientes angulares. A reta formada entre um ponto do contorno e o seu centro de massa o_i possui um coeficiente angular m_k , com $k=1,2,\dots,n$, onde n é a quantidade de *pixels* do contorno. As retas que formam o eixo de simetria possuem coeficientes angulares r_j com $j=1,2$. Os coeficientes m_i das retas podem ser comparados com os coeficientes r_j dos eixos gerando um vetor de diferenças. O valor da diferença mínima define qual reta m_i possui a mesma inclinação do eixo r_j . Assim os pontos p_{1k} e p_{2k} do contorno podem ser marcados, delimitando o comprimento de cada um dos eixos.

A reconstrução espacial dos eixos coplanares formados por $\mathbf{X}_i = [p1_i \ o_i]$ e $\mathbf{Y}_i = [p2_i \ o_i]$ para a primeira e segunda imagem não ocupam o mesmo lugar geométrico no espaço. Sendo assim, os pontos p_{1k} , p_{2k} , o_1 e o_2 não correspondem a um mesmo ponto no espaço. Em função disto, o procedimento adotado trata da obtenção de pontos médios entre as projeções das duas imagens. Um conjunto de pontos \mathbf{X}'_i e \mathbf{Y}'_i médios reconstruídos a partir de \mathbf{X}_i e \mathbf{Y}_i devem ser reprojitados sobre cada uma das imagens, a fim de se obter pontos de correspondência nas duas imagens.

A origem o_i de cada sistema de coordenadas representa o centro de gravidade do objeto, que é obtido através do cálculo do ponto médio do agrupamento de pontos do contorno. O cálculo é dado pelas seguintes expressões:

$$cg_1(x, y) = med(\sum Im_1(x, y)) \quad \forall Im_1(x, y) = 1 \quad (7.17)$$

$$cg_2(x, y) = med(\sum Im_2(x, y)) \quad \forall Im_2(x, y) = 1 \quad (7.18)$$

A figura 7.19 apresenta a localização dos pontos médios que passam a representar o centro de gravidade cg_i do objeto para cada imagem $i=1,2$.

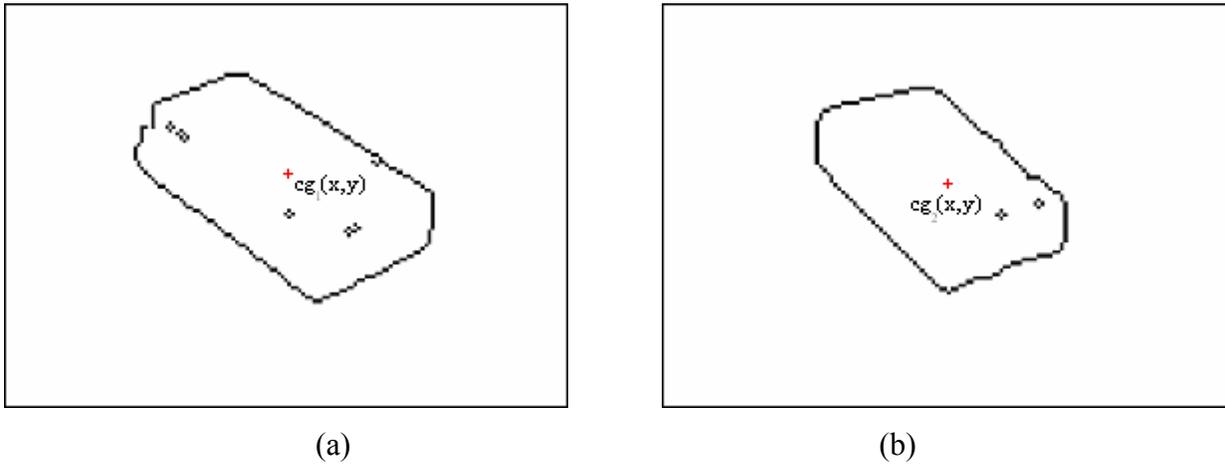


Figura 7.19: Posição dos centros de gravidade para as duas imagens.

Como os centros de gravidade cg_1 e cg_2 não representam o mesmo ponto no espaço, é necessário estabelecer um centro de gravidade espacial médio, obtido pela média da posição tridimensional de cg_1 e cg_2 . A obtenção do centro de gravidade médio depende do conhecimento dos eixos principais do objeto em cada plano de imagem, previamente calculados. O processo de obtenção do centro de gravidade médio será apresentado em 7.2.2.1, cujo objetivo é definir a origem para um sistema de eixos no espaço, comum às duas imagens. A seção 7.2.2.2 apresenta os procedimentos para formar uma caixa de contorno tipo OBB comum às duas projeções, baseado no novo conjunto de eixos do sistema.

7.2.2.1. Reconstrução espacial do centro de gravidade

A reconstrução tridimensional do ponto cg_1 não coincide necessariamente com a mesma localização do ponto cg_2 reconstruído no espaço. Assim eles não representam o mesmo ponto em coordenadas 3D, pois não há intersecção entre as duas retas a_1 e a_2 , como apresenta a figura 7.20. O objetivo é construir um centro de gravidade espacial comum às duas imagens, a partir do centróide de cada objeto de imagem. O centróide pode ser obtido diretamente pela média dos pontos de contorno da imagem, da forma dada pelas expressões 7.17 e 7.18.

Pode ser visto na figura 7.20, que o centro de gravidade cg_1 estará localizado a uma distância αa_1 sobre a reta a_1 e o centro de gravidade cg_2 estará localizado a uma distância βa_2 sobre a reta a_2 . Pode-se assumir que a menor distância entre estas duas retas concorrentes, fornece um ponto médio que representa um centro de gravidade médio comum para ambas as projeções.

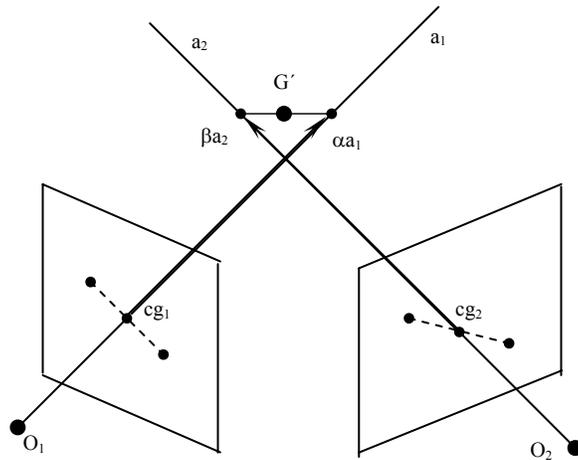


Figura 7.20: Centro de Gravidade médio (G').

A menor distância entre as retas a_1 e a_2 é dada como:

$$\alpha \mathbf{a}_1 = R^T (\beta \mathbf{a}_2 - T) \quad (7.19)$$

Onde o vetor distância \mathbf{d} , possui a seguinte configuração,

$$\mathbf{d} = \alpha \mathbf{a}_1 - R^T \beta \mathbf{a}_2 + R^T T \quad (7.20)$$

E fazendo,

$$\mathbf{d} = \alpha \mathbf{a}_1 + \beta \mathbf{b} + \mathbf{t} \quad (7.21)$$

Onde,

$$\mathbf{b} = \beta R^T \mathbf{a}_2$$

$$\mathbf{t} = R^T T$$

Obtém-se,

$$d^2 = \alpha^2 \mathbf{a}_1^T \mathbf{a}_1 + \beta^2 \mathbf{b}^T \mathbf{b} + \mathbf{t}^T \mathbf{t} + 2\alpha\beta \mathbf{a}_1^T \mathbf{b} + 2\alpha \mathbf{a}_1^T \mathbf{t} + 2\beta \mathbf{b}^T \mathbf{t}$$

Onde a diferença do vetor distância é dada por,

$$\frac{\partial d^2}{\partial \alpha} = 2\alpha \mathbf{a}_1^T \mathbf{a}_1 + 2\beta \mathbf{a}_1^T \mathbf{b} + 2\mathbf{a}_1^T \mathbf{t} \quad (7.22)$$

$$\frac{\partial d^2}{\partial \beta} = 2\beta \mathbf{b}^T \mathbf{b} + 2\alpha \mathbf{a}_1^T \mathbf{b} + 2\mathbf{b}^T \mathbf{t} \quad (7.23)$$

As equações 7.22 e 7.23 podem ser reescritas na forma da equação 7.24, a seguir:

$$\begin{bmatrix} \mathbf{a}_1^T \mathbf{a}_1 & \mathbf{a}_1^T \mathbf{b} \\ \mathbf{a}_1^T \mathbf{b} & \mathbf{b}^T \mathbf{b} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} -\mathbf{a}_1^T \mathbf{t} \\ -\mathbf{b}^T \mathbf{t} \end{bmatrix} \quad (7.24)$$

A partir de 7.24, pode-se encontrar os fatores α e β como sendo:

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \mathbf{a}_1^T \mathbf{a}_1 & \mathbf{a}_1^T \mathbf{b} \\ \mathbf{a}_1^T \mathbf{b} & \mathbf{b}^T \mathbf{b} \end{bmatrix}^{-1} \begin{bmatrix} -\mathbf{a}_1^T \mathbf{t} \\ -\mathbf{b}^T \mathbf{t} \end{bmatrix} \quad (7.25)$$

O ponto médio é dado pela soma do vetor $\alpha \mathbf{a}_1$ com a metade do vetor distância \mathbf{d} , como mostra a equação 7.26, a seguir.

$$\mathbf{G}' = \alpha \mathbf{a}_1 + \frac{\mathbf{d}}{2} = \alpha \mathbf{a}_1 + \frac{\alpha \mathbf{a}_1}{2} - \frac{\beta R^T \mathbf{a}_2}{2} + \frac{R^T T}{2} \quad (7.26)$$

A figura 7.20 apresenta o centro de gravidade médio \mathbf{G}' , obtido a partir dos pontos reconstruídos, $\alpha \mathbf{a}_1 = [X_1, Y_1, Z_1]^T$ e $\beta \mathbf{a}_2 = [X_2, Y_2, Z_2]^T$. O centro de gravidade médio $\mathbf{G}' = [X, Y, Z]^T$ representa a origem de um sistema de eixos 3D comum aos eixos orientados planares de centros cg_1 e cg_2 .

Se o ponto $\mathbf{G}' = [X, Y, Z]^T$ for re-projetado sobre os planos de imagens das duas vistas, existirá um ponto \mathbf{g}' comum às $i=1,2$ imagens em coordenadas de câmera, onde $cg_1' = [x_1, y_1, 1]^T$ é dado por:

$$cg_1' = \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} x/z \\ y/z \\ 1 \end{bmatrix} \quad (7.27)$$

E cg_2' , sendo a posição relativa na segunda imagem, dado pela transformação de corpo rígido através da equação,

$$cg_2' = Rcg_1' + T \quad (7.28)$$

A figura 7.21 apresenta a projeção de \mathbf{G}' sobre os dois planos de imagem. A projeção cg_1' se refere ao centro de gravidade médio projetado sobre a imagem, e é representado na figura pelo símbolo 'o'. O centróide original obtido a partir das equações 7.17 e 7.18, está marcado na figura pelo símbolo '+

Como o centro de gravidade mudou para uma posição corrigida, um novo conjunto de eixos principais deve ser obtido a partir dos novos centros de gravidade cg_i' .

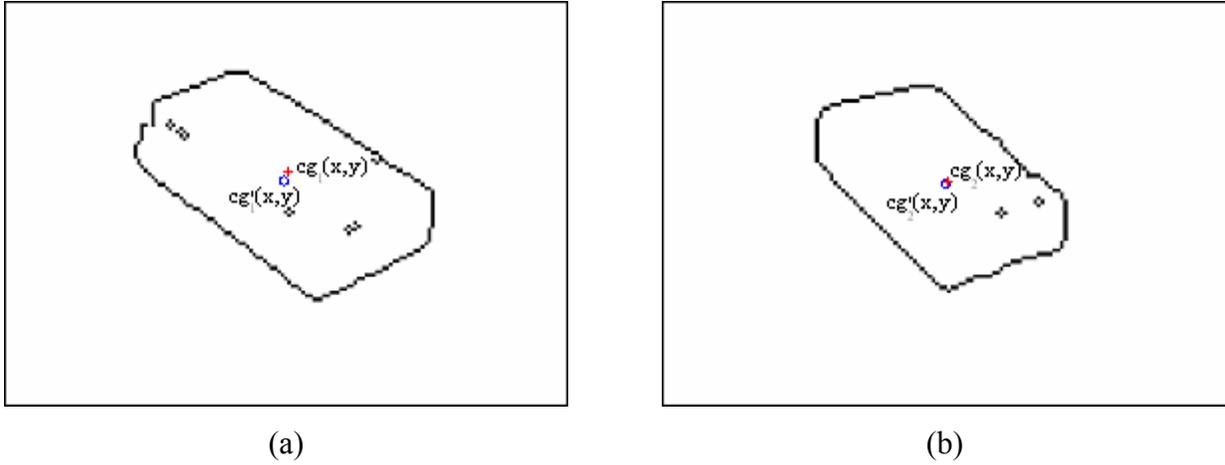


Figura 7.21: Projeção do G' sobre os dois planos de imagem.

Os vetores dos eixos principais \mathbf{X}_i e \mathbf{Y}_i estão em coordenadas de imagem (*pixels*) e podem ser convertidos em coordenadas de câmera pelas equações 7.29 e 7.30 a seguir,

$$\mathbf{X}'_i = \hat{k}^{-1} \mathbf{X}_i \quad (7.29)$$

$$\mathbf{Y}'_i = \hat{k}^{-1} \mathbf{Y}_i \quad (7.30)$$

Onde \hat{k} é a matriz de parâmetros intrínsecos da câmera. Os valores de \mathbf{X}'_i e \mathbf{Y}'_i possuem coordenadas métricas e são utilizados para a criação dos eixos médios comum às duas imagens. A geração destes pontos obedece ao mesmo procedimento adotado para a obtenção do centro de gravidade médio \mathbf{G}' . O novo conjunto destes pontos obtidos é identificado como:

$$E_{m,i} = [p_1 \quad p_2 \quad p_3 \quad p_4 \quad p_5]^T \quad (7.31)$$

Onde $E_{m,i}$ representa as coordenadas dos pontos extremos médios p_k , com $k=1,\dots,5$, em coordenadas de câmera para cada imagem $i=1,2$. A figura 7.22 apresenta as projeções de $E_{m,i}$ em coordenadas de imagem. As posições dos pontos extremos dos eixos principais, que foram

gerados antes da correção, estão marcadas pelo símbolo '+'. Os eixos corrigidos são formados pelos pontos extremos médios denominados de p_k , e estão marcados na figura com o símbolo 'o'.

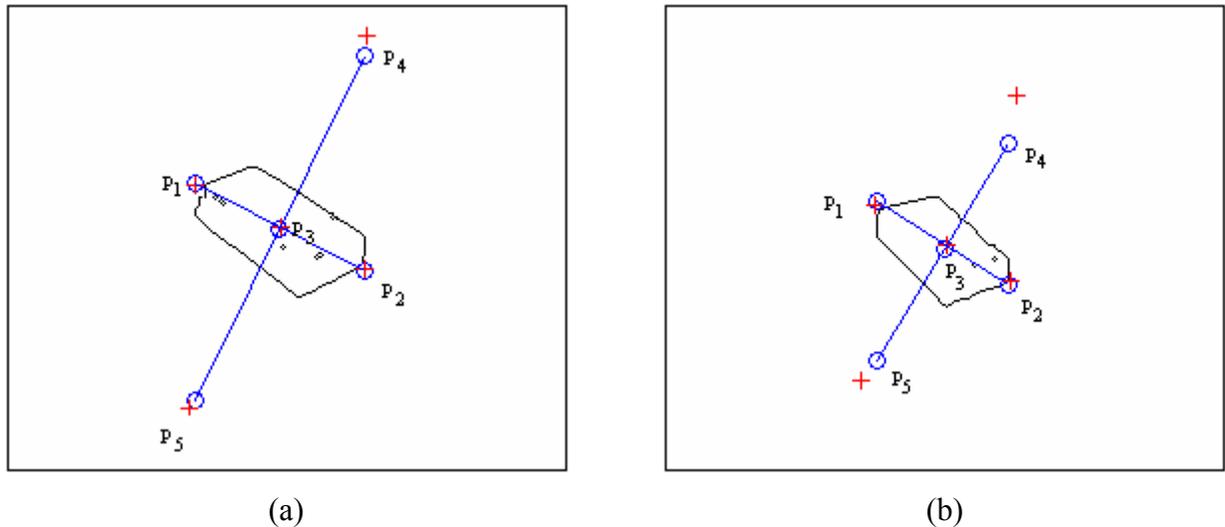


Figura 7.22: Correção dos eixos principais a partir do novo $cg'i$.

A reconstrução a partir dos novos eixos principais gera um único sistema de eixos no espaço, que é comum às duas imagens. Um terceiro eixo Z pode ser adicionado ao sistema pelo produto vetorial dos eixos X e Y conhecidos, conforme o método apresentado pela figura 7.11. A criação deste eixo bem como a geração da caixa de contorno é apresentada em 7.2.2.2.

7.2.2.2. Obtenção da caixa de contorno

Para a criação da caixa de contorno do tipo OBB, o método proposto prevê duas etapas. A primeira consiste em achar a orientação espacial do objeto e construir a caixa virtual 3D. A segunda etapa trata da projeção do OBB sobre os planos de imagem e da realização do ajuste pelo método proposto no capítulo 6.

Para o desenvolvimento da primeira etapa, serão considerados os eixos espaciais principais X'_i e Y'_i , reconstruídos a partir do sistema de eixos $p'_m = [p_1 p_2 p_3 p_4 p_5]^T$. Onde p'_m representa as

projeções de $E_{m,i}$ da figura 7.22. O sistema de eixos no espaço pode ser construído da seguinte forma:

$$V_{3,1} = P_1 - P_3 \quad (7.32)$$

$$V_{3,4} = P_4 - P_3 \quad (7.33)$$

Onde $V_{3,1}$ representa o vetor de coordenadas do segmento de reta entre P_1 e P_3 . E, $V_{3,4}$ representa o vetor de coordenadas do segmento de reta entre P_4 e P_3 . Para as duas equações 7.32 e 7.33 o ponto $P_3 = cg'$.

Como $V_{3,1}$ e $V_{3,4}$ formam um plano, um eixo Z pode ser criado ortogonal a eles. A obtenção do eixo Z é dada por:

$$V_{3,6} = V_{3,1} \times V_{3,4} \quad (7.34)$$

$$V_{3,4}^* = V_{3,6} \times V_{3,1} \quad (7.35)$$

Onde $V_{3,1}$ e $V_{3,4}$ são os eixos X e Y originais, $V_{3,6}$ é o eixo Z ortogonal aos dois primeiros. Como $V_{3,1}$ e $V_{3,4}$ podem não formar um ângulo reto, o eixo Y é recalculado através de $V_{3,4}^*$ forçando a ortogonalidade para XYZ . Um sistema de eixos 3D pode ser formado como,

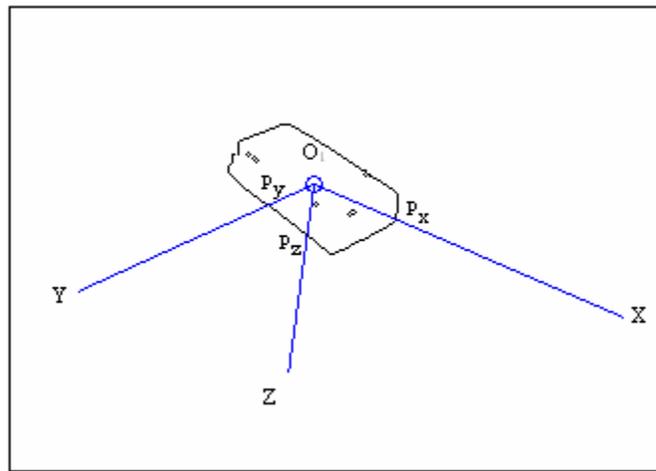
$$\mathbf{V}_1 = [V_{3,1} \quad V_{3,4}^* \quad V_{3,6}]^T \quad (7.36)$$

Assim \mathbf{V}_1 possui as coordenadas dos eixos espaciais para a primeira imagem. De forma semelhante, \mathbf{V}_2 pode ser obtido para a segunda imagem como,

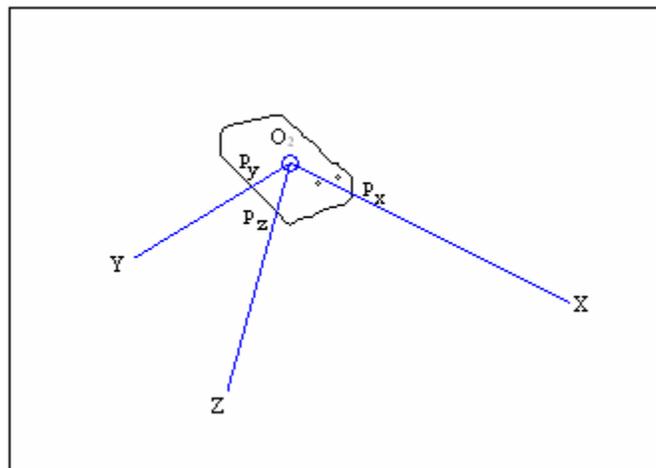
$$\mathbf{V}_2 = R\mathbf{V}_1 + t \quad (7.37)$$

Onde R e t são os parâmetros de pose da segunda câmera em relação à primeira.

A figura 7.23, apresenta o resultado do processo de criação do eixo Z , e dos eixos X e Y a partir da reconstrução dos sistemas de eixos dos planos.



(a)



(b)

Figura 7.23: Projeção dos eixos principais xyz nos planos de imagem (a) e (b).

A partir dos eixos conhecidos é possível construir uma caixa de contorno do tipo OBB, comum às duas imagens. As interseções dos eixos XYZ com os pontos de contorno p_x , p_y e p_z do objeto, permitem delimitar a região de enquadramento pela caixa de contorno. A figura 7.24 apresenta a construção da caixa de contorno. Cada ponto do contorno p_x , p_y e p_z origina o centro de uma das faces da caixa virtual.

A soma destes vetores nas quatro direções permite obter os vértices da caixa de contorno inicial.

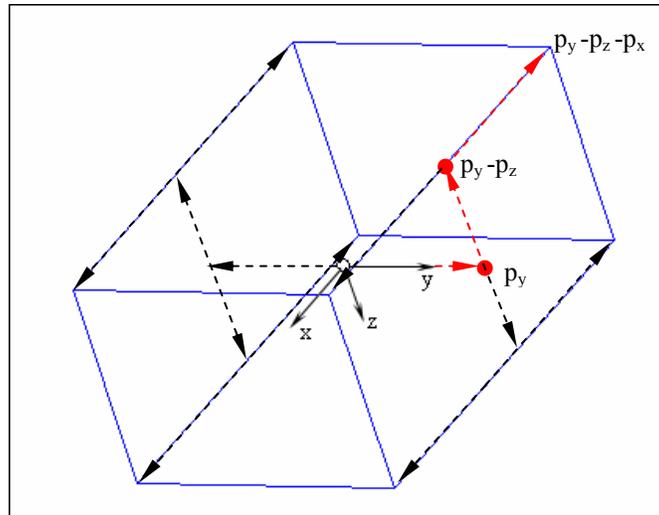


Figura 7.24: Construção espacial da caixa de contorno.

A caixa virtual é formada por duas faces principais chamadas de face anterior F_a e face posterior F_p . A face anterior F_a é formada pelos vértices v_1, v_2, v_3 e v_4 . Os vértices são calculados por,

$$v_1 = p_x - p_z + p_y \quad v_2 = p_x - p_z - p_y \quad (7.38)$$

$$v_3 = p_x + p_z - p_y \quad v_4 = p_x + p_z + p_y$$

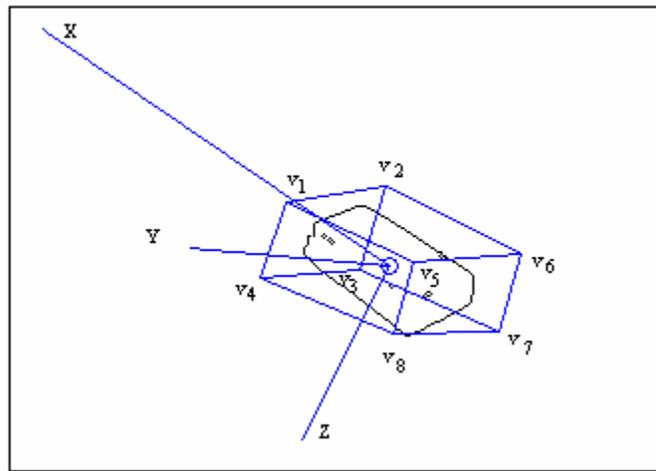
A face posterior F_p é formada pelos vértices v_5, v_6, v_7 e v_8 . Estes vértices são calculados por,

$$v_5 = -p_x - p_z + p_y \quad v_6 = -p_x - p_z - p_y \quad (7.39)$$

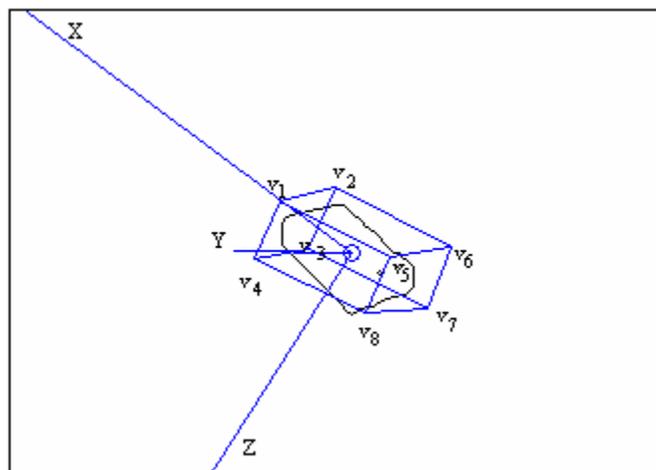
$$v_7 = -p_x + p_z - p_y \quad v_8 = -p_x + p_z + p_y$$

As demais faces são formadas pela combinação dos vértices de F_a e F_p .

O resultado da formação dos vértices é apresentado na figura 7.25 para cada uma das imagens.



(a)



(b)

Figura 7.25: Ajuste do OBB para as imagens de exemplo.

A figura 7.25 mostra que a caixa de contorno obtida para o par de imagens não se ajusta completamente às arestas do objeto. Isto ocorre porque os eixos orientados passam pelo centro geométrico de cada face da caixa de contorno, que fazem a intersecção com os pontos médios das arestas do objeto. Para resolver o problema deve-se dar um deslocamento aos eixos XYZ a fim de se obter a nova orientação. A construção do novo conjunto de eixos é demonstrada na figura 7.26.

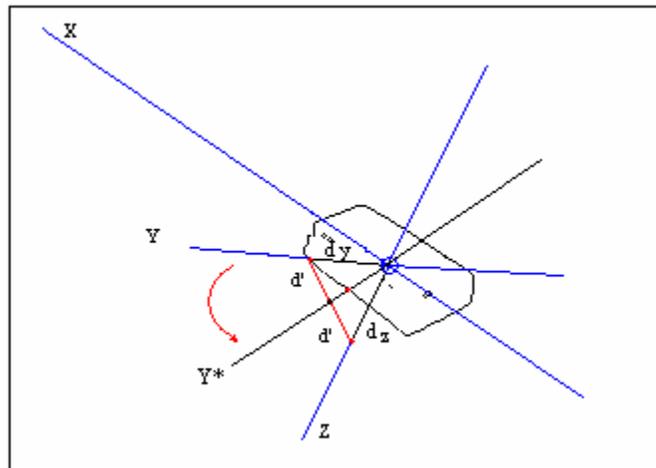


Figura 7.26: Exemplo de obtenção do eixo Y^* a partir de Y .

A figura 7.26 apresenta o esboço para o reposicionamento de eixos. A distância d_y entre a origem do sistema e o ponto p_y , pode ser projetada sobre o eixo Z , obtendo-se $d_z = d_y$. Unindo-se d_y a d_z pode ser obtida uma distância média d' . A reta formada entre o centro de gravidade e este ponto médio permite formar o novo eixo Y^* . A interseção deste eixo com os elementos de contorno origina um novo ponto p_{y^*} .

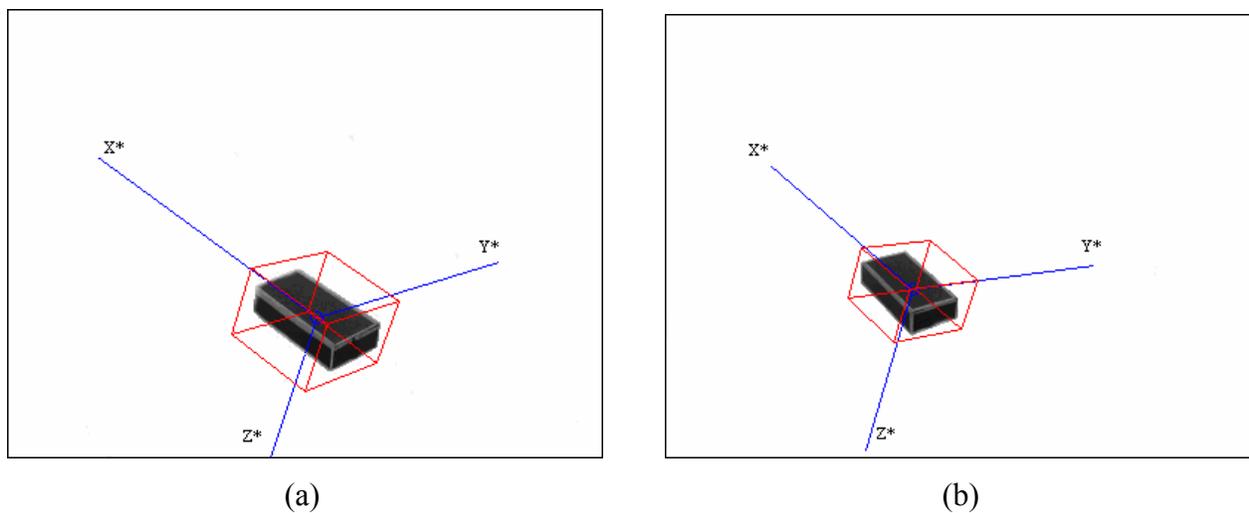
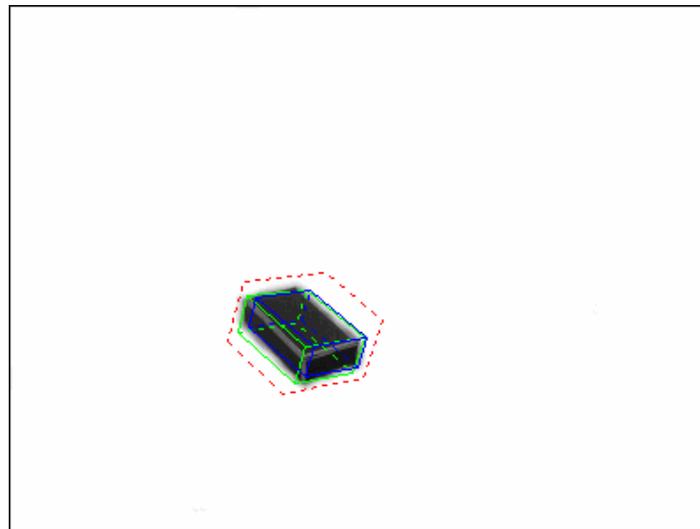
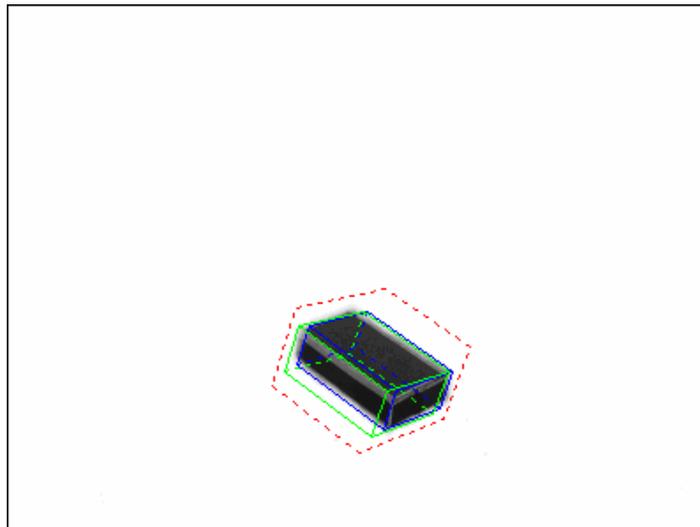


Figura 7.27: Caixa de contorno virtual para referência inicial do ajuste.

Uma vez estabelecida a caixa de contorno inicial, pode ser realizado o ajuste de arestas pelo método apresentado no capítulo 6. A figura 7.28 apresenta o resultado do enquadramento via uma caixa de contorno tipo OBB. Cada uma das imagens possui três caixas de contorno. A mais externa definida pelas linhas pontilhadas se refere à projeção da caixa de contorno inicial. As outras duas formam a resposta do enquadramento.



(a)



(b)

Figura 7.28: Resultado final do ajuste da caixa de contorno.

Pode ocorrer que partes de um objeto que são vistos por uma câmera, podem estar ocultos para a segunda câmera. Em função disto a estimativa do volume pode não representar a realidade. Para cada imagem, duas caixas de contorno são geradas. Na figura 7.28, o enquadramento representado pelas arestas na cor azul se refere ao ajuste da caixa virtual sobre o objeto da mesma imagem. O enquadramento representado pelas arestas na cor verde se refere ao ajuste sobre o objeto com os parâmetros da segunda imagem. Isto é feito para garantir que a maior quantidade de informação do objeto esteja envolvida pelo volume gerado pelas duas caixas de contorno.

A criação de um OBB para cada imagem permite definir a orientação do objeto. O volume ocupado pelo objeto depende da combinação de duas caixas de contorno. A união das duas caixas de contorno define o volume total do objeto. O item 7.3 a seguir apresenta os resultados do volume calculado para a peça de exemplo.

7.3. Estimativa do volume

Cada um dos eixos principais do objeto faz a interseção com um dos pontos do seu contorno determinando o comprimento de cada eixo. Conhecendo-se os eixos principais do objeto, é possível estimar o seu volume aproximado pelo cálculo do volume da caixa de contorno ajustada. A figura 7.29 apresenta o arranjo de uma caixa de contorno construída no espaço.

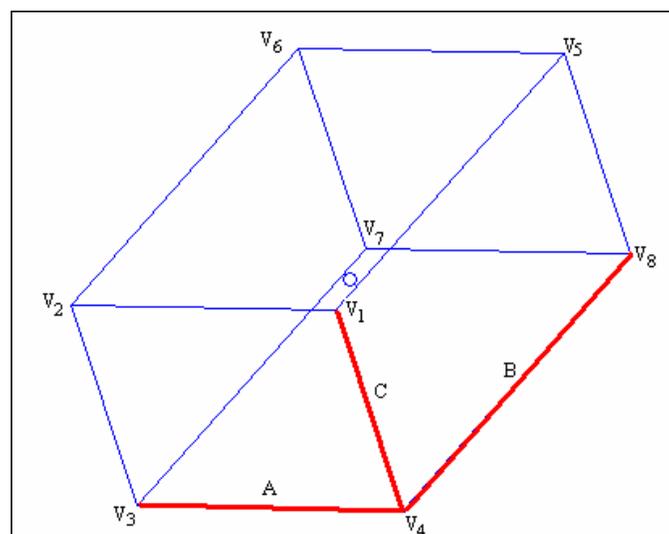


Figura 7.29: Identificação dos vértices para a caixa de contorno 3D.

A partir da figura 7.29, o cálculo do volume V_i de cada caixa de contorno projetada sobre a imagem i , com $i=1,2$ pode ser obtido pela seguinte relação:

$$V_i = A_i \cdot B_i \cdot C_i \quad (7.40)$$

Onde, A , B e C são as distâncias entre os vértices e podem ser obtidos como,

$$A = |\mathbf{V}_4 - \mathbf{V}_3| \quad B = |\mathbf{V}_8 - \mathbf{V}_4| \quad C = |\mathbf{V}_4 - \mathbf{V}_1| \quad (7.41)$$

O volume total V_t é dado pela composição dos volumes das duas caixas de contorno que foram gerados a partir da imagem 1 e da imagem 2, como ilustrado na figura 7.28. O volume pode ser calculado pela seguinte relação:

$$V_t = \frac{V_1 + V_2}{2} + |V_1 - V_2| \quad (7.42)$$

A tabela 7.3 apresenta a comparação entre os valores medidos diretamente do objeto da figura 7.16 e os valores calculados pela aproximação da caixa de contorno. O objeto possui dimensões de 11cm x 5.4cm x 2.5cm.

Volume aproximado do objeto	
Volume Medido	148.5 cm ³
Volume obtido do OBB ajustado a partir da imagem 1	142.8 cm ³
Volume obtido do OBB ajustado a partir da imagem 2	124.5 cm ³
Volume obtido pela composição das duas imagens	151.9 cm ³

Tabela 7.3: Tabela comparativa de valores de volume medido e calculado.

A tabela 7.4 apresenta a porcentagem do erro no volume encontrado através do ajuste das caixas de contorno.

Percentual de erro dos volumes encontrados	
OBB da imagem 1 – volume real	3.8 %
OBB da imagem 2 – volume real	16.1 %
OBB composto – volume real	2.3 %

Tabela 7.4: Tabela comparativa de erros de volume.

Pela análise dos valores de volume encontrados em função dos erros obtidos, pode-se verificar pela tabela 7.4 que o volume gerado pela composição das duas caixas de contorno possui o menor erro dentre as estimativas de volume.

7.4. Resumo do capítulo

Os trabalhos desenvolvidos por Chan(2001) e Jiantao(2004) tratam da construção da caixa de contorno baseado em modelos 3D de objetos, onde todas as entidades geométricas são conhecidas. De forma diferente, o método descrito neste capítulo se baseia na recuperação da informação dos atributos de um objeto a partir de imagens e da sua reconstrução tridimensional. Neste caso nem sempre é possível se obter todas as informações do objeto. Normalmente existem atributos que estão presentes em uma das imagens e não na outra, de forma que não existe maneira segura de ajustar uma caixa de contorno baseando-se somente na informação do par de imagens. Neste contexto foram apresentadas duas formas de gerar uma caixa de contorno. A primeira usando a definição de eixos alinhados (AABB), onde o objeto deve estar previamente alinhado com o sistema de eixos do gabarito de calibração. Para eliminar esta dependência, foi proposta também a criação de uma caixa de contorno de eixos orientados (OBB) orientada pelos eixos principais do objeto. Foi possível construir um OBB a partir da imagem sem usar o gabarito de calibração como referência inicial. A teoria proposta para a determinação da posição de um sólido e seu volume, foi apresentada neste capítulo através da utilização de métodos de processamento de imagens, e da formulação matemática para obtenção das direções principais. Exemplos de resultados para ajuste por AABB e OBB foram ilustrados através de figuras.

Capítulo 8

Aplicação do método em uma célula de manufatura

A utilização de sistemas de visão computacional para a realização de controle de qualidade, está se tornando muito comum em ambientes automatizados de manufatura. As respostas geradas por sistemas de visão podem permitir a correção de procedimentos de maneira a se evitar interrupções nos processos de fabricação. Também podem ser utilizados para reduzir etapas de um processo. Dentro destes contextos, o método proposto nos capítulos 6 e 7, pode ser aplicado a um problema real descrito no item 8.1 a seguir.

8.1. Descrição do funcionamento da estação de soldagem da célula de manufatura

No laboratório de automação da manufatura descrito em Rudek (2001), uma peça como a da figura 8.1 pode ser produzida sem intervenção humana, através de processos que vão desde a usinagem e soldagem, até a montagem do produto final.



Figura 8.1: Imagens da peça a ser posicionada.

A célula é formada por um robô de seis graus de liberdade, uma bancada de posicionamento da peça e um sistema de solda. A figura 8.2 apresenta a imagem do robô e a bancada de posicionamento da peça já com o gabarito de calibração da câmera.

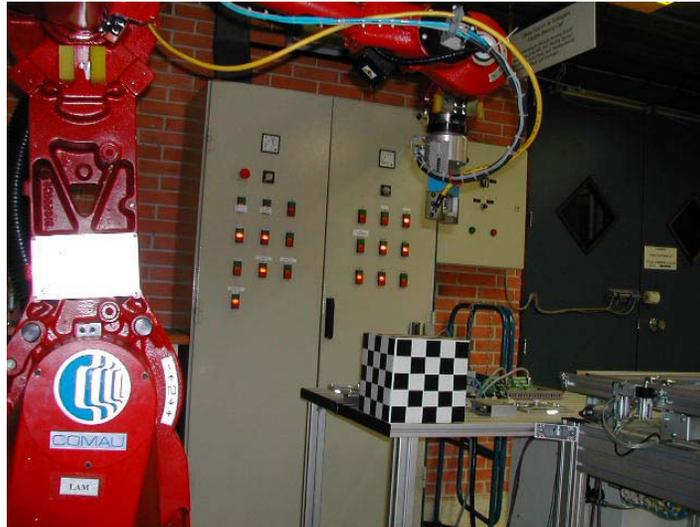
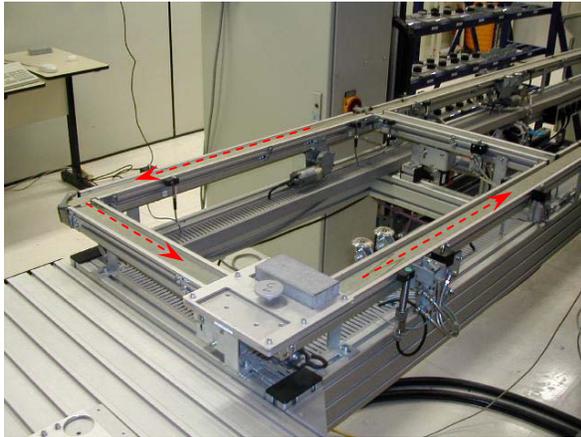


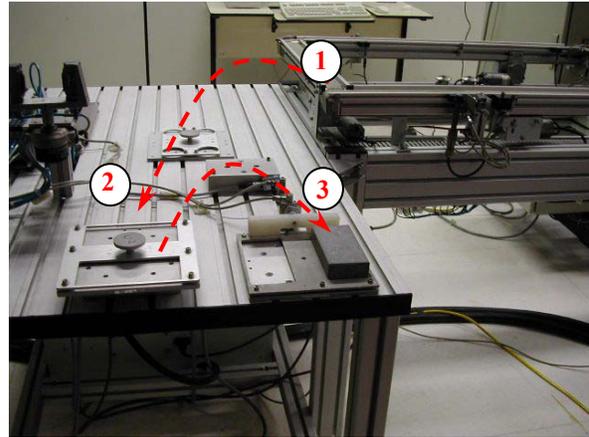
Figura 8.2: Vista parcial da estação de soldagem.

Uma etapa importante do processo de fabricação realizado por esta célula consiste em uma operação de soldagem sobre a caixa metálica. Para a execução deste procedimento a peça deve ser retirada de uma esteira de transporte através de um manipulador robótico e deve ser depositada sobre uma bancada, para ter a sua posição ajustada por um atuador pneumático. Após o ajuste ela deve ser transportada pelo robô até a estação de soldagem. A figura 8.3 apresenta o processo com as etapas identificadas de (a) até (d), descritas a seguir:

- (a) O objeto é primeiramente transportado em um *pallet* por uma esteira até a chegada a estação.
- (b) Em seguida o *pallet* é retirado da esteira pelo robô e colocado sobre a bancada. Cada *pallet* pode transportar duas peças. O robô pega uma das peças e coloca em um posicionador com atuação pneumática.
- (c) Após a peça ser ajustada pelo posicionador, é pega em seguida pelo robô que dá continuidade ao processo.
- (d) Idem a imagem (c) com vista de uma posição mais elevada.



(a)



(b)



(a)



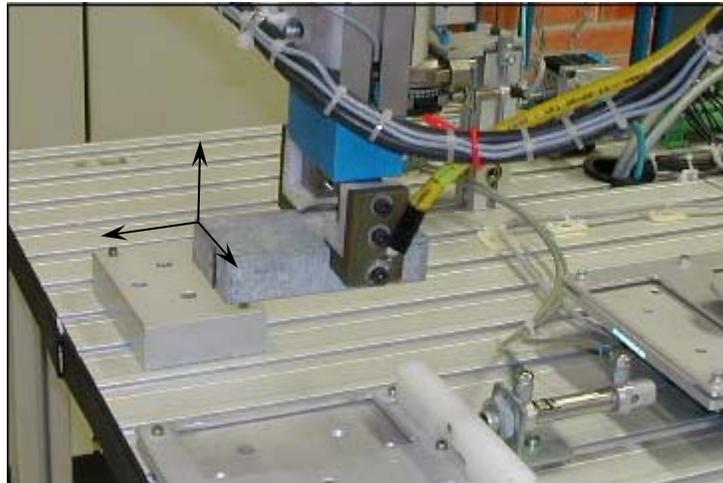
(b)

Figura 8.3: (a) esteira de transporte. (b) bancada. (c) peça sendo pega pelo robô. (d) peça sendo transportada.

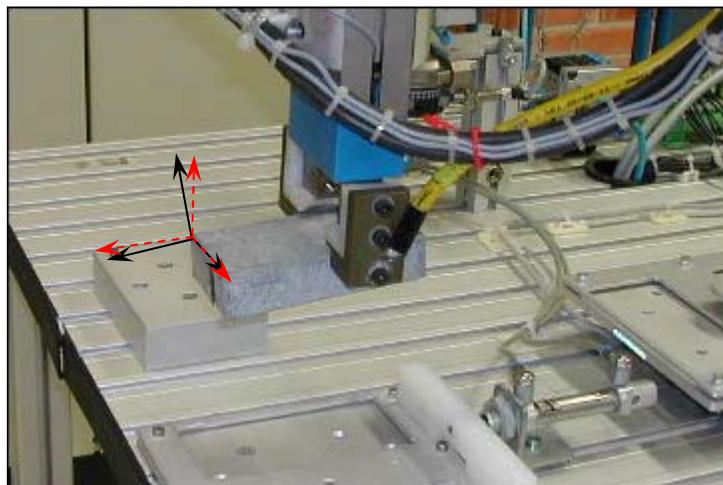
Durante este processo, três problemas podem ocorrer:

- i. *Problema de posicionamento*: Uma modificação na posição do objeto pode acontecer quando a peça é colocada no atuador de posição ou quando ele libera a peça.
- ii. *Falta de peça*: Se não existir uma peça no *pallet* retirado da esteira de transporte, o posicionador irá esperar a peça até que sistema aborte a operação por *time-out*.
- iii. *Peças Diferentes*: O *pallet* pode ter transportado uma peça diferente do objeto esperado para o processo.

O primeiro problema citado é referente a um erro de posição da peça a ser pega. A figura 8.4 apresenta esta situação.



(a)



(b)

Figura 8.4: (a) Peça na posição correta. (b) peça com posição incorreta.

Para identificar problemas como o apresentado pela figura 8.4. pode ser usado o método de ajuste da caixa de contorno desenvolvido nos capítulos 6 e 7. Através da caixa de contorno é possível encontrar as direções principais do objeto e verificar se a posição dos eixos correspondem ao esperado. Para o melhor andamento do processo, é importante que esta verificação seja feita antes que o manipulador pegue a peça. Isto evita o movimento

desnecessário do braço robótico. O objetivo é aplicar o método para identificar as informações geométricas da peça a ser manipulada, e com isto permitir uma decisão sobre o andamento do processo.

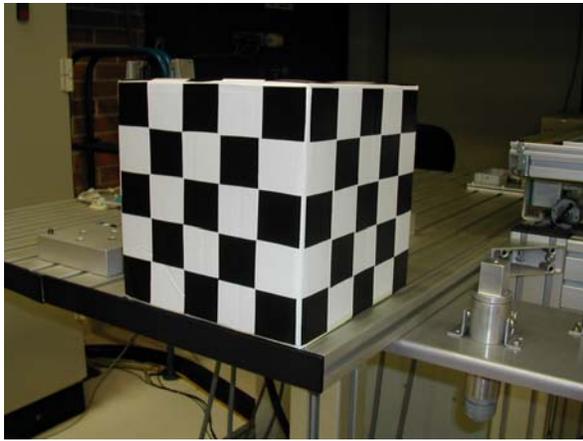
O propósito do ajuste de uma caixa de contorno sobre o objeto, é verificar se a peça que vai ser transportada está convenientemente posicionada. Caso haja posicionamento incorreto, a célula de manufatura pode requisitar o re-posicionamento da peça pelo robô, ou a correção do movimento com a compensação do erro observado. Esta estratégia é importante para o processo, pois evita que a operação de solda que será feita depois, possa ocorrer de forma incorreta. Ao invés de se ter um controle de qualidade após a solda, aceitando ou refugando peças manufaturadas, o sistema pode primeiro corrigir o movimento, evitando o refugo de peças, e trazendo com isto economia na produção.

A seguir no item 8.2, será apresentada a aplicação do método da caixa de contorno para a solução do problema descrito anteriormente.

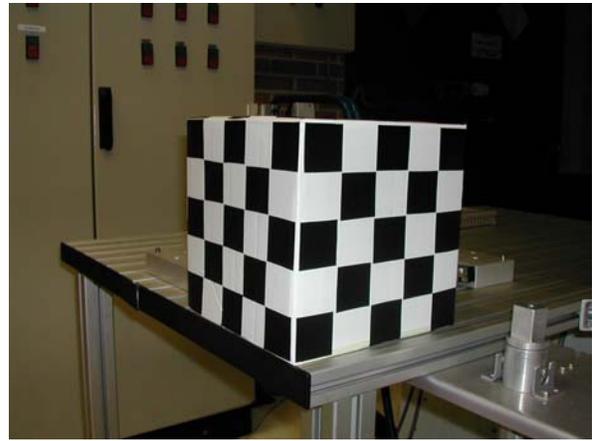
8.2. Aplicação do método da caixa de contorno

O método da caixa de contorno será aplicado à peça localizada sobre a bancada, na posição do atuador pneumático, onde ocorrerá a verificação se o objeto se encontra na orientação desejada. Espera-se obter como resultado uma orientação e um volume para a caixa de contorno. Se esta posição e o volume forem diferentes do esperado, o sistema de controle da célula pode rejeitar a operação.

Para aplicação do método todos os parâmetros do ambiente devem ser conhecidos. A etapa inicial diz respeito à calibração das câmeras, conforme as técnicas descritas nos capítulos anteriores. A figura 8.5 apresenta o gabarito de calibração posicionado sobre a bancada. As imagens foram adquiridas com uma câmera CCD Canon de 3.2 Megapixels de resolução com iluminação obtida do próprio ambiente. O processo foi implementado através de um programa desenvolvido em MatLab, de forma a interagir com os procedimentos de calibração desenvolvidos por Zhang (2000).



(a)



(b)

Figura 8.5: Configuração do par de imagens para a calibração do sistema.

A partir das imagens da figura 8.5, pode ser executado o método de Zhang como descrito no capítulo 6, para a calibração dos parâmetros intrínsecos e para a estimação dos parâmetros de pose da câmera. A figura 8.6 apresenta a extração das coordenadas dos cantos do gabarito pelo método de Zhang usados para realizar a correspondência entre o par de imagens, necessários a criação da matriz essencial.

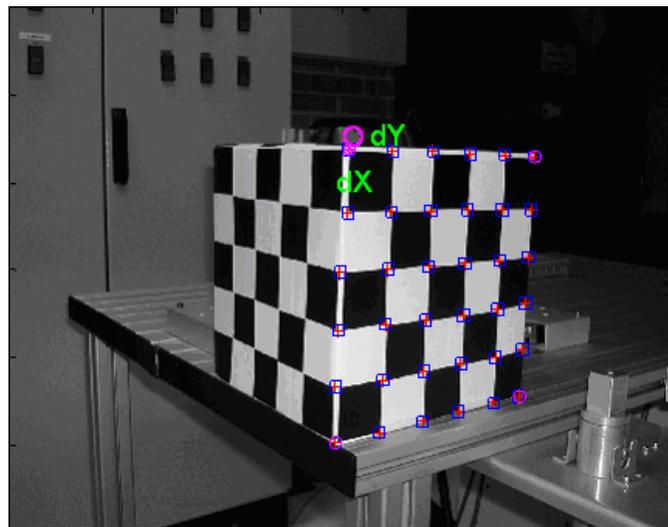


Figura 8.6: Detecção de cantos pelo método de Zhang (2000) para uma das imagens.

A tabela 8.1, apresenta o resultado da calibração com os parâmetros intrínsecos estimados, e a tabela 8.2 apresenta os parâmetros de pose que relacionam a primeira câmera e a segunda câmera num mesmo referencial.

Definição dos parâmetros intrínsecos	
<i>Focal Length (fc):</i>	[391.54802 391.54802]
<i>Principal point (cc):</i>	[199.50000 149.50000]
<i>Skew (alpha_c):</i>	[0.00000]
<i>Distortion (kc):</i>	[0.00000 0.00000 0.00000 0.00000 0.00000]

Tabela 8.1: Resultado da calibração das câmeras pelo método de Zhang.

Definição dos parâmetros de pose	
<i>Translation vector (Tc_ext):</i>	[-179.256262 -52.430956 308.136994]
<i>Rotation matrix (Rc_ext):</i>	[-0.035553 0.962250 -0.269836
	0.989562 0.071627 0.125043 0.139650 -0.262574 -0.954753]
<i>Pixel error:</i>	[2.05400 2.35970]

Tabela 8.2: Resultados dos parâmetros de pose.

As informações da tabela 8.1 permitem formar a matriz K de parâmetros intrínsecos definida pela equação 3.21 no capítulo 3. As informações da tabela 8.2 permitem definir a matriz de rotação \mathbf{R} e o vetor de translação \mathbf{T} , da equação 2.5 do movimento de corpo rígido definida no capítulo 2.

Para se realizar o ajuste da caixa de contorno sobre o objeto é necessário identificar a região de interesse (*ROI*). A região de interesse é dada pela subtração de uma imagem contendo o objeto, da outra imagem contendo apenas a bancada, seguida pelos procedimentos de segmentação apresentados no capítulo 4. A figura 8.7 a seguir apresenta duas imagens de uma das câmeras. A imagem 8.7 (a) contém a peça que será subtraída da imagem 8.7 (b) sem a peça. O resultado da segmentação executada sobre a imagem resultante da subtração, gera a peça segmentada apresentada pela figura 8.8.

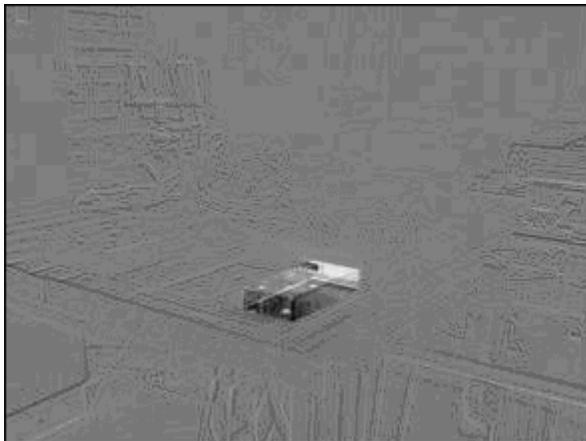


(a)

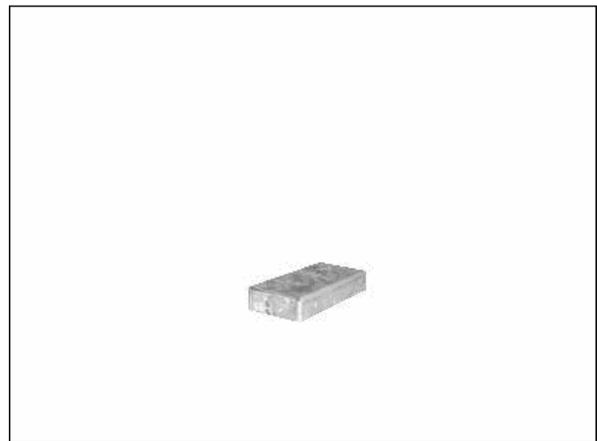


(b)

Figura 8.7: (a) imagem da bancada com a peça. (b) imagem da bancada sem a peça



(a)

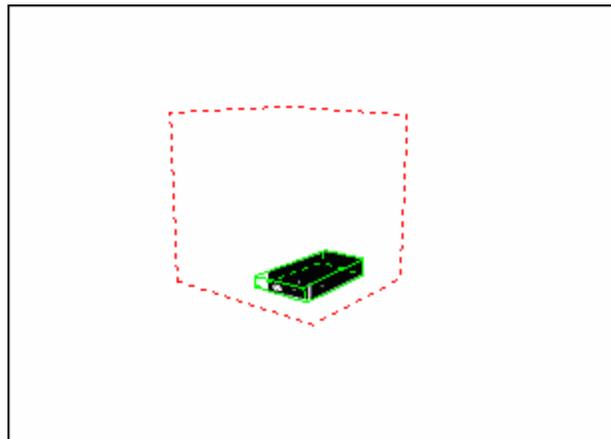


(b)

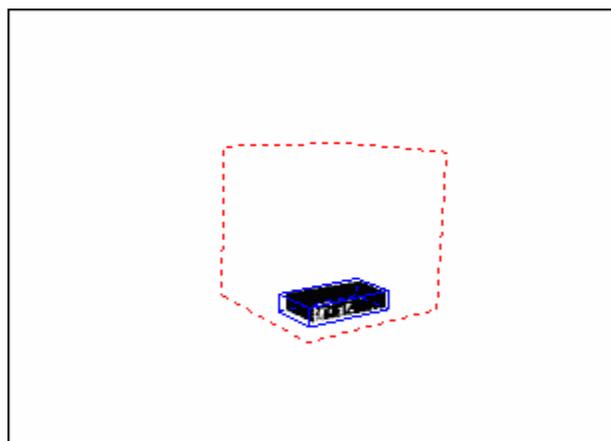
Figura 8.8: (a) Imagem resultante da subtração. (b) imagem com segmentada.

Após a identificação da região de interesse deve-se realizar o procedimento de ajuste da caixa de contorno sobre o objeto. Para esta aplicação será criado um ajuste do tipo AABB por eixos alinhados com o gabarito de calibração. Este tipo de enquadramento pode ser realizado porque a simetria da bancada, da peça, e do gabarito são equivalentes. Com isto se houver uma diferença de alinhamento da peça, o volume será diferente do esperado.

Como apresentado pela figura 8.9, é possível encontrar um bom ajuste para o objeto. A linha pontilhada mais externa define a caixa virtual usada para iniciar o processo de ajuste da caixa de contorno. O processo de redimensionamento do AABB é dado pela técnica de ajuste das arestas da caixa de calibração baseada nos pontos de fuga, conforme descrito no capítulo 6. Através do volume obtido pela caixa de contorno, pode ser feita uma comparação com o volume real da peça. O volume real do objeto pode ser previamente calculado e armazenado no sistema de controle da planta. Após a aquisição de imagem e do ajuste, o volume estimado pode ser comparado com o valor armazenado. Se houver diferença significativa neste valor, a estação pode abortar a execução da tarefa.



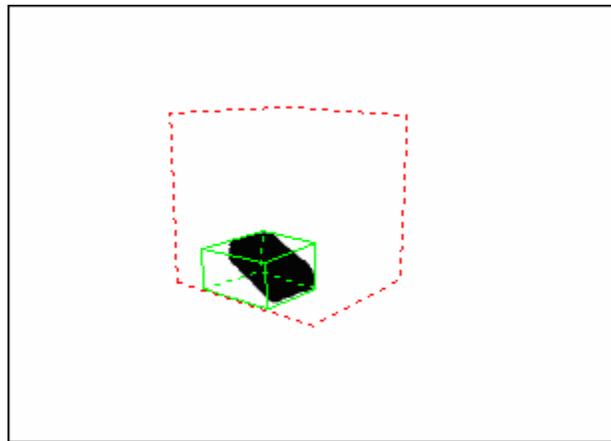
(a)



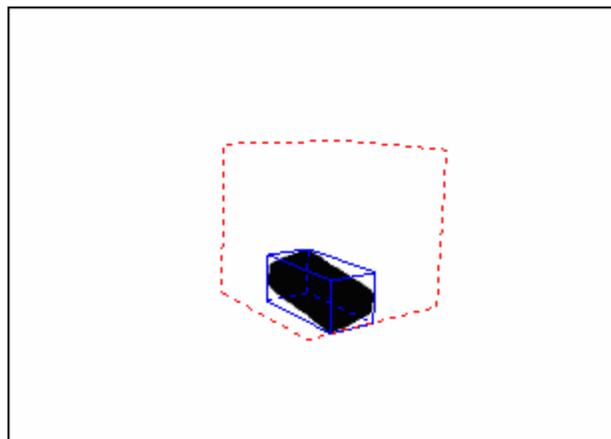
(b)

Figura 8.9: Imagens resultantes após o ajuste do *bouding box* AABB.

A figura 8.10 apresenta um exemplo onde o volume calculado do objeto será diferente do valor real. Isto acontece porque a posição do objeto difere da posição da caixa de contorno inicial. Esta análise é viável porque o ambiente de posicionamento de peças nunca muda. Pelo fato de se requerer sempre uma mesma posição é possível operar com um AABB, como descrito no capítulo 6.



(a)



(b)

Figura 8.10: Diferença de volume ocupado em função de um posicionamento indevido.

8.3. Resumo do capítulo

O capítulo apresentou a aplicação do método proposto para um problema real de uma célula de manufatura. O modelo de aquisição de imagens foi baseado na utilização de uma câmera fixa, posicionada em duas localizações distintas, para formar o par de imagens. As imagens foram tomadas da região da bancada onde o objeto é manipulado pelo robô. Foi utilizado um ajuste AABB para a determinação do espaço ocupado por um objeto, porque o ambiente permanece constante durante todo o processo e a peça operada possui simetria equivalente ao gabarito de referência.

O método pode ser validado pela comparação da posição e volume real de uma peça, com o valor calculado obtido a partir das imagens. Desta forma pode-se decidir se um objeto pode ser transportado pelo robô, ou se a operação deve ser cancelada ou corrigida.

Para o processo exemplificado neste capítulo, a utilização do método pode ajudar na solução de alguns problemas como:

a) Identificação da posição da peça no *pallet*:

Normalmente um objeto é transportado sobre um *pallet* entre as diversas células de um ambiente de manufatura. Pode ocorrer por algum motivo não previsto, que um *pallet* não contenha uma peça, ou que a peça possa estar em uma posição diferente da programada para aquele trecho do transporte. Assim, o atuador robótico não consegue dar continuidade ao processo. O método proposto pode identificar se uma peça ocupa a posição correta. Caso contrário, o robô pode modificar a posição de pegada.

b) Verificação da existência da peça:

O sistema proposto pode permitir a verificação, se a peça existe na posição esperada para aquele processo. Se o *pallet* está vazio então a célula não precisa responder aos comandos programados naquele momento.

c) Classificação de tipos de peça diferentes:

Um sistema de manufatura pode trabalhar com a fabricação de mais de um tipo de produto simultaneamente. Assim o sistema pode verificar quais são as características da peça que está chegando à célula e ajustar o programa adequado para cada tipo de objeto.

d) Detecção de colisão:

Conhecendo o volume ocupado pela peça, pode-se evitar a colisão do objeto transportado pelo braço robótico; ou pode-se ainda permitir a um robô autônomo desviar de obstáculos no seu trajeto.

e) Embalagem:

A teoria proposta permite estimar um volume através de uma caixa de contorno. Esta caixa de contorno pode definir a região ocupada por um objeto de forma a construir moldes ou embalagens para o objeto.

A utilização de uma caixa de contorno (AABB) permite identificar a partir de imagens, se um objeto está na posição devida e se a sua orientação está conforme a esperada. A determinação do volume, bem como das direções principais permitem resolver um problema comum dos processos automatizados, que é o de garantir a qualidade final do produto fabricado.

Capítulo 9

Conclusão

A tese apresentou um método de aplicação da visão computacional para o problema da reconstrução espacial de um sólido a partir de duas projeções de sua imagem. Muitos métodos de reconstrução de sólidos são apresentados na literatura e os resultados destes trabalhos demonstram a grande dificuldade em se obter as informações métricas de um objeto, diretamente de sua imagem. A proposta apresentada configura um caminho diferente do adotado na maioria dos processos de reconstrução, uma vez que não requer a reconstrução completa da superfície de um objeto.

O objetivo principal do método é descrever a posição e o volume aproximado do objeto, considerando um conjunto reduzido de informações. Na prática o método propõe que as informações espaciais sobre o objeto podem ser descritas pela reconstrução de apenas seis pontos da imagem. Estes pontos são formados pela correspondência entre duas imagens dos seis vértices visíveis de um gabarito de calibração conhecido. Uma caixa virtual pode ser construída usando os pontos de fuga das arestas conhecidas presentes na imagem do gabarito. Através de um processo de movimentação destas arestas é possível ajustar a caixa virtual sobre o contorno do objeto, criando uma caixa de contorno (*bounding box*). A maioria das aplicações que envolvem a utilização das caixas de contorno está relacionada com a delimitação de regiões ou com acompanhamento de trajetória e análise de colisão de objetos. Muitos destes processos se baseiam em imagens sintéticas onde os parâmetros de um ambiente tridimensional são previamente conhecidos. A proposta da tese difere destes métodos em dois aspectos. Primeiro porque se baseia na reconstrução 3D de pontos de uma imagem real. E segundo, usa o conceito de pontos de fuga para realizar o ajuste ao mínimo volume do objeto.

O texto apresentou o exemplo de um enquadramento sobre uma peça que é manipulada de forma automática em ambiente real de manufatura. Com a aplicação do método é possível identificar se o objeto se encontra na posição esperada, e se ele possui a orientação necessária à continuidade do processo. Apesar do enfoque do trabalho ser direcionado a um tipo específico de peça e de problema, este processo poderia ser aplicado a outros objetos de formas variadas.

Alguns projetos futuros podem acrescentar modificações no método de forma a incluir algumas melhorias. Uma melhoria importante seria a geração de uma caixa de contorno ajustada ao objeto diretamente a partir da imagem sem usar uma caixa de referência. Isto pode tornar os processos mais dinâmicos, evitando a dependência da construção física de um gabarito. Também, pode ser otimizada a rotação das arestas, baseada numa movimentação dos pontos de fuga ao invés de deixá-los fixos. Outra aplicação possível seria usar o conhecimento dos pontos de fuga para escolher o melhor posicionamento das câmeras. Neste caso, se nem todas as arestas são visíveis, significa que a posição da câmera não está adequada para a aquisição, e um sistema automático poderia escolher uma melhor posição para a câmera baseada nos pontos de fuga das arestas do gabarito. Também, uma melhoria importante está relacionada à avaliação do método em relação a problemas de iluminação, onde a região de interesse pode não ser identificada com precisão. Um banco de dados associado ao sistema de visão poderia operar com este método para classificar e separar peças de formas geométricas e dimensões diferentes.

A experiência adquirida pela elaboração deste trabalho permitiu consolidar um grupo de estudo permanente, integrando as duas instituições PUC PR e UNICAMP, de forma a atuar em projetos conjuntos relacionados à utilização da visão computacional para a automação de processos.

Referências Bibliográficas

- Achour, K; Benkhelif, M. *A new approach to 3D reconstruction without camera calibration*. Pattern Recognition, 34, pp.2467-2476, 2001
- Augusteijn, M. F.; Clemens, L. E. *A neural-network approach to the detection of texture boundaries*, Engineering Applications of Artificial Intelligence, vol. 9, no. 1, pp. 75-81, 1996.
- Banerjee, S. *Camera Models and Affine Multiple Views Geometry*. Dept. Computer Science and Engineering, IIT Delhi, 2001.
- Barequet, G.; Har-Peled, S. *Efficiently Approximating the Minimum-Volume Bounding Box of a Point Set in Three Dimensions*. Journal of Algorithms, n.38; pp. 91-109, 2001.
- Beis, J. S.; Lowe, D. G. *Indexing without invariants in 3D object recognition*. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 21, no. 10, 1999, pp.1000-1014.
- Bourke, Paul. *Polygonising a Scalar Field*. <http://astronomy.swin.edu.au/~pbourke/modelling/polygonise>; 1997.
- Castleman, Kenneth R. *Digital Image Processing*. Prentice-Hall, 1996.
- Chan, C.K.; Tan, S.T. *Determination of the minimum bounding box of an arbitrary solid: an iterative approach*. Computer and Structures, n.79, pp.1433-1449, 2001.

- Chan, C.K.; Tan, S.T. *Extreme points of a large 3D point set along multiples directions*. Computer-Aided Design, n.37, pp.17-34, 2005.
- Chase, Brian. *Calibration of Scanning Laser Ranger Cameras With Applications for Machine Vision*. Dissertação de Mestrado, University of Tennessee, Knoxville, 1999.
- Cheng, H. D.; Xu, H. X. *A novel fuzzy logic approach to contrast enhancement*, *Pattern Recognition*. vol.33, pp. 809-819, 2000.
- Chu, C. W.; Hwang, S.; Jung, S. K. *Calibration-free Approach to 3D Reconstruction Using Light Stripe Projections on a Cube Frame*. IEEE, pp.13-19; 2001.
- Coma, O.; Mascle, C., Véron, P. *Geometric and form feature recognition tools applied to a design for assembly methodology*. Computer-Aided Design, n.35, PP.1193-1210, 2003.
- Devernay, F.; Faugeras, O. *From projective to Euclidian Reconstruction*. Proceedings of Int. Conference on Computer Vision & Pattern Recognition, pp.264-269, 1996.
- Facon, Jacques; *Morfologia Matemática: Teoria e Exemplos*. CITS, 1996.
- Faugeras, O.; Mourrain, B. *About the correspondence of points between N images*. IEEE, pp. 37-44, 1995
- Faugeras, O.; Quan, L.; Strum, P. *Self-Calibration of a 1D Projective Camera and Its Application to the Self-Calibration of a 2D Projective Camera*. IEEE, Transactions on Pattern Analysis and Machine Intelligence, vol 22, n.10, 2000.
- Fiore, P. D. *Linear Epipolar Algorithm for Multiframe Orientation*. IEEE, pp.557-560, 2000;
- Floriani, Leila de; Puppo, Enrico. *Constrained Delaunay Triangulation for Multiresolution Surface Description*. Istituto per la Matematica Applicata – C.N.R.; IEEE 1988;

- Fonseca, José Manuel. *Sistemas Sensoriais*. <http://www-ssdp.dee.fct.unl.pt> consultado em 03/02/2004.
- Forsyth, D. A.; Ponce, J. *Computer Vision. A Modern Approach*. Prentice Hall, 2003.
- Galo, Mauricio; Tozzi, Clésio L. *The Concept of Matching Parallelepiped and its use in the Correspondence Problem*. IEEE, pp.410-414, 1999.
- Gasteratos, A.; Andreidis, I. *Non-linear image processing in hardware*. Pattern Recognition, vol. 33, pp. 1013-1021, 2001.
- Gonzalez, Rafael C.; Woods, Richard E. *Digital Image Processing, 2/E*. Prentice Hall, 2002.
- Gordon, R.; Ranganyyan, R. M. *Feature enhancement of film mamograms using fixed and adaptive neighbourhoods*. Appl. Optics, vol. 23, no. 4, pp. 560-564, 1984.
- Ha, J.E.; Kweon, I. S. *3D structure recovery and calibration under varying intrinsic parameters using known angles*. Pattern Recognition, 34, pp. 351-359, 2001.
- Hahn, Joseph M. *Lecture Notes for PHY 405 Classical Mechanics*. Department of Astronomy & Physics; Saint Mary's University, 2004.
- Hasan, Y. M. Y.; Karam, L. J. *Morphological reversible contour representation*. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22, no. 3, pp. 227-240, 2000.
- Heikkilä, J. ; Silvén, O. *A Four-step Camera Calibration Procedure with Implicit Image Correction*. Infotech Oulu and Department of Electrical Engineering. University of Oulu, Finland.
- Jain, Ramesh; Kasturi, Rangachar; Schunck, Brian G. *Machine Vision*. McGraw-Hill Series in Computer Science, 1995.

- Jelinek, D.; Taylor, C. J. *Reconstruction of Linearly Parameterized Models from Single Images with a camera of Unknown Focal Length*. IEEE transactions on Pattern Analysis and Machine Intelligence, vol. 23, n.7, pp. 767-773, 2001.
- Kurka, P. R. G.; Rudek, M.; *3-D Image Signal Processing For Automates Operations Using Range Cube*. XI DYNAMINE, 2005.
- Kurka, P. R. G.; Rudek, M.; *3-D Volume and Position Recovering Using a Virtual Reference Box*. IEEE Transactions on Image Processing, (draft) 2006.
- Kwon, Young-Hoo. *Inertia Tensor*. <http://www.kwon3d.com>, consultado em 26/03/2006.
- Lorensen, Willian; Cline, Harvey E. *Marching Cubes: A High resolution 3D surface Construction Algorithm*. Computer Graphics, vol.21, n.04, 1987.
- Luong, Q. T.; Faugeras, O. D. *Self-calibration of a camera using multiple images*. IEEE, pp.9-12, 1992.
- Ma Y., S. Soatto, J. Kosecká, S. S. Sastry, *An Invitation to 3-D Vision From Images to Geometric Models*. Series: Interdisciplinary Applied Mathematics, Vol. 26, Springer. 526 p., 2004.
- Majchrzak, Daniel; Sarkar, Sudeep. *Parallelizing Motion Segmentation by Perceptual Organization of XYT*. IEEE- Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04), 2004.
- Mohr, Roger; Triggs, Bill. *Projective Geometry for Image Analysis*. SPRS, 1996.
- Motta, J.M.S.T.; Mc Master, R. S. *Experimental Validation of a 3-D Vision-Based Measurement System Applied to Robot Calibration*. Journal of the Braz. Soc. Mechanical Sciences, vol. XXIV, p. 234-238, July 2002.

- Nielson, Gregory M. *On Marching Cubes*. IEEE Transactions on Visualization and Computer Graphics, vol.9, n.3, 2003.
- Noll, Valdir. *Desenvolvimento de Uma Câmera Digital CMOS para Uso em Estabilização Ativa na Área de Holografia Eletrônica*. Tese de Doutorado, UFSC, Florianópolis, 2003.
- Oisel, L.; et al. *One Dimensional Dense Disparity Estimation for Three-Dimensional Reconstruction*. IEEE Transactions on Image Processing, vol 12, no.9, 2003.
- Oppenheim, A. V.; Shafer, R. W. *Discrete-time signal processing*. Englewood Cliffs, NJ: Prentice Hall, 1989.
- Orth, A. *Desenvolvimento e Implementação de um Sistema de Reconhecimento Automático de Peças Mecânicas em uma Célula Flexível de Manufatura*. Projeto RAP. Laboratório de Automação Industrial, UFSC. Congresso da Sociedade Brasileira de Computação, SBC 1998.
- Quak, E.; Weyrich, N. *Decomposition and reconstruction algorithms for spline wavelets on a bounded interval*. Applied and Computational Harmonic Analysis, vol. 1, no. 3, pp. 217-231, 1994.
- Rogers, D. F.; Adams, J.A.; *Mathematical Elements for Computer Graphics*. McGraw-Hill, 1976.
- Rudek, M. *Posicionamento e Dimensionamento de Sólidos numa Linha de Montagem Através de Análise de Imagens*. Projeto de Doutorado, 2001.
- Ryoo, Seung Taek. *Segmentation Based Environment Modeling Using a Single Image*. International Conference, ICIAR, Proceedings, Part I, p.98-105, 2004.
- Salvi, J.; Armangué, X.; Battle, J. *A Comparative Review of Camera Calibrating Methods With Accuracy Evaluation*. Pattern Recognition, 35, 1617-1635, 2002.

- Shen, D.; Davatzikos, C. *An adaptive-focus deformable model using statistical and geometric information*. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp. 906-912, 2000.
- Shin, D.; Kurfess, T. R. *Three-dimensional metrology of surface extracted from a cloud of measured points using a new point-to-surface assignment method An application to PCB-mounted solder pastes*. Precision Engineering, n.28, pp.302-313, 2004.
- Soares, Marco. *Propriedades de seções planas*. <http://myspace.eng.br/eng/secplan.asp>, consultado em 26/03/2006.
- Spitz, S. N.; Rechicha, A. A. G. *Accessibility analysis using computer graphics hardware*. IEEE Trans. on Visualization and Computer Graphics, vol. 6, no. 3, pp. 208-219, 2000.
- Tommaselli, Antonio M. G.; Tozzi, Clésio L. *Line Based Camera Calibration in Machine Vision Dynamic Applications*. SBA Controle e Automação, vol.10, n.2, 1999.
- Trucco, E; Verri, A. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, 1998.
- Tsai, R.; Huang, T. *Uniqueness and estimation of three dimensional motion parameters of rigid objects with curved surface*. IEEE Trans. Pattern Anal. Machine Intell., vol 6, pp.13-26, 1984.
- Twardowski, Tomasz; Cyganek, B.; Borgoz, J. *Gradient Based Dense Stereo Matching*. ICIAR 2004, pp.721-728, 2004.
- Urbanek, M.; Horaud, R.; Sturm. P. *Combining Off- and On-line Calibration of a Digital Camera*. IEEE, pp.99-106, 2001.
- Venturi, Jacir J. *Álgebra Vetorial e Geometria Analítica*. 6.a Ed. Curitiba, 1988.

- Wang, R.-Z.; Lin, C.-F.; Lin, J.-C. *Image hiding by optimal LSB substitution and genetic algorithm*. Pattern Recognition, vol. 34, pp. 671-683, 2001.
- Wang, Wen-Ge; Hu, Zhang-Yi, Shun, Zong-Yu. *3-D Shape Measurement based Computer Vision*. Proceedings of the First International Conference on Machine Learning and Cybernetics, pp.910-914, 2002
- Xie, Jun; Tat Tsui, Hung. *Wide Baseline Stereo Matching by Corner-Edge-Regions*. ICIAR 2004, pp. 713-720, 2004.
- Yuan, J.S.; Fitzsimons, C.F. *A Mesh Generator for Tetrahedral Elements Using Delaunay Triangulation*. IEEE Transactions on Magnetics, vol.29, n.2, pp.1906-1909, 1993.
- Zhang, B.; Fu, M.; Yan, H. *A nonlinear neural network model of mixture of local principal component analysis: application to handwritten digits recognition*. Pattern Recognition, vol. 34, pp. 203-214, 2001.
- Zhang, Quan-Bing; Wang, Hai-Xian, Wei, Sui. *A New Algorithm for 3D Projective Reconstruction Based on Infinite Homography*. Proceedings of the Second International Conference on Machine Learning and Cybernetics, Xi'an, pp.2882-2886, 2003.
- Zhang, Zhengyou. *A Flexible New Technique for Camera Calibration*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22. N.11, pp.1330-1334, 2000
- Zhang, Zhengyou. *Estimating Motion and Structure from Correspondences of Line Segments Between Two Perspectives Images*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.17, n.12, pp.1129-1139, 1995.

Anexo I

Características das Imagens Digitais

Conforme introduzido no capítulo 2, as imagens digitais possuem características diferenciadas em função da representação de intensidades ou cores. Este anexo ilustra a forma com que cada tipo de imagem associa seus *pixels* a um determinado valor de intensidade ou de cor. A notação adotada para intensidade é o símbolo l (*level*).

1. Imagem Binária:

Como descrito no capítulo 2, uma imagem binária armazena em seus *pixels* os valores $l=0$ para preto e $l=1$ para branco, sem nenhum nível intermediário de intensidade. A figura 1 apresenta uma imagem de exemplo $A_{147 \times 47}$ em preto e branco. A região destacada de A forma uma matriz de valores zero e um, conforme o exemplo da figura 2.

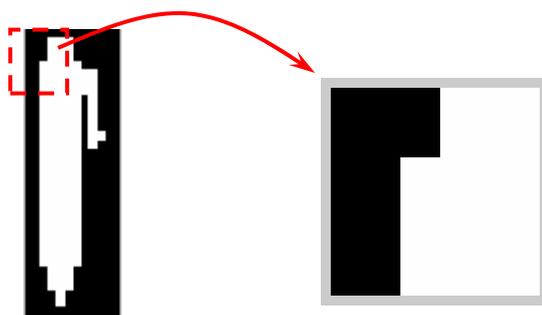


Figura 1: Exemplo de imagem binária $A_{147 \times 47}$.

```

0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0
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 1
0 0 1 1 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1

```

Figura 2: Valores dos *pixels* para a imagem de exemplo $A_{147 \times 47}$ normalizada.

A figura 2 apresenta a parte em destaque da imagem exemplo, a fim de ilustrar a composição da matriz de 0s e 1s que forma a imagem binária. A imagem foi normalizada para o intervalo $[0,1]$. Para uma imagem não normalizada, com 8 bits por *pixel*, a cor preta teria o valor 0 e o branco o valor 255.

2. Imagens em níveis de cinza

Uma imagem em níveis de cinza possui valores de intensidade $0 \leq l \leq l_{\text{máx}}$, onde $l_{\text{máx}}$ depende do número de bits por *pixel*. Numa imagem normalizada, como descrito no capítulo 4, o valor da intensidade estará compreendido no intervalo $0 \leq l \leq 1$. A figura 3 apresenta um exemplo de imagem $A_{147 \times 47}$ em níveis de cinza.

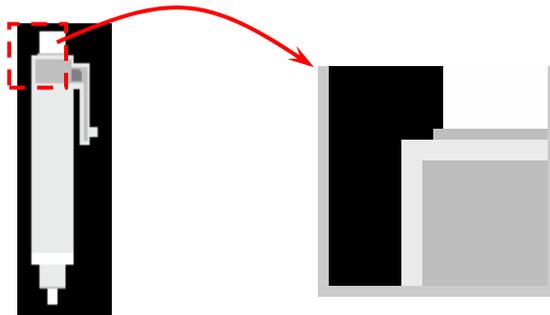


Figura 3: Exemplo de imagem em níveis de cinza.

0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	0	255	255	255	255	255	255	255	255	255
0	191	191	191	191	191	191	191	191	191	191
235	235	235	235	235	235	235	235	235	235	235
235	235	235	235	235	235	235	235	235	235	235
191	191	191	191	191	191	191	191	191	191	191
191	191	191	191	191	191	191	191	191	191	191

Figura 4. Valores dos *pixels* para a imagem de exemplo em níveis de cinza, não normalizada.

Numa imagem não normalizada de 8 bits por *pixel*, existem 256 níveis discretos de intensidade, variando do valor 0 do preto ao valor 255 do branco. A figura 4 apresenta um segmento da matriz de intensidades da imagem exemplo.

3. Imagens Indexadas

Uma imagem indexada é uma imagem cujos valores dos *pixels* são o índice para um mapa de cores. Um arquivo de imagem deste tipo sempre armazena os valores dos *pixels* e seu respectivo mapa de cores.

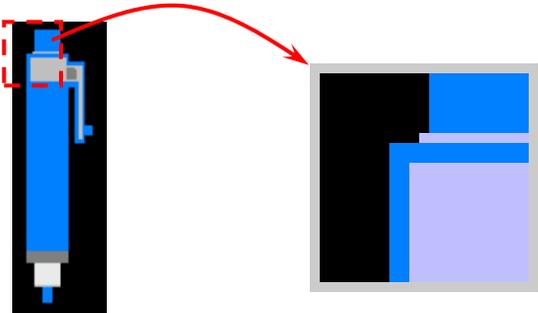


Figura 5: Exemplo de imagem colorida indexada.

A figura 5 apresenta o exemplo de uma imagem de 256 cores. O esquema de indexação é apresentado pela figura 6, onde cada valor de *pixel* se refere a uma linha no mapa de cores.

pixels da imagem											mapa de cores		
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	19	19	19	19	19	19	19	19	19	0	0.1255	0
0	0	19	19	19	19	19	19	19	19	19	0	0.1255	0.2510
0	0	19	19	19	19	19	19	19	19	19	0	0.1255	0.5020
0	0	19	19	19	19	19	19	19	19	19	0	0.1255	1.0000
0	0	19	19	19	19	19	19	19	19	19	0	0.2510	0
0	0	19	19	19	19	19	19	19	19	19	0	0.2510	0.2510
0	0	19	19	19	19	19	19	19	19	19	0	0.2510	0.5020
0	0	19	19	19	19	19	19	19	19	19	0	0.2510	1.0000
0	0	19	19	19	19	19	19	19	19	19	0	0.3765	0
0	0	19	19	19	19	19	19	19	19	19	0	0.3765	0.2510
0	0	19	19	19	19	19	19	19	19	19	0	0.3765	0.5020
0	219	219	219	219	219	219	219	219	219	219	0	0.3765	1.0000
19	19	19	19	19	19	19	19	19	19	19	0	0.5020	0
19	19	19	19	19	19	19	19	19	19	19	0	0.5020	0.2510
219	219	219	219	219	219	219	219	219	219	219	0	0.5020	0.5020
219	219	219	219	219	219	219	219	219	219	219	0	0.5020	1.0000

Figura 6: Valores dos *pixels* para um fragmento da imagem exemplo representando os índices para um mapa de cores.

Observa-se pela figura 6 que um determinado valor de *pixel* na matriz de imagem com $l=19$, corresponde a linha 20 no mapa de cores, com a contagem de linhas começando em 0. O mapa de cores completo para esta imagem é formado por 256 linhas, uma para cada nível de intensidade do *pixel*, e por 3 colunas. Cada coluna se refere a uma cor primária, sendo elas o vermelho, o verde e o azul (*Red, Green, Blue*). A combinação destes 3 valores da linha forma a cor correspondente àquele *pixel*.

4. Imagens RGB

Uma imagem RGB ou *truecolor* possui os valores das cores primárias, vermelho, verde e azul associadas diretamente ao *pixel* da imagem. Para isto são necessários três planos de imagens, compondo uma matriz de dimensões $M \times N \times 3$. Cada plano de imagem contém os valores de *pixels* referentes a uma cor primária RGB (*Red, Green, Blue*).

Pelo observado na figura 8, verifica-se que a imagem RGB do exemplo $A(i,j)$ com $i=16$ e $j=2$, possui como valor do *pixel*, a combinação dos três planos onde $l = [255,192,192]$. A combinação destes três valores fornecem a cor do ponto, segundo a representação do espaço de cores RGB , apresentado na figura 9.

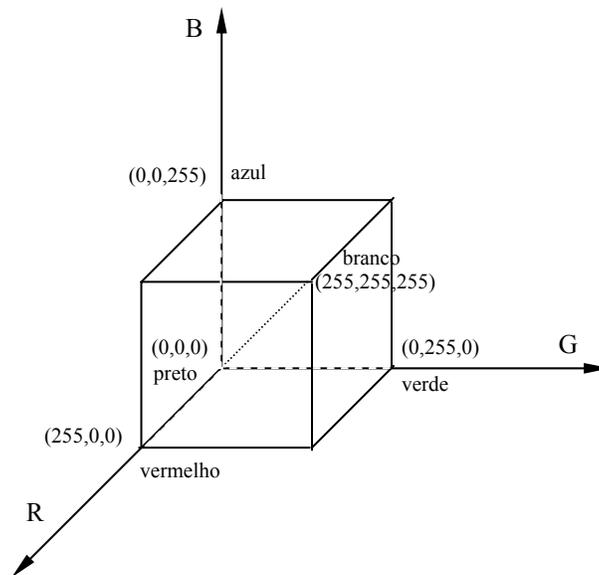


Figura 9: Representação do cubo de espaço de cores RGB .

Pela figura 9, verifica-se que qualquer posição do cubo pode ser mapeada como uma função distância entre a origem e as coordenadas (R,G,B) . A diagonal do cubo representa os níveis de cinza, para $R=G=B$. Maiores detalhes a respeito de processamento de imagens com cores podem ser obtidas de Gonzalez (2002).

Anexo II

Morfologia Matemática

Segundo Facon (1996) o princípio básico da morfologia matemática consiste em extrair uma informação relativa à geometria de um conjunto desconhecido de pontos, pertencentes a uma imagem, através de uma transformação a partir de um outro conjunto bem definido, chamado elemento estruturante. A partir do elemento estruturante, pode-se quantificar de que maneira ele faz parte da imagem.

1. Operadores Morfológicos Básicos

O elemento é uma imagem normalmente representada pelo símbolo \mathbf{B}_x , (ou simplesmente \mathbf{B}) que significa o elemento estruturante B , centrado no *pixel* x . A forma do elemento estruturante, é dada pela marcação dos pontos relevantes “•” e não relevantes “.”, como abaixo:

$$B = \begin{Bmatrix} \cdot & \bullet & \cdot \\ \bullet & \bullet & \bullet \\ \cdot & \bullet & \cdot \end{Bmatrix}$$

O ponto central do elemento estruturante é seu centro físico. Caso esteja em outra posição, o ponto central deverá ser marcado propriamente. Somente os *pixels* relevantes são considerados nas operações morfológicas..

Uma imagem qualquer é representada por um conjunto de pontos A , como o exemplo abaixo:

$$A = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \bullet & \bullet & \cdot & \bullet & \cdot & \cdot \\ \cdot & \bullet & \bullet & \bullet & \cdot & \bullet & \cdot \\ \cdot & \cdot & \bullet & \bullet & \bullet & \bullet & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix}$$

Nos exemplos do anexo o valor 1 representa o símbolo “•” e o valor 0 o símbolo “.”

2. Erosão Binária

Definição 1: A erosão binária de um conjunto A pelo elemento estruturante B , é:

$$\text{Ero}^B(A) = A \text{ ero } B$$

O elemento estruturante deve deslizar sobre a imagem A . A imagem contém informação sobre o fundo (“0” Informação irrelevante) e sobre os objetos da imagem (“1” Informação relevante). A definição acima significa que o elemento B_x , centrado no *pixel* x de A , deve emparelhar-se com a vizinhança de x , ou seja, B deve estar contido no conjunto de pontos da vizinha de x .

O exemplo a seguir descreve passo a passo um processo de erosão binária. Considere o ponto central do elemento estruturante no centro geométrico de B . O ponto central do elemento estruturante serve para definir a posição do resultado de A com B .

1.a iteração:

$$A = \begin{bmatrix} 00000 \\ 01110 \\ 01110 \\ 01110 \\ 00000 \end{bmatrix} \text{ ero } B = \begin{Bmatrix} 010 \\ 010 \\ 010 \end{Bmatrix} \Rightarrow \begin{bmatrix} 010 \\ 010 \\ 010 \end{bmatrix} \not\subset \begin{Bmatrix} 000 \\ 011 \\ 011 \end{Bmatrix} = \begin{bmatrix} 00000 \\ 00000 \\ 00000 \\ 00000 \\ 00000 \end{bmatrix}$$

Como B não se encaixa (não está contido) na vizinhança de x , o pixel resultante é marcado como irrelevante.

2.a e 3.a iteração:

$$A = \begin{bmatrix} 00000 \\ 01110 \\ 01110 \\ 01110 \\ 00000 \end{bmatrix} \text{ ero } B = \begin{Bmatrix} 010 \\ 010 \\ 010 \end{Bmatrix} \Rightarrow \left[\begin{Bmatrix} 010 \\ 010 \\ 010 \end{Bmatrix} \subset \begin{Bmatrix} 000 \\ 011 \\ 011 \end{Bmatrix} \right] = \begin{bmatrix} 00000 \\ 00000 \\ 00000 \\ 00000 \\ 00000 \end{bmatrix}$$

4.a iteração:

$$A = \begin{bmatrix} 00000 \\ 01110 \\ 01110 \\ 01110 \\ 00000 \end{bmatrix} \text{ ero } B = \begin{Bmatrix} 010 \\ 010 \\ 010 \end{Bmatrix} \Rightarrow \left[\begin{Bmatrix} 010 \\ 010 \\ 010 \end{Bmatrix} \subset \begin{Bmatrix} 011 \\ 011 \\ 011 \end{Bmatrix} \right] = \begin{bmatrix} 00000 \\ 00000 \\ 01000 \\ 00000 \\ 00000 \end{bmatrix}$$

Como B se encaixa (está contido) na vizinhança de x , o pixel resultante é marcado como relevante.

N-ésima iteração:

$$A = \begin{bmatrix} 00000 \\ 01110 \\ 01110 \\ 01110 \\ 00000 \end{bmatrix} \text{ e } B = \begin{Bmatrix} 010 \\ 010 \\ 010 \end{Bmatrix} \text{ onde } A \text{ ero } B = \begin{bmatrix} 00000 \\ 00000 \\ 01110 \\ 00000 \\ 00000 \end{bmatrix}$$

Definição 2: A erosão binária de um conjunto A pelo elemento estruturante B , é:

$$\text{Ero}^B(A) = A \text{ ero } B = A \theta \tilde{B}$$

Por esta definição, o conjunto a ser erodido A , é deslocado em função das posições permitidas pelo elemento estruturante B . Agora translada A e não mais o B . Os deslocamentos são realizados em relação ao ponto central de \tilde{B} .

O exemplo a seguir demonstra a operação de erosão pela definição 2.

$$A = \begin{bmatrix} 00000 \\ 01110 \\ 01110 \\ 01110 \\ 00000 \end{bmatrix} \text{ ero } \tilde{B} = \left\{ \begin{bmatrix} 010 \\ 010 \\ 010 \end{bmatrix} \right\} = \begin{bmatrix} 00000 \\ 01110 \\ 01110 \\ 01110 \\ 00000 \end{bmatrix} \cap \begin{bmatrix} 01110 \\ 01110 \\ 01110 \\ 00000 \\ 00000 \end{bmatrix} \cap \begin{bmatrix} 00000 \\ 00000 \\ 01110 \\ 01110 \\ 01110 \end{bmatrix} = \begin{bmatrix} 00000 \\ 01110 \\ 01110 \\ 00000 \\ 00000 \end{bmatrix}$$

2.1. Efeitos da Erosão.

1. A erosão modifica o conjunto A , que fica menor em todos os casos.
2. O elemento estruturante elimina os conjuntos inferiores a ele em tamanho.
3. A erosão aumenta os furos interiores aos conjuntos.
4. Usando um elemento estruturante adequado em tamanho e orientação é possível eliminar ligações entre objetos.

3. Dilatação Binária

Definição 1: A dilatação binária pode ser definida como:

$$dil^B(A) = A \text{ dil } B$$

As operações de erosão e dilatação são duais, sendo a interpretação da dilatação complementar a interpretação da erosão. Ou seja, “ B_x está incluído em A ” e “a interseção de B_x e A não é vazia”. Segundo a definição acima, o elemento estruturante B deve percorrer a imagem A . O elemento estruturante, em cada posição, deve verificar uma possível interseção com a vizinhança de x . Caso ocorra a interseção, o ponto central na imagem resultado será marcada como relevante (1), caso contrário como irrelevante (0). O exemplo descreve o processo.

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{e} \quad B = \begin{Bmatrix} 0 \\ 1 \\ 1 \end{Bmatrix}$$

1.a iteração:

Os pontos relevantes de B_x , não tem interseção com os *pixels* correspondentes na vizinhança de x . Então na imagem resposta o ponto é marcado como irrelevante.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \text{ dil } \begin{Bmatrix} 0 \\ 1 \\ 1 \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Ponto irrelevante porque B_x não tem interseção com o conjunto X .

2.a iteração:

Os pontos relevantes de B_x têm interseção com os *pixels* correspondentes na vizinhança de x , em pelo menos um ponto. Então na imagem resposta o ponto é marcado como relevante.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \text{dil} \begin{Bmatrix} 0 \\ 1 \\ 1 \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Ponto relevante porque B_x tem interseção com o conjunto A .

N-ésima iteração:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \text{dil} \begin{Bmatrix} 0 \\ 1 \\ 1 \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Imagem de resposta de $A \text{ dil } B$.

Definição 2: A dilatação binária pode ser definida também como:

$$\text{dil}^B(A) = A \text{ dil } B = A \oplus \tilde{B}$$

Pela definição 2, A é deslocado em função das posições permitidas pelo elemento estruturante \tilde{B} . Os deslocamentos são realizados em relação ao ponto central de \tilde{B} . A figura 1 e a figura 2 apresentam um exemplo do processo.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \text{dil} \begin{Bmatrix} \tilde{1} \\ 1 \\ 0 \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \cup \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figura 1. Exemplo de dilatação com elemento estruturante não simétrico.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \text{dil} \begin{Bmatrix} \tilde{1} \\ 0 \\ 1 \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix} \cup \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

Figura 2. Exemplo de dilatação com elemento estruturante simétrico.

Observa-se que no exemplo da figura 2, o elemento central de Bx é irrelevante, portanto o conjunto A não entra na união dos conjuntos A transladados.

3.1. Efeitos da Dilatação:

1. A dilatação permite aumentar partículas da imagem;
2. Também permite o preenchimento pequenos buracos;
3. Conecta partículas de imagem que estão próximas;

Para uma referência completa sobre estes processos morfológicos, deve-se consultar Facon (1996) e Gonzalez (2000).

Anexo III

Decomposição em Valores Singulares (SVD)

A utilização da decomposição em valores singulares (SVD) aparece diversas vezes no texto, como na obtenção da matriz de rotação e translação a partir da matriz essencial, ou para obtenção dos componentes principais a partir de uma matriz de covariância. Este anexo procura dar uma visão geral sobre esta ferramenta amplamente utilizada em visão computacional. Maiores detalhes sobre SVD podem ser obtidos em Castlemam (1996), Truco (1998), Forsyth (2003) e Ma (2004). A decomposição em valores singulares é útil para solução de muitos problemas como, inversão de matrizes, cálculo de rank, estimação de mínimos quadrados e projeções.

A teoria sobre SVD diz que qualquer matriz \mathbf{A} de dimensão $N \times N$ pode ser expressa como,

$$\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T \quad (1)$$

Onde as colunas de \mathbf{U} e \mathbf{V} são os autovetores de $\mathbf{A}\mathbf{A}^t$, e \mathbf{S} é uma matriz diagonal de dimensão $N \times N$ que contém os valores singulares de \mathbf{A} distribuídos na sua diagonal. Os elementos da diagonal chamados de valores singulares são descritos da forma,

$$S = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \end{bmatrix} \quad \text{com, } \sigma_1 \geq \sigma_2 \geq \dots \sigma_n \geq 0 \quad (2)$$

Os valores singulares fornecem informação sobre a singularidade de uma matriz quadrada. Uma matriz quadrada é não singular se todos os seus valores singulares são diferentes de zero. O valor de σ_i descreve o quão singular é a matriz \mathbf{A} . A razão,

$$C = \frac{\sigma_1}{\sigma_n} \quad (3)$$

É chamada número de condição e mede o grau de singularidade de \mathbf{A} . A questão dos autovalores e autovetores também é frequentemente abordada no texto. A seguir são apresentadas as suas definições formais. As propriedades vinculadas a autovalores e autovetores são amplamente discutidas nas referências anteriormente citadas.

Autovalores: Para uma matriz \mathbf{A} de dimensão $N \times N$ existem N escalares λ_k com $k=1, \dots, N$ da forma,

$$|\mathbf{A} - \lambda_k \mathbf{I}| = 0 \quad (4)$$

Onde λ_k é o único conjunto de valores característicos (autovalores) da matriz \mathbf{A} . cada autovalor pode ser interpretado como um valor que, ao ser subtraído de cada elemento da diagonal torna a matriz singular. O *rank* de uma matriz é o número de autovalores não nulos.

Autovetores: O vetor \mathbf{v}_k de dimensão $N \times 1$ que satisfaz a igualdade,

$$\mathbf{A} \mathbf{v}_k = \lambda_k \mathbf{v}_k \quad (5)$$

É chamado de autovetor ou vetor característico de \mathbf{A} . Existe um autovetor correspondente a cada um dos autovalores.

A importância da decomposição em valores singulares é dada para a solução de sistemas de equações lineares não homogêneas, ou sistemas de equações lineares homogêneas de *rank* deficiente, e também garante a ortogonalidade de uma matriz estimada numericamente.