

Universidade Federal do Maranhão
Centro de Ciências Exatas e Tecnologia
Curso de Pós-Graduação em Engenharia de Eletricidade

*METODOLOGIA SEMI-AUTOMÁTICA
PARA RECONSTRUÇÃO 3D DE SÓLIDOS
GEOMÉTRICOS BASEADA EM IMAGEM*

Irlandino Oliveira Almeida

São Luís
2007

Universidade Federal do Maranhão
Centro de Ciências Exatas e Tecnologia
Curso de Pós-Graduação em Engenharia de Eletricidade

*METODOLOGIA SEMI-AUTOMÁTICA
PARA RECONSTRUÇÃO 3D DE SÓLIDOS
GEOMÉTRICOS BASEADA EM IMAGEM*

Irlandino Oliveira Almeida

Dissertação apresentada ao Curso de Pós-Graduação em Engenharia de Eletricidade da UFMA como parte dos requisitos necessários para obtenção do grau de Mestre em Engenharia Elétrica.

São Luís
2007

Almeida, Irlandino Oliveira

Metodologia semi-automática para reconstrução 3D de sólidos geométricos baseada em imagem / Irlandino Oliveira Almeida. - São Luís, 2007.

56f.:il.

Dissertação (Mestrado em Engenharia de Eletricidade) - Centro de Ciências Exatas e Tecnologia, Universidade Federal do Maranhão, 2007.

1. Modelagem baseada em imagens 2. Calibração de câmeras 3. *Template matching*. I. Título.

CDU 004

METODOLOGIA SEMI-AUTOMÁTICA PARA RECONSTRUÇÃO 3D DE SÓLIDOS GEOMÉTRICOS BASEADA EM IMAGEM

Irlandino Oliveira Almeida

Aprovado em

BANCA EXAMINADORA

Prof. Dr. Anselmo Cardoso Paiva

Doutor em Informática (UFMA)

Orientador

Prof. Dr. Aristófanés Corrêa e Silva

Doutor em Informática (UFMA)

Co-orientador

Prof. Dr. Paulo Cezar Pinto Carvalho

Doutor em Informática (PUC-Rio)

Examinador Externo

Prof. Dr. Alexandre César Muniz de Oliveira

Doutor em Computação Aplicada (UFMA)

Examinador Interno

”Tudo tem seu tempo e até certas manifestações mais vigorosas e originais entram em voga ou saem de moda. Mas a sabedoria tem uma vantagem: é eterna”.

Baltasar Gracián

Dedicatória

Aos meus pais, José Ribamar Araújo Almeida e Rosina Oliveira Almeida(*In Memoriam*);

aos meus irmãos, Israel, Iran, Itaan, e Leude;

à minha irmã Regiane;

à minha avó, Simplícia(*In Memoriam*);

as minhas amigas Aline, Clissiane, Francislene, Helaine, Joelma e Lindonete;

aos amigos Carlos, Ricardo, Ronald e Osvaldo;

e a Miguel Mohana.

Agradecimentos

Em primeiro lugar, a Deus.

Ao meu orientador, Prof^o. Dr. Anselmo Cardoso Paiva, e ao meu Co-orientador, Prof^o. Dr. Aristófanês Corrêa e Silva, pela orientação segura, amizade e companheirismo durante a realização deste trabalho.

Aos amigos do LABPAI, em especial Thamila e Maneco, pelo companheirismo nos momentos difíceis deste trabalho.

A todos que colaboraram de maneiras diversas para a realização deste trabalho.

RESUMO

Apresentamos uma metodologia semi-automática para reconstrução tridimensional utilizando um ambiente de modelagem no qual o usuário, a partir de uma determinada imagem e de pontos nela selecionados com seu correspondente no espaço do mundo, realiza a calibração da câmera. Após este processo, a cena poderá ser composta a partir de objetos pré-estabelecidos fornecidos pelo ambiente de modelagem tridimensional baseado na imagem. Esta metodologia pode ser aplicada na reconstrução de modelos do mundo real, como instalações elétricas e industriais.

Palavras-Chave: Modelagem baseada em imagens, Calibração de câmeras, *template matching*.

ABSTRACT

We present a semi-automatic methodology for three-dimensional reconstruction using an image-based modeling environment. This is done through an initial camera calibration step that makes possible to the viewer, identify objects in the 2D acquired image and then get its position from the camera parameters. This methodology can be applied in the real word model reconstruction like electric and industry installations.

Keywords: image-based modeling, camera calibration, template matching.

Lista de Figuras

1.1	Ambiente de Modelagem do Facade	9
1.2	Processo de reconstrução feito pelo Juiz Virtual.	9
1.3	Projeto Bamiyan.	10
1.4	Reconstrução de face humana.	11
2.1	Computação gráfica tradicional. Fonte: (Carvalho 2002)	14
2.2	Evolução do processo de geração de imagens.Fonte: (Carvalho 2002)	14
2.3	Uma câmera virtual observando o panorama. Fonte: (Carvalho 2002)	15
2.4	Câmera <i>pinhole</i> e projeção perspectiva. Fonte: (Velho <i>et al.</i> 2005)	18
2.5	Determinação da projeção perspectiva.Fonte: (Carvalho <i>et al.</i> 1998)	19
2.6	Rotação entre dois referenciais.	21
2.7	Translação entre dois referenciais.	21
2.8	Transformação de um ponto P no mundo real para um ponto p no plano da imagem. Fonte: (Carvalho <i>et al.</i> 1998)	25
2.9	Imagens obtidas com diferentes orientações onde os pontos dos vértices dos quadrados são utilizados como parâmetros de entrada ao processo. Fonte: (Zhang 2000)	27
3.1	Cálculo da distância Hausdorff.	36
4.1	Etapas do processo de reconstrução 3D	38
4.2	Processo de criação do <i>template</i> com fecho convexo.	42
4.3	Translação de um cubo para a origem.	42
4.4	Posicionamento de um cubo ao longo do plano do modelo, para geração de 5 <i>templates</i> de diferentes tamanhos.	43
4.5	Janela de captura de pontos de referência cena-imagem.	46

4.6	Etapa de fornecimento do conjunto de pontos de referência para efetuar a calibração.	46
4.7	Barra de ferramentas com os objetos componentes do modelo. . .	47
4.8	Janelas de Visualização 3D.	47
4.9	Modelo <i>wireframe</i> de um dos objetos componentes do modelo. . .	48
4.10	Fornecimento dos pontos do objeto e reconstrução de um componente selecionado.	48
4.11	Etapa de detecção dos vértices dos objetos nas bordas da imagem.	49
4.12	Obtenção dos locais de <i>matching</i>	50
4.13	Reconstrução finalizada.	50

Lista de Abreviaturas e Siglas

CAD	Computer Aided Design(Projeto Auxiliado por Computador)
IHM	Interação Homem-Máquina
RV	Realidade Virtual
SCC	Sistema de Coordenadas da Câmera
SCI	Sistema de Coordenadas da Imagem
SCM	Sistema de Coordenadas do Mundo
SCP	Sistema de Coordenadas em <i>Pixel</i>

Sumário

1	INTRODUÇÃO	6
1.1	Objetivos	8
1.2	Trabalhos Relacionados	8
1.3	Organização do Trabalho	12
2	MODELAGEM BASEADA EM IMAGEM	13
2.1	Introdução	13
2.2	Fotografia 3D	15
2.2.1	Reconstrução Tridimensional	17
2.3	Modelo de Câmera	18
2.4	Calibração de câmara	24
2.4.1	Método de Tsai	25
2.4.2	Método de Zhang	26
3	PROCESSAMENTO DE IMAGENS	28
3.1	Introdução	28
3.2	Detecção de Bordas	29
3.3	Detecção de vértices	31
3.4	Fecho Convexo	33
3.5	Template Matching	34
4	METODOLOGIA PARA RECONSTRUÇÃO 3D DE SÓLIDOS GEOMÉTRICOS	37
4.1	Etapas do processo de reconstrução	38
4.1.1	Aquisição	40
4.1.2	Detecção de bordas e dos vértices	40

4.1.3	Calibração	40
4.1.4	Construção de <i>templates</i>	41
4.1.5	<i>Matching</i>	43
4.1.6	Visualização tridimensional	44
4.2	Ambiente Integrado de Modelagem	45
4.2.1	Localização dos pontos dos objetos pelo usuário	45
4.2.2	Localização automática dos pontos dos objetos	48
5	CONCLUSÃO	51
	REFERÊNCIAS BIBLIOGRÁFICAS	54

CAPÍTULO 1

INTRODUÇÃO

Ao decorrer do desenvolvimento da humanidade, o homem sempre procurou por formas mais claras de expressão. A pintura em cavernas na pré-história era uma forma de relatar acontecimentos decorrentes da caça e do cotidiano da vida daquela sociedade (Sherman and Craig 2003). O surgimento da escrita, anos mais tarde, foi a resposta do homem à dificuldade de documentar a expressão oral.

Com a invenção dos primeiros computadores, a sociedade mundial viu nascer a era digital com a proposta de facilitar e auxiliar na realização das tarefas do cotidiano nas mais diversas áreas do conhecimento como engenharia, mecânica, biologia e geografia, dentre outras.

Em busca de melhoria na comunicação usuário/computador, avançou-se no campo da Interação Homem-Máquina (IHM) com interfaces que podiam oferecer a possibilidade de representação da informação de um modo realístico, organizando-a de uma maneira espacial (Osório 2004). Este tipo mais avançado de interface é chamado de Realidade Virtual (RV), na qual um usuário pode navegar, imergir e interagir em um ambiente sintético tridimensional, de forma natural e intuitiva, permitindo o uso de dispositivos multisensoriais (Kirner and Pinho 2006).

Graças aos investimentos feitos pela indústria no desenvolvimento de equipamentos específicos para Realidade Virtual, esta área sofreu um grande avanço e hoje áreas como Engenharia, Medicina e Geociências têm buscado soluções integradas com RV para melhor representação, percepção e análise do espaço físico envolvido (Corseuil 2003).

Para construir este tipo de *software*, é necessário o envolvimento de uma

grande equipe de profissionais que nem sempre obedecem aos prazos estabelecidos por causa de penosos processos de digitalização e modelagem de objetos tridimensionais. Além disso, para fazer a modelagem destes objetos, são utilizados *softwares* específicos de CAD para realizar a modelagem baseada em geometria que requerem profissionais altamente qualificados.

Para resolver este problema e melhorar o projeto de ambientes virtuais tridimensionais é que vêm surgindo técnicas que possam realizar a reconstrução de cenas 3D.

Atualmente, são empregados diversos métodos para a realização deste processo a partir de uma ou várias fotografias, mediante o uso de técnicas de modelagem baseada em imagem. Com as técnicas desta área, é possível automatizar de maneira considerável o processo de aquisição de imagens e geração de modelos geométricos. É possível, também, ao contrário do que ocorre com a modelagem baseada em geometria, refazer a reconstrução cada vez que for modificado algo nos objetos do mundo real, sem que para isso tenham que ser modelados novamente os objetos na cena.

Mais especificamente uma subárea da área de modelagem baseada em imagem é a Fotografia 3D. Fotografia 3D reúne um conjunto de técnicas das áreas de Computação Gráfica e Visão Computacional para aquisição de dados digitalizados que permite que seja recriado, no computador, o modelo digital de objetos 3D do mundo real (Velho *et al.* 2005).

Uma das aplicações direta destas técnicas é a geração de *as built* de prédios, instalações industriais, instalações elétricas, sítios arqueológicos, cidades históricas e etc.

A construção de *software* para visualização de cenas tridimensionais é um processo complexo e demorado. A escolha de ferramentas que ofereçam uma maior abstração na implementação deste tipo de software torna-se um fator decisivo para a obtenção de resultados satisfatórios em curto prazo. Além disso, as ferramentas devem ser capazes de facilitar a interação com o usuário final, possibilitando navegação dentro da cena, permitindo a simulação de situações reais.

1.1 Objetivos

O propósito deste trabalho é obter a localização no espaço 3D de objetos tridimensionais e fazer a reconstrução destes objetos a partir de imagens bidimensionais. Pretende-se trabalhar o problema da reconstrução de uma classe de modelos bem definida, permitindo a reconstrução dos objetos 3D, visando a obter a menor intervenção possível do usuário no que diz respeito ao fornecimento de algum tipo de dado ou parâmetros externos. Para isto, foi desenvolvido um modelo simplificado numa maquete composta de sólidos geométricos, tais como cubos e paralelepípedos.

1.2 Trabalhos Relacionados

Geografia, História, Patrimônio Histórico, Arquitetura e Engenharia são algumas das áreas em que a modelagem baseada em imagens vem sendo largamente utilizada.

(Debevec 1996), desenvolveu um sistema para modelagem e *rendering* de cenas arquitetônicas a partir de fotografias. Neste sistema, denominado *Facade*, os usuários constroem modelos 3D de uma cena do mundo real, especificando uma coleção de primitivas geométricas, fazendo a correspondência entre as bordas do modelo e as bordas marcadas na fotografia. O sistema determina a forma e a posição das primitivas de forma automática, sem que seja necessária a definição do tamanho destas formas pelo usuário, como pode ser visto na Figura 1.1.

Outra forma de aplicação deste tipo de modelagem está na área desportiva. Um exemplo pode ser encontrado no método descrito por (Carvalho *et al.* 1998), que desenvolveu uma técnica para integração da informação dada por uma imagem que contém pontos de referência em um sistema de modelagem geométrica. Baseado nesta metodologia, foi desenvolvido um ambiente para análise de lances de partidas de futebol chamado de Juiz Virtual (Szenberg 2001). Neste ambiente, é possível fazer a reconstrução tridimensional de uma determinada cena de uma partida de futebol obedecendo às dimensões reais da cena real que gerou a imagem, o que possibilita fazer medições (Figura 1.2). É utilizado, neste processo, o método de calibração de câmeras.

Sítios arqueológicos também têm sido reconstruídos com auxílio da modelagem

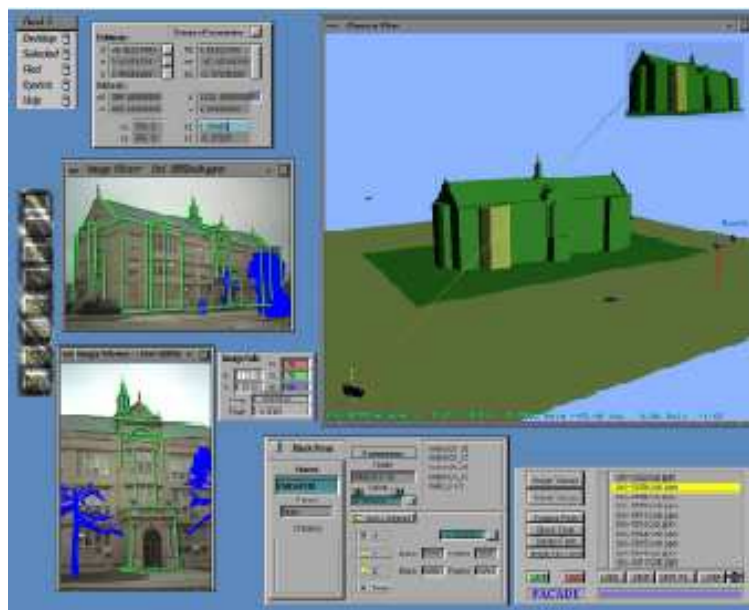
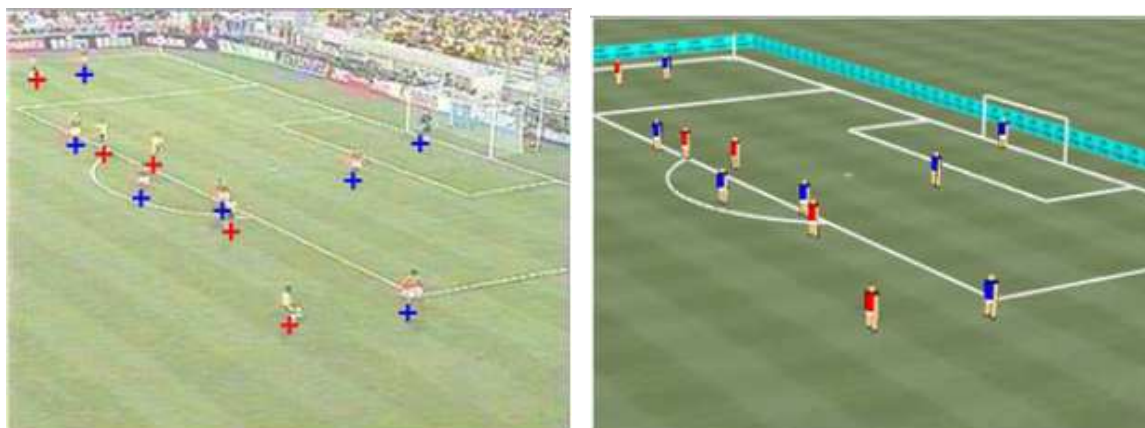


Figura 1.1: Ambiente de Modelagem do Facade



(a) Imagem com a cena original e alguns pontos de referência para efetuar a calibração da câmera

(b) Imagem sintética equivalente a cena da Figura 1.2(a)

Figura 1.2: Processo de reconstrução feito pelo Juiz Virtual.

baseada em imagem. Em (Gruen *et al.* 2005), é feita a reconstrução do sítio arqueológico de Bamiyan, Afeganistão,(Figura 1.3(a)) com suas três estátuas de Buda que foram destruídas durante a invasão da região pelos soviéticos numa guerra que durou 23 longos anos. O projeto fez a reconstrução não só da área arquitetônica (Figura 1.3(b)), mas também da geografia do local (Figura 1.3(c)), juntando técnicas de multi-resolução geométrica e imagens multi-temporais de

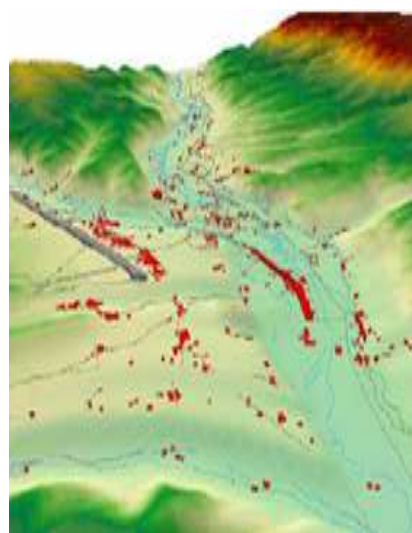
alta resolução.



(a) Penhasco de Bamiyan



(b) Reconstrução 3D do sítio
arqueológico



(c) Reconstrução 3D da geografia da
região

Figura 1.3: Projeto Bamiyan.

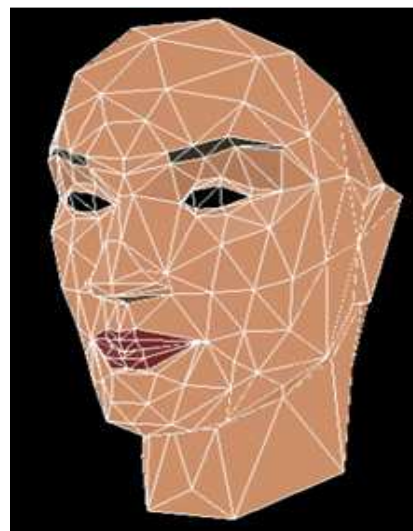
A qualidade da reconstrução 3D baseada em imagens depende diretamente da localização da posição da câmera que gerou a imagem (Williams and Lee 2006). Por isso, em (Vázquez *et al.* 2003) pode ser encontrado um método que seleciona automaticamente a posição da câmera que gerou a imagem da cena. O método proposto utiliza uma unidade de medida conhecida por *viewpoint entropy* que captura a quantidade de informação capturada de um ponto de vista, permitindo selecionar a melhor vista da cena.

Para cada classe de objetos, devem ser aplicadas técnicas específicas que permitam a reconstrução desta classe de objetos, variando de classes simples às mais complexas. Uma classe mais complexa é a das faces humanas. Modelar faces humanas tem uma grande importância para as áreas de entretenimento como jogos, filmes de animação, vídeo-conferência e etc. Tendo como entrada imagens

capturadas por uma câmera em movimento, (Zhang 2001) apresenta uma proposta para modelar este tipo de objeto feita em etapas citadas a seguir: num primeiro momento é feita a captura de duas imagens de ângulos diferentes. Estas são as imagens bases. Então em seguida, o usuário seleciona cinco pontos de referência nestas imagens correspondendo aos cantos dos dois olhos, topo do nariz e os dois cantos da boca (Figura 1.4(a)). O sistema então gera uma malha geométrica (Figura 1.4(b)) que corresponde à face representada na imagem. Por último é aplicada a textura.



(a) imagem base com os pontos de referência



(b) malha geométrica gerada

Figura 1.4: Reconstrução de face humana.

Os trabalhos expostos anteriormente têm diferentes soluções para problemas relacionados às diferentes áreas do conhecimento. No entanto, todos apresentam características semelhantes no que diz respeito à utilização de imagens para obtenção de um modelo tridimensional. Estas imagens constituem a entrada do processo e passam por várias etapas até chegarem à cena tridimensional virtual que representa um modelo do mundo real. Na metodologia aqui proposta, apenas imagens também são necessárias para realizar a reconstrução tridimensional, deixando o usuário livre de grandes intervenções no processo.

1.3 Organização do Trabalho

A estrutura do trabalho está organizada de seguinte forma:

No Capítulo 2, é dado o enfoque na área de reconstrução tridimensional cobrindo os conceitos fundamentais utilizados em cada etapa da metodologia criada para efetuar a reconstrução do nosso modelo, partindo do surgimento e dos conceitos sobre Modelagem Baseada em Imagem, passando pelas técnicas. Serão expostos alguns dos trabalhos relacionados da área de Modelagem Baseada em Imagem que contribuíram para o embasamento teórico deste trabalho e motivação para o desenvolvimento da técnica que foi implementada.

No Capítulo 3, é feita a exposição dos conceitos fundamentais da área de Processamento de Imagens utilizados neste trabalho.

No Capítulo 4, é apresentado o funcionamento do processo com a apresentação dos resultados obtidos e finalmente, no Capítulo 5, é feita a conclusão do trabalho com a apresentação de sugestões para trabalhos futuros.

MODELAGEM BASEADA EM IMAGEM

2.1 Introdução

Tentar recriar no computador um ambiente que simulasse cenas do cotidiano do mundo real de forma automática tem motivado inúmeras pesquisas na área de Computação Gráfica. Sistemas Virtuais Tridimensionais têm sido úteis para a solução de inúmeros problemas do dia-a-dia. É o caso de locais onde a presença do homem seria de difícil acesso, cheia de riscos e com custos significativamente elevados.

No entanto, a geração de modelos complexos para a composição das cenas virtuais, tradicionalmente, é realizada com a utilização de programas específicos para criação de objetos geométricos. Este processo é conhecido como modelagem baseada em geometria. Nesta abordagem, cada objeto presente na cena tem que ser modelado um a um, atribuindo-lhes suas características óticas tais como, material, cor e transparência (Carvalho 2002).

Segundo (Velho *et al.* 2005), o processo de modelagem de ambientes virtuais pode ser dividido em duas etapas:

1. **Criação de modelos geométricos:** um objeto pode ser recriado no computador a partir de métodos de aquisição, ou a partir de especificações geométricas obtidas com a ajuda de técnicas modelagem;

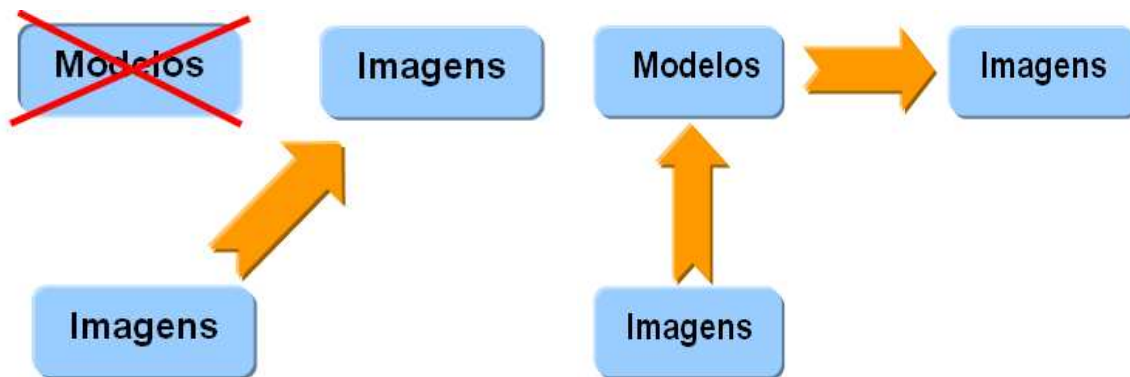
2. **Processamento de modelos geométricos:** envolve o processo de visualização e análise dos objetos que estão sendo descritos.

Segundo (Carvalho 2002), a Computação Gráfica têm como objetivo lidar com o problema da geração de imagens, através de programas de computador, a partir dos dados de uma cena, como poder ser visto na Figura 2.1.



Figura 2.1: Computação gráfica tradicional. Fonte: (Carvalho 2002)

Porém, com o surgimento de novas técnicas da área de Visão Computacional, o esquema da Figura 2.1 pode ser refeito de duas maneiras (Figura 2.2). Na primeira (Figura 2.2(a)), a geração de novas imagens é realizada com a utilização de um conjunto de imagens, não sendo gerado nenhum modelo tridimensional geométrico dos objetos da cena representada na imagem original. Já na Figura 2.2(b), é ilustrado o processo de geração de modelos a partir de imagens. Com esses modelos, pode-se fazer novas imagens sintéticas. O processo representado na Figura 2.2(a) é chamado de visualização baseada em imagens; e na Figura 2.2(b), de modelagem baseada em imagens.



(a) Visualização baseada em imagens

(b) Modelagem baseada em imagens

Figura 2.2: Evolução do processo de geração de imagens. Fonte: (Carvalho 2002)

A visualização baseada em imagens pode ser utilizada na construção de panoramas virtuais, nos quais uma imagem é pintada em uma superfície

panorâmica com o uso de técnicas de mapeamento de textura (Heckbert 1986). Uma câmera virtual é utilizada para observar a superfície panorâmica interativamente (Figura 2.3), e o usuário tem a liberdade de girar a câmera em torno de seu centro e mudar o fator de zoom. A imagem mapeada na superfície é chamada de imagem panorâmica.

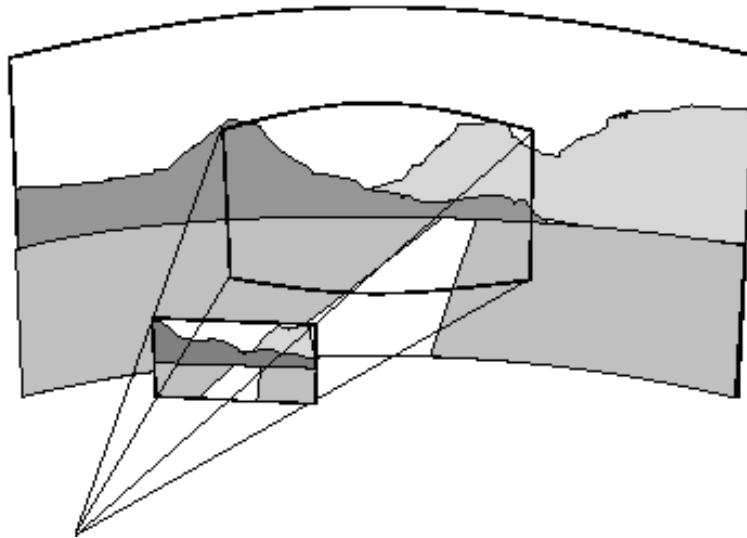


Figura 2.3: Uma câmera virtual observando o panorama. Fonte: (Carvalho 2002)

A modelagem baseada em imagens é uma tecnologia para a re-criação de modelos de objetos em 3D para visualização sob outros pontos de vista. Somente imagens são necessárias para produzir estes modelos em 3D. Nenhum equipamento adicional é necessário. Para atingir este objetivo, é que se convencionou chamar de Fotografia 3D ou Fotografia Tridimensional a área de estudo que engloba diversas técnicas de Visão Computacional, Processamento de Imagens, Modelagem Geométrica e Computação Gráfica.

A utilização da Fotografia 3D possibilita o desenvolvimento de aplicações que sejam capazes de fazer a reconstrução digital de uma cena do mundo real apenas utilizando imagens destas cenas a partir de uma ou mais câmeras.

2.2 Fotografia 3D

Conforme a necessidade de desenvolvimento de novos programas que fossem capazes de atender as necessidades de diversas áreas, surgiu a Fotografia 3D que

reune técnicas das áreas de Visão Computacional e Computação Gráfica.

Fotografia 3D é a utilização das tecnologias para capturar a forma e a aparência de objetos do mundo real e tentar reconstruí-lo no mundo digital. Isto possibilita que estes objetos sejam vistos de novos ângulos de visualização, que, em alguns casos, seria impossível visualizar.

Este é um dos motivos pelo qual sistemas de Fotografia 3D vêm sendo utilizados para facilitar a vida de quem trabalha nas mais diversas áreas de conhecimento como Arquitetura, Patrimônio Histórico, Geografia e Engenharia. Um outro motivo para este incremento no interesse pela Fotografia 3D são os avanços da indústria que fez cair os custos dos equipamentos utilizados para este fim.

As técnicas de Fotografia 3D podem ser empregadas de duas formas: ativa e passiva (Kho 2006). Na ativa, são usados *scanners* 3D e câmeras precisamente calibradas para fazer a recuperação da cena, sendo necessárias condições extremamente favoráveis. O Resultado do emprego deste tipo de técnica são cenas com grandes níveis de detalhes; no entanto, não é recomendado o seu uso para captura de objetos distantes ou em movimento.

Na Fotografia 3D passiva, são utilizados a luz existente na cena e câmeras simples. A cena capturada por este tipo de técnica não é tão rica em detalhes quanta a adquirida pela Fotografia 3D ativa e nem tão precisa, mas é uma alternativa mais barata e pode ser usada para captura de objetos que estejam distantes e/ou em movimento.

O processo de fotografia 3D está relacionado a construção da representação de um modelo do objeto tridimensional a partir de dados digitalizados que são coletados por partes, tendo como resultado dois conjuntos de dados classificados como mapa de distâncias e mapa de atributos, respectivamente (Velho *et al.* 2005). O mapa de distâncias define os retalhos que compõem a superfície dos objetos e o mapa de atributos define as características visuais destes objetos, tais como cor e reflectância. As partes coletadas são reunidas para formar uma superfície completa. Para isso, é necessário alinhar os retalhos, integrá-los e convertê-los para um modelo apropriado.

Todo este processo é feito em etapas listadas a seguir:

1. **Captura:** os dados fotométricos e geométricos são coletados utilizando-se

sensores;

2. **Alinhamento:** os retalhos são alinhados através de um sistema de coordenadas locais pré-definido, onde a localização de cada retalho é relacionada com os seus retalhos vizinhos.
3. **Integração:** é feita a construção de uma superfície contínua simplificada utilizando os retalhos já alinhados, possibilitando a reconstrução da geometria e da topologia dos objetos.
4. **Conversão:** o modelo simplificado produzido é convertido e adaptado para um modelo sem redundâncias e com as características das formas dos objetos.

A implementação deste processo pode ser feita através de duas abordagens: reconstrução no espaço da imagem e reconstrução no espaço da cena 3D (Velho *et al.* 2005). A primeira abordagem baseia-se no espaço local do dispositivo em que a imagem foi capturada, explorando as estruturas destes dispositivos. A segunda é baseada em um espaço global da cena.

2.2.1 Reconstrução Tridimensional

As técnicas de construção de modelos 3D geralmente utilizam *software* de CAD que possuem formas de objetos previamente definidos para que o usuário monte estruturas mais complexas combinando estas formas, uma-a-uma, o que acaba consumindo tempo e dinheiro (El-Hakim *et al.* 2004). Neste cenário, é que fazer a reconstrução semi-automática ou automática torna-se uma alternativa viável.

Para a realização da reconstrução 3D de objetos do mundo real é necessário primeiramente definir a forma de obtenção dos dados, que, no caso, são os objetos de interesse. A obtenção destes dados pode ser feitas utilizando equipamentos como *scanners* 3D ou câmeras digitais. No caso das imagens obtidas através de câmeras digitais, as informações obtidas estão restritas a duas dimensões, requerendo assim o uso de diversas técnicas para obtenção da profundidade ou coordenada 3D (Bardsley 2004).

O princípio básico que permite a recuperação de informação de profundidade a partir de imagens é o princípio de visão estéreo. A visão estéreo é baseada no

fato de que, se um mesmo ponto de uma cena for observado por dois pontos de vista diferentes, é possível reprojeter dois raios a partir da imagem na cena que se interceptam no ponto onde a luz foi refletida pela cena. Esse processo é chamado triangulação (Velho *et al.* 2005).

Os métodos de estéreo passivo não interagem com os objetos da cena, o sensor utilizado afere medições do objeto em suas condições naturais. Os métodos ativos enviam algum tipo de sinal a ser refletido pelo objeto e então medido pelo sensor, o tipo de sinal emitido está completamente relacionado ao tipo de sensor a ser utilizado para medição (Velho *et al.* 2005).

2.3 Modelo de Câmera

O processo de geração de imagens por uma câmera obedece a um modelo, que, em sua versão mais simples, é denominado de *pinhole*. Neste modelo, representado na Figura 2.4, uma caixa totalmente fechada possui um orifício por onde a luz passa. Os raios que passam por este orifício, quando interceptam o plano de fundo desta caixa, formam uma imagem. Esta imagem formada é gerada invertida e, por isso, considera-se a imagem formada no plano que fica situado à mesma distância do orifício, mas posicionado à frente dele. Nesta situação, o orifício pode ser considerado como centro ótico O e equivale ao olho de um observador que vê o mundo por uma janela.

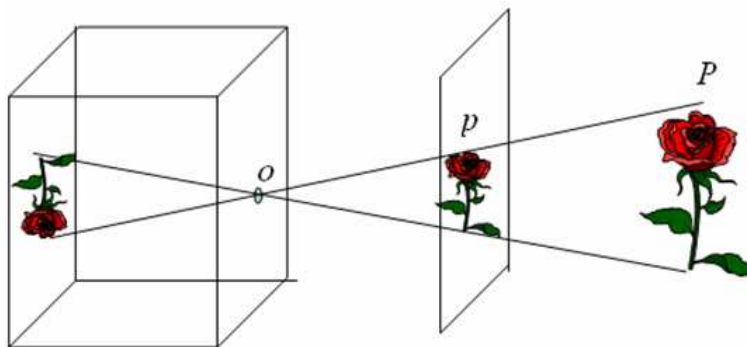


Figura 2.4: Câmera *pinhole* e projeção perspectiva. Fonte: (Velho *et al.* 2005)

Cada ponto P do mundo tridimensional é associado a um ponto p presente no plano onde a imagem é formada. Esta associação é conhecida como projeção

perspectiva (Velho *et al.* 2005).

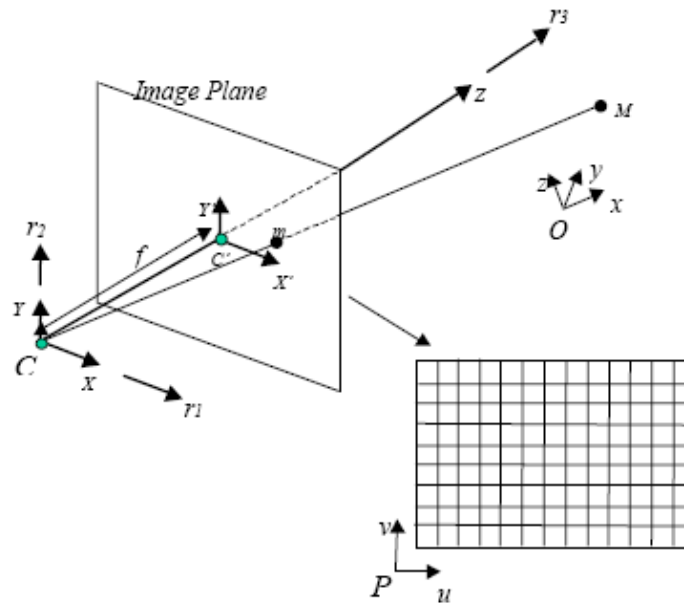


Figura 2.5: Determinação da projeção perspectiva. Fonte: (Carvalho *et al.* 1998)

Considerando um sistema de coordenadas tridimensionais $CXYZ$, com sua origem no centro ótico da câmera C e o eixo Z perpendicular ao plano da imagem com os eixos X e Y paralelos ao mesmo plano, conforme pode ser visto na Figura 2.5. A projeção no plano da imagem forma um outro sistema de coordenadas bidimensional $C'X'Y'$, que define a projeção perspectiva de um ponto XYZ por

$$X' = fX/Z \quad (2.1)$$

$$Y' = fY/Z, \quad (2.2)$$

onde f é chamado de distância focal e representa a distância entre o centro ótico da câmera e o plano da imagem. Em coordenadas homogêneas, esta relação pode ser expressa como:

$$\begin{bmatrix} X' \\ Y' \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (2.3)$$

O sistema de coordenadas da imagem gerada por uma cena do mundo real obedece a um sistema de coordenadas que difere do sistema de coordenadas do mundo real. Para realizar o mapeamento entre estes dois sistemas, é necessário que exista uma transformação entre as coordenadas destes sistemas. Para que isto ocorra, são comumente considerados quatro sistemas de coordenadas (Velho *et al.* 2005), listados a seguir:

- **Sistema de Coordenadas do Mundo (SCM):** é o sistema que descreve a cena no mundo real;
- **Sistema de Coordenadas da Câmera (SCC):** sistema da tela virtual no plano de projeção da câmera virtual, onde ficam as imagens capturadas pela câmera;
- **Sistema de Coordenadas da Imagem (SCI):** sistema de referência bidimensional, associado à superfície de exibição do dispositivo de saída gráfica;
- **Sistema de Coordenadas em *Pixel* (SCP):** sistema também bidimensional, porém com as coordenadas sendo expressas em *pixels*, onde as imagens aparecem sem as distorções provocadas pelos equipamentos.

O processo de aquisição da imagem de um ponto no espaço, através de uma câmera digital, é realizado através de uma seqüência de transformações que relacionam os quatro sistemas.

A primeira transformação realizada é feita entre o SCM e o SCC. Esta transformação baseia-se na idéia de que é desconhecida a orientação da câmera nos três eixos do SCM em relação a um determinado referencial. Esta situação está representada na Figura 2.6.

Além disso, sua posição em relação a um determinado referencial também é desconhecida, como mostra a Figura 2.7.

Então, a transformação de um ponto \tilde{P} do SCM para um ponto P do SCC é definida por:

$$\tilde{P} = RP + T, \quad (2.4)$$

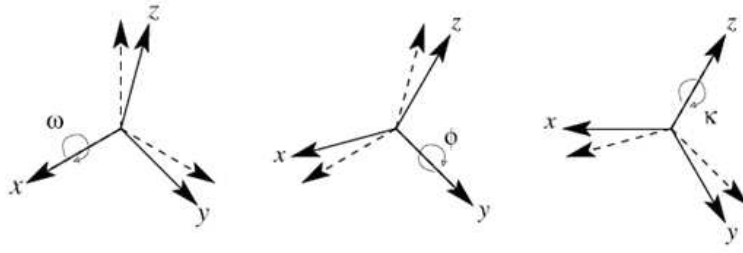


Figura 2.6: Rotação entre dois referenciais.

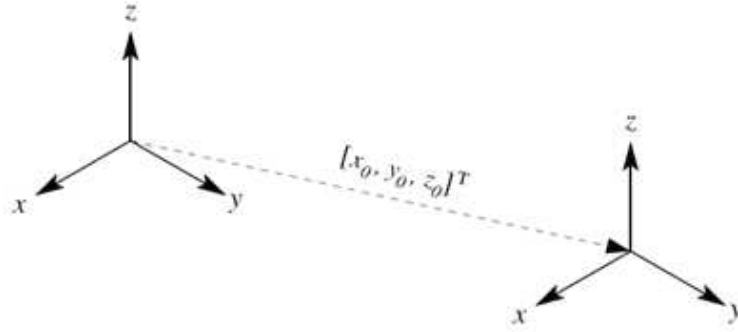


Figura 2.7: Translação entre dois referenciais.

onde R determina a orientação e T a posição da câmera. Em coordenadas homogêneas, temos:

$$\begin{bmatrix} \tilde{X} \\ \tilde{Y} \\ \tilde{Z} \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \quad (2.5)$$

onde \tilde{X} , \tilde{Y} e \tilde{Z} representam os eixos do SCC e R é definido por uma matriz de 9 elementos, resultante da multiplicação das três matrizes de rotação referente a cada eixo.

Assim:

$$R = R_1(\kappa)R_2(\phi)R_3(\omega) \quad (2.6)$$

onde,

$$R = \begin{bmatrix} \cos \kappa & \sin \kappa & 0 \\ -\sin \kappa & \cos \kappa & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \phi & 0 & -\sin \phi \\ 0 & 1 & 0 \\ -\sin \phi & 0 & \cos \phi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \omega & \sin \omega \\ 0 & -\sin \omega & \cos \omega \end{bmatrix} \quad (2.7)$$

Como resultado desta multiplicação, tem-se:

$$R = \begin{bmatrix} \cos \phi \cos \kappa & \cos \omega \sin \kappa + \sin \omega \sin \phi \sin \kappa & \sin \omega \sin \kappa - \cos \omega \sin \phi \cos \kappa \\ -\cos \omega \sin \kappa & \cos \omega \cos \kappa - \sin \omega \sin \phi \sin \kappa & \sin \omega \cos \kappa + \cos \omega \sin \phi \sin \kappa \\ \sin \phi & -\sin \omega \cos \phi & \cos \omega \cos \phi \end{bmatrix} \quad (2.8)$$

ou simplesmente,

$$R = \begin{bmatrix} r_{1x} & r_{1y} & r_{1z} \\ r_{2x} & r_{2y} & r_{2z} \\ r_{3x} & r_{3y} & r_{3z} \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}, \quad (2.9)$$

onde r_1, r_2, r_3 são vetores linhas de R . T representa um vetor translação,

$$T = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (2.10)$$

Logo,

$$[R|T] = \begin{bmatrix} r_1 & t_x \\ r_2 & t_y \\ r_3 & t_z \end{bmatrix} = \begin{bmatrix} r_{1x} & r_{1y} & r_{1z} & t_x \\ r_{2x} & r_{2y} & r_{2z} & t_y \\ r_{3x} & r_{3y} & r_{3z} & t_z \end{bmatrix} \quad (2.11)$$

Do SCC ao SCI é realizada uma transformação perspectiva, como visto na Equação 2.3.

Do SCI ao SCP,

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & \tau & u_c \\ 0 & s_y & v_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad (2.12)$$

onde s_x e s_y representam o número de *pixels* por unidade de comprimento, nas direções vertical e horizontal, sendo na maior parte das câmeras, considerados iguais; u_c e v_c representam a posição, também em *pixels*, da projeção ortogonal C da origem sobre o plano de projeção, com C , em muitas câmeras, representando o centro da imagem e u_c e v_c iguais à metade das dimensões da imagem; τ é a tangente do ângulo formado entre as colunas e linhas de *pixels*. Na maioria dos casos, $\tau=0$, quando as colunas forem perpendiculares às linhas.

Então, para levar um ponto P do SCM a um ponto p do SCP, é feita a composição da transformação realizando a multiplicação das equações 2.5, 2.3 e 2.12. Daí vem:

$$[p] \cong \begin{bmatrix} s_x & \tau & u_c \\ 0 & s_y & v_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} [P] \quad (2.13)$$

Ou ainda,

$$[p] \cong \begin{bmatrix} fs_x & f\tau & u_c \\ 0 & fs_y & v_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & T \end{bmatrix} [P] \quad (2.14)$$

A matriz

$$K = \begin{bmatrix} fs_x & f\tau & u_c \\ 0 & fs_y & v_c \\ 0 & 0 & 1 \end{bmatrix}$$

é chamada de matriz de calibração e reúne os parâmetros intrínsecos da câmera, enquanto

$$\begin{bmatrix} R & T \end{bmatrix}$$

representa os extrínsecos (Velho *et al.* 2005).

2.4 Calibração de câmera

A qualidade da reconstrução tridimensional baseada em imagens depende diretamente da localização da posição da câmera que gerou a imagem (Williams and Lee 2006). Por isso, fazer a calibração da câmera constitui um pré-requisito para alguns estágios deste processo.

Nos casos em que se deseja tratar elementos não situados em um único plano, é necessário determinar a câmera que gerou a imagem.

Como foi visto na seção anterior, uma câmera é composta, ou pode ser modelada, por parâmetros intrínsecos, compostos pela geometria interna e características ópticas e digitais, e extrínsecos, dados pela sua localização, direção de visualização e inclinação. Assim, através destes parâmetros é possível recuperar diversas informações que permitem realizar uma modelagem da cena (Carvalho *et al.* 1998) e até mesmo inserir novos objetos, respeitando, entre diversas características, a oclusão.

Calibrar uma câmera consiste em, dada uma amostra de n pontos $P_i = (x_i, y_i, z_i)$ no espaço 3D e suas projeções correspondentes $p_i = (u_i, v_i)$ na imagem, encontrar uma transformação T que mapeie cada ponto P_i no ponto correspondente p_i (Figura 2.8). Em outras palavras, calibração de câmera, no contexto de visão computacional, consiste no processo de determinar os parâmetros intrínsecos e extrínsecos da câmera (Faugeras 1993).

Técnicas para a calibração de câmera são divididas em três categorias (Bardsley 2004): linear, não-linear e múltiplos-passos (*two-step*). Técnicas lineares usam o modelo de câmera *pinhole* e não levam em conta os efeitos da distorção provocados pelas lentes de muitos dos dispositivos de aquisição; Técnicas não lineares estabelecem uma relação entre os parâmetros de câmera para encontrar uma solução para este problema usando um processo iterativo de minimização; Na última categoria, tem-se a combinação das técnicas lineares e não-lineares para encontrar uma solução direta para alguns parâmetros e estimar outros.

Escrevendo as relações descritas pela Equação 2.14, com o uso de equações lineares na variável de posição de um objeto, os coeficientes destas equações serão exatamente funções dos dados que a calibração determina. O processo de calibração de câmera pode se resumir a encontrar tais valores a partir de um

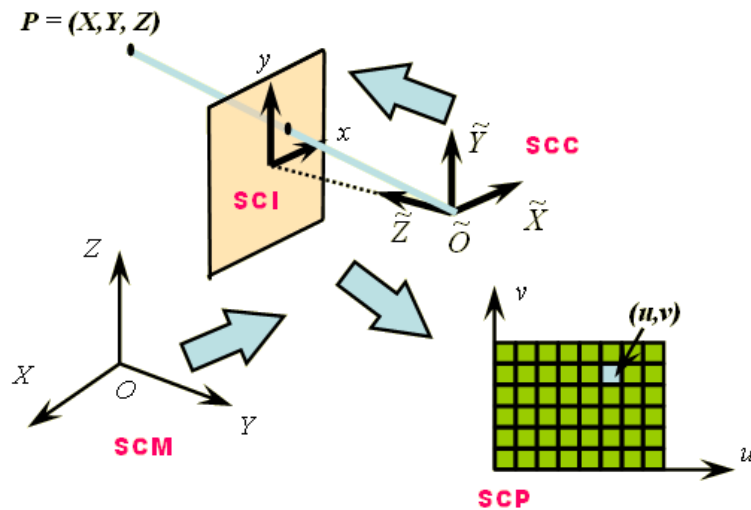


Figura 2.8: Transformação de um ponto P no mundo real para um ponto p no plano da imagem. Fonte: (Carvalho *et al.* 1998)

conjunto de pontos 3D e 2D correspondentes.

A resolução desse sistema de equações, conhecendo os conjuntos de pontos p_i e P_i é o problema central da calibração, e por isso existem diversos métodos para encontrar tais soluções, tais como os propostos em Tsai (Tsai 1987) e Zhang (Zhang 2000). De um modo geral, o processo de calibração pode ser descrito da seguinte forma:

- Determinar um conjunto de pontos no sistema de coordenadas do mundo e suas respectivas projeções nas imagens;
- Resolver as equações lineares;
- Otimizar os dados obtidos, utilizando o resultado da etapa anterior como solução inicial.

2.4.1 Método de Tsai

Roger Tsai, propôs dois métodos para calibrar câmera, um coplanar (são considerados todos os pontos que estão situados no mesmo plano) e outro não-coplanar. O modelo de câmera de Tsai é baseado na projeção perspectiva *pinhole* e leva em consideração as distorções causadas pelas lentes. O método proposto

recupera a orientação interior e exterior, coeficiente de distorção e fator de escala que melhor enquadra as medidas das coordenadas da imagem (Tsai 1987).

Na implementação deste método, são necessários um conjunto de coordenadas 3D referentes ao SCM e seus correspondentes no SCP, formando um vetor cena-imagem, com 5 valores. O cálculo é feito em dois passos (*two-step*): primeiro são obtidas a posição e orientação da câmera e depois os seus parâmetros internos.

Cada par de pontos cena-imagem $(X_i, Y_i, Z_i; u_i, v_i)$ fornece uma equação encontrada como mostrado a seguir. Com $U_1 = r_{xx}/T_y$, $U_2 = r_{xy}/T_y$, $U_3 = T_x/T_y$, $U_4 = r_{yx}/T_y$, $U_5 = r_{yy}/T_y$, temos:

$$u_i = f \frac{r_{xx}X_i + r_{xy}Y_i + T_x}{r_{zx}X_i + r_{zy}Y_i + T_z}, v_i = f \frac{r_{xx}X_i + r_{xy}Y_i + T_y}{r_{zx}X_i + r_{zy}Y_i + T_z} \quad (2.15)$$

$$\frac{u_i}{v_i} = \frac{r_{xx}X_i + r_{xy}Y_i + T_x}{r_{yx}X_i + r_{yy}Y_i + T_y} = \frac{\frac{r_{xx}}{T_y}X_i + \frac{r_{xy}}{T_y}Y_i + \frac{T_x}{T_y}}{\frac{r_{yx}}{T_y}X_i + \frac{r_{yy}}{T_y}Y_i + 1} \quad (2.16)$$

Daí,

$$v_i X_i U_1 + v_i Y_i U_2 + v_i U_3 - u_i X_i U_4 - u_i Y_i U_5 = u_i \quad (2.17)$$

Os valores de U_1, \dots, U_5 são encontrados resolvendo um sistema de equações lineares com mais equações do que incógnitas (mínimos quadrados). As condições de ortonormalidade permitem, a partir daí, obter R , T_x e T_y . f e T_z podem ser obtidos por:

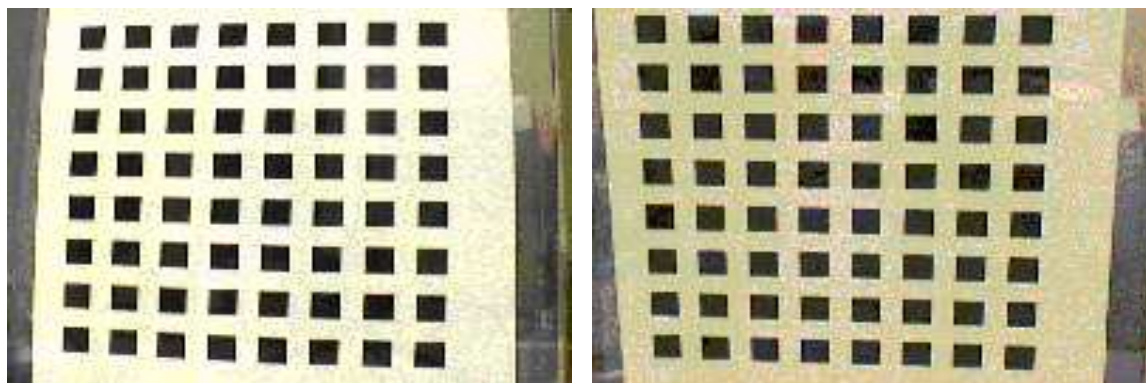
$$\begin{cases} (r_{xx}X_i + r_{xy}Y_i + T_x)f - u_i T_z = u_i(r_{zx}X_i + r_{zy}Y_i) \\ (r_{yx}X_i + r_{yy}Y_i + T_y)f - v_i T_z = v_i(r_{zx}X_i + r_{zy}Y_i) \end{cases} \quad (2.18)$$

2.4.2 Método de Zhang

Métodos de calibração coplanares, como o de Tsai, são fáceis de implementar, porém não são tão precisos quando aplicados a processos de reconstrução tridimensional (Velho *et al.* 2005). (Zhang 2000) propôs um método de calibração que necessita apenas de uma câmera para encontrar um padrão

coplanar bidimensional, dando origem a algumas imagens (pelos menos duas) com orientações diferentes (Figura 2.9). Nestas imagens, são conhecidas as coordenadas do sistema de referência padrão, os parâmetros intrínsecos são os mesmos para todas elas e os parâmetros extrínsecos mudam quando o padrão for reposicionado.

Este método de calibração está inserido nas técnicas de múltiplos-passos e utiliza: no primeiro passo, aproximação linear para determinação de uma solução analítica; no segundo passo, estimação da distorção radial da lente (dois termos); e, no terceiro passo, otimização não-linear de todos os parâmetros



(a)

(b)

Figura 2.9: Imagens obtidas com diferentes orientações onde os pontos dos vértices dos quadrados são utilizados como parâmetros de entrada ao processo. Fonte: (Zhang 2000)

Os passos para implementação do processo são:

1. **Determinação dos parâmetros intrínsecos:** centro da imagem, fatores de escala;
2. **Determinação dos parâmetros extrínsecos:** matriz de rotação 3D e vetor de translação 3D;
3. **Estimação dos coeficientes de distorção:** distorção radial da lente;
4. **Refinamento não-linear de todos os parâmetros do modelo:** incluem-se os parâmetros de distorção radial, através de minimização.

PROCESSAMENTO DE IMAGENS

3.1 Introdução

A grande maioria das tecnologias de fotografia 3D, para realizar a calibração e reconstrução de objetos 3D do mundo real, necessita manipular imagens que contém os objetos desejados, logo, técnicas para manipulação de imagens são necessárias.

Uma área dentro da Ciência da Computação que estuda tais técnicas é chamada de Processamento de Imagens. O objetivo do processamento de imagens é manipular imagens de modo que a entrada e o resultado do processo sejam imagens. Mas, segundo (Jahne 2002), esse processo não é feito apenas em um passo e sim em várias etapas. (Gonzalez and Woods 1992) define que as etapas para o processo de manipulação de uma imagem são: aquisição da imagem, pré-processamento, segmentação, representação/descrição e reconhecimento/interpretação.

A aquisição da imagem é feita por meio de algum dispositivo de entrada como câmeras digitais. Na etapa de pré-processamento, procura-se minimizar a presença de ruídos na imagem decorrentes da aquisição ou ainda realçar características importantes, tais como bordas e vértices, tornando a etapa de segmentação mais eficiente. A etapa de segmentação é responsável em agrupar os *pixels* pertencentes a um mesmo objeto ou região. As informações de interesse podem também ser

extraídas das imagens e representadas de uma outra forma. A representação consiste das várias formas de armazenar a fronteira e o interior de objetos segmentados, podendo conter informações sobre a forma e a topologia dos objetos. A descrição quantifica estas informações através da extração de características estruturais. E, finalmente, com base na descrição, o reconhecimento associa um rótulo a cada objeto segmentado enquanto a interpretação associa um significado ao conjunto de objetos segmentados.

O problema a ser estudado neste trabalho, é que dada uma imagem que apresenta a visualização, total ou parcial, de um determinado modelo, calibrar a câmera que gerou a imagem de forma semi-automática, a fim de sobrepor objetos virtuais. Para fazer isto, pressupõe-se que é feita alguma detecção de um determinado padrão conhecido nas imagens reais. A detecção pode ser realizada através de intervenção humana (semi-automática), informando pontos que caracterizam localização do padrão na imagem e seus correspondentes no padrão real, ou sem esta intervenção, com o sistema localizando e reconhecendo automaticamente o padrão desejado na imagem. O processo de aquisição foi feito através de câmera digital. Na etapa de pré-processamento, aplicou-se o filtro de Canny para detectar as bordas e o método de Harris para detecção de vértices. Foi aplicado a técnica de *template matching* para fazer a segmentação e classificação dos objetos presentes na imagem. Para isso, para cada objeto constituinte na imagem, foi construído um *template* baseado na câmera calibrada, utilizando o algoritmo de fecho convexo para selecionar apenas a região exterior do *template*.

3.2 Detecção de Bordas

As bordas de uma imagem compreendem os limites entre duas regiões com uma diferença brusca de intensidade. Muitas aplicações de processamento de imagens utilizam esta técnica como um pré-processamento para auxiliar na extração de informações importantes para o reconhecimento e classificação de objetos (Neoh and Hazanchuk 2004).

Canny definiu em seu método um conjunto de metas que um detector de bordas deveria ter (do Vale *et al.* 2004). São eles:

- **Taxa de erro ou detecção:** o detector de bordas deveria detectar e achar

somente bordas, nenhuma borda deveria faltar;

- **Localização:** a distância entre os pixels de borda encontradas pelo detector de bordas e a borda real deveriam ser o menor possível;
- **Resposta:** o detector de bordas deve identificar somente um ponto de borda onde existe uma única borda verdadeira.

Para uniformizar o ruído e localizar as bordas, o detector de Canny necessita de um filtro para realizar uma convolução que otimize os três critérios citados anteriormente. Se considerarmos uma borda de uma dimensão variando no contraste e, então, convolucionando a borda com a função de uniformização de Gauss, o resultado será uma variação contínua do valor inicial ao final, com uma inclinação máxima no ponto onde existe um "degrau". Se esta continuidade é diferenciada em relação a x , esta inclinação máxima será o máximo da nova função em relação a original.

As bordas na imagem são indicadas pelos máximos da convolução da máscara. Para realizar este processo, é usada uma função de Gauss de 2-Dimensões na direção de x e y . Os valores das máscaras de Gauss dependem da escolha do valor de sigma nas equações:

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad (3.1)$$

$$G'(x) = \frac{-x}{\sqrt{2\pi}\sigma^3} e^{-\frac{x^2}{2\sigma^2}} \quad (3.2)$$

A aproximação do filtro de Canny para detecção de bordas é G' . A convolução de G' na imagem resulta em uma imagem I que mostrará as bordas, mesmo na presença de ruído. Para melhor desempenho na implementação computacional, uma convolução de Gauss de 2-dimensões pode ser separada em duas de 1-dimensão. (do Vale *et al.* 2004) descreveu, computacionalmente, os passos seguidos. São eles:

1. Ler a imagem I a ser processada;

2. Criar uma máscara Gaussiana bidimensional G para convoluir com I , dando origem a I' s. O desvio padrão desta Gaussiana é um parâmetro do detector de bordas;
3. Criar duas máscaras unidimensionais para a diferenciação da imagem suavizada, nas direções x (linha) e y (coluna), denominando-as de G_x e G_y ;
4. Convoluir a imagem I com G_x ao longo das linhas, gerando a imagem I_x e, analogamente, ao longo das colunas para gerar I_y ;
5. Calcular a magnitude em cada *pixel* (x, y) na forma que segue:

$$M(x, y) = \sqrt{I_x(x, y)^2 + I_y(x, y)^2} \quad (3.3)$$

6. Executar a supressão não máxima, que é o anulamento dos *pixels* cujos valores não sejam máximos locais na direção perpendicular à borda, sendo que este processo produz um afinamento das bordas, atendendo assim o terceiro critério de desempenho de Canny;
7. Computar e aplicar uma limiarização adaptativa (histerese), que consiste em uma limiarização baseada em dois limiares τ_1 e τ_2 , onde $\tau_1 \cong 2\tau_2$ ou $\tau_1 \cong 3\tau_2$. Aplicando a limiarização duas vezes, uma para τ_1 e outra para τ_2 , o algoritmo efetua um processo de complementação das discontinuidades da primeira limiarização aproveitando o resultado da segunda.

3.3 Detecção de vértices

Dependendo das características dos objetos, aplicam-se diferentes técnicas para a obtenção de um modelo tridimensional. No caso de sólidos geométricos, uma informação que ajuda no processo de detecção em uma imagem é a localização dos cantos ou vértices dos objetos (Deriche and Giraudon 1993).

O método de Harris foi proposto em 1988, por Harris e Stephens (Harris and Stephens 1988), e considera o mínimo e o máximo valor próprio, α e β respectivamente, da matriz de covariância do gradiente de uma imagem binária:

$$M = \begin{bmatrix} (\partial I/\partial x)^2 & (\partial I/\partial x)(\partial I/\partial y) \\ (\partial I/\partial x)(\partial I/\partial y) & (\partial I/\partial y)^2 \end{bmatrix}, \quad (3.4)$$

onde $(\partial I/\partial x)$ e $(\partial I/\partial y)$ são os gradientes da imagem I nas direções x e y , respectivamente.

Um vértice é detectado quando o gradiente tem uma estrutura bidimensional, ou seja, quando os dois valores próprios são elevados e similares em magnitude (Azevedo *et al.* 2005). Para evitar uma decomposição explícita dos valores próprios de M , Harris definiu a seguinte função de medida:

$$R = \det(M) - k(\text{trace}(M))^2, \quad (3.5)$$

onde $\det(M) = \alpha\beta$ e $\text{trace}(M) = \alpha + \beta$.

Harris sugere $k = 0.04$, mas como usualmente este parâmetro necessita ser estimado, o cálculo pode ser feito de outra forma, utilizando-se a Equação 3.6:

$$R = \frac{\det(M)}{\text{trace}(M) + \epsilon}, \quad (3.6)$$

onde a constante

$$\epsilon = 2 \times 10^{-52}$$

é utilizada para evitar uma divisão por zero, quando M for nulo. R terá um valor elevado se os valores próprios forem similares e elevados.

Para minimizar os resultados obtidos, pode-se assumir que R seja superior a um dado limiar. Um filtro M de suavização gaussiano com um desvio padrão σ aplicado aos termos, diminui a sensibilidade a ruídos, porém reduz a precisão da localização dos pontos fortes. Para solucionar este problema, aplica-se o operador morfológico de dilatação, com uma janela de tamanho r . Escolhem-se os vértices mais fortes, rejeitando-se os que estiverem muito próximos dentro de uma certa área de raio d .

3.4 Fecho Convexo

Para realizar o processo de template matching, que identifica os objetos presentes na imagem, é feita a construção de templates dos objetos do modelo utilizando-se o algoritmo de fecho convexo para descrever o contornos destes objetos. Este processo de criação será detalhada no Capítulo seguinte.

Um conjunto S é chamado de fecho convexo quando é o menor polígono convexo que, dado um conjunto disperso de pontos P , consegue abranger no seu interior todos os pontos do conjunto S .

Alguns dos algoritmos para encontrar uma solução para este problema são Varredura de Graham (Graham Scan), Embrulho para presente (Jarvis), Quickhull e Divisão-e-Conquista (Merge-Hull). A seguir, será descrito cada um deles conforme apresentado em (Figueiredo and Carvalho 1991).

- **Varredura de Graham (Graham Scan):** Encontra-se o ponto P com menor coordenada y e então os pontos restantes são ordenados pelo ângulo em relação à linha horizontal que passa pelo ponto P . Com ajuda de uma pilha, os dois primeiros pontos são empilhados para, em seguida, percorrer-se os pontos ordenando-os da seguinte forma:

1. Se o ponto X em questão está à esquerda do vetor penúltimo-último elemento da pilha (último elemento da pilha é o topo) então empilhamos e olhamos o próximo;
2. Se o ponto X em questão está à direita deste mesmo vetor, então desempilhamos e voltamos a analisar X .

Ao final tem-se na pilha uma lista de pontos que formam o fecho convexo dos pontos;

- **Embrulho para presente (Jarvis):** Inicialmente, determina-se o ponto P do conjunto com menor coordenada y . Em seguida encontra-se o ponto Q , dentre os que outros pontos, com menor ângulo em relação a P e a linha horizontal. Esse é um ponto que é vértice do fecho convexo, e mais, a aresta PQ pertence ao fecho convexo. O processo segue, encontrando-se o próximo ponto com menor ângulo em relação a aresta PQ , e assim por diante. A execução termina quando o ponto P inicial é encontrado novamente;

- **Quickhull.** Seu funcionamento é similar ao algoritmo de ordenação Quicksort. Divide-se o plano em 5 grupos, utilizando-se dos pontos com menor e maior coordenada x e maior e menor coordenada y do conjunto. A região interna a este quadrilátero é desconsiderada, pois são todos pontos interiores. A seguir, aplica-se a cada um dos grupos restantes um algoritmo recursivo, que calcula o ponto P mais distante do segmento correspondente (na primeira vez é um segmento do quadrilátero). A recursão é aplicada até uma parte possuir somente dois pontos, que são então conectados;
- **Divisão-e-Conquista (Merge-Hull):** A idéia é dividir o problema ao meio, resolver recursivamente as duas metades para depois reuni-las novamente. A parte da conquista consiste em descobrir as duas arestas tangentes (superior e inferior) às duas metades. Isso pode ser feito da seguinte forma: encontra-se o ponto mais à direita da metade da esquerda e o ponto mais à esquerda da metade da direita, em seguida "suba" a aresta que liga esses dois pontos até que ela seja a tangente superior e a inclua no fecho (retirando as arestas interiores que estão nos subfechos). Analogamente "desça" a aresta inicial até que ela seja a tangente inferior.

Neste trabalho foi utilizado o algoritmo do Embrulho para presente.

3.5 Template Matching

O *template matching* é uma técnica utilizada para localizar um objeto em uma imagem, possivelmente com uma dimensão maior que a do *template*. Isto de forma automática, apenas informando que imagem está se buscando e qual o *template* (Sonka *et al.* 1994). Basicamente, esta técnica consiste em criar um *template* para cada classe do problema em questão e depois comparar o exemplo de teste com todos os *templates* disponíveis. Aquele *template* que tiver a menor distância será a classe escolhida.

Uma maneira de se fazer a combinação entre as imagens (*template matching*) é percorrer *pixel a pixel* o *template* e verificar em que ponto da imagem base, percorrendo-a da mesma maneira, ele está localizado. No entanto, esta técnica é muito lenta, dependendo do tamanho das imagens. Então, motivado por um

tempo de processamento inferior ao da comparação *pixel a pixel*, foi desenvolvido, neste trabalho, um algoritmo baseado em distâncias entre somente os pontos dos vértices dos objetos presente nas imagens, baseada na técnica da "Distância de Hausdorff" (Huttenlocher *et al.* 1993).

A distância de Hausdorff pode ser definida da seguinte forma:

$$h(A, B) = \max\{\min\{d(a, b)\}\}, \quad (3.7)$$

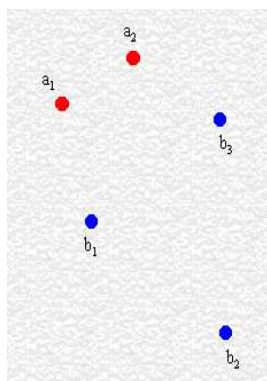
onde a e b são pontos do conjuntos A e B respectivamente e $d(a, b)$ é o cálculo da distância entre estes pontos utilizando uma métrica, como por exemplo a distância Euclidiana.

Esta técnica calcula sucessivas distâncias entre pontos de objetos. Primeiramente, é escolhido um ponto a_i do conjunto A (Figura 3.1(a)). Logo após, é calculado a distância deste ponto a_i e todos os pontos b_i do conjunto B , conforme pode ser visto na Figura 3.1(b). Então é escolhida a menor distância entre todas as que foram calculadas no passo anterior (Figura 3.1(c)). Isto é feito para todos os pontos do primeiro conjunto (Figura 3.1(d)). Por último, é escolhida a maior destas menores distâncias (Figura 3.1(e)).

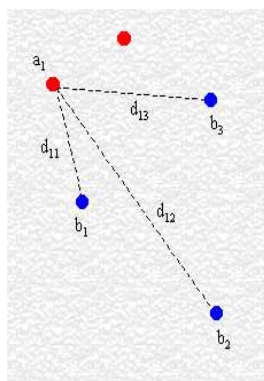
A técnica de "Distância de Hausdorff" pode ser aplicada no cálculo entre polígonos convexos utilizando essa mesma idéia de distâncias, com a adição de um segundo passo que calcula as distâncias também entre os pontos do conjunto B para o conjunto A (muita das vezes a distância $h(A, B)$ não é igual a $h(B, A)$). Por último, é escolhido o maior valor entre esses dois encontrados, conforme descreve a Equação 3.8. Mas para este trabalho foi utilizado apenas o primeiro passo da técnica ($h(A, B)$).

$$H(A, B) = \max(h(A, B), h(B, A)) \quad (3.8)$$

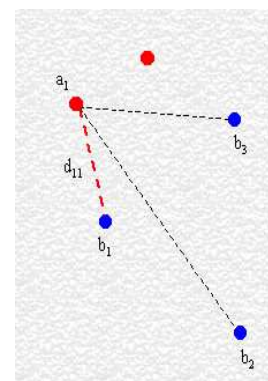
O *template matching* é uma técnica que ao mesmo tempo segmenta e classifica os objetos de uma determinada classe. O sucesso desta técnica pode ser comprometido devido aos ruídos produzidos durante a etapa de aquisição que aumenta a variabilidade dos objetos presente na imagem.



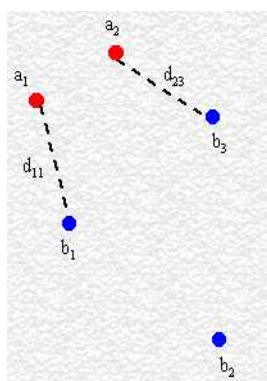
(a) Conjuntos de pontos A e B



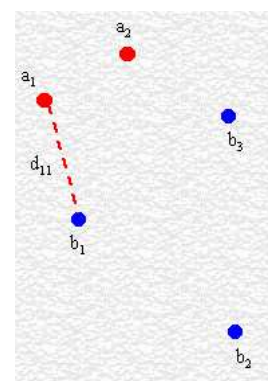
(b) Cálculo da distância entre um ponto do conjunto A e todos os pontos do conjunto B



(c) Obtenção da menor distância



(d) Menores distâncias obtidas



(e) Maior distância entre as menores

Figura 3.1: Cálculo da distância Hausdorff.

METODOLOGIA PARA RECONSTRUÇÃO 3D DE SÓLIDOS GEOMÉTRICOS

Para fazer a reconstrução a partir de uma imagem que representa um modelo 3D bem definido do mundo real, são necessárias várias etapas que incluem alguns dos algoritmos mais utilizados das áreas de Visão Computacional e Processamento de Imagens, dentre eles, calibração de câmera, detecção de bordas, detecção de vértices e *template matching*.

O problema estudado neste trabalho é a obtenção de um modelo 3D a partir de uma imagem que representa a visualização dos objetos de uma classe de objetos que possua características bem definidas, tais como a geometria, de modo que a intervenção do usuário seja a menor possível, no que diz respeito ao fornecimento de algum tipo de dado ou parâmetros externos.

A metodologia proposta, neste trabalho, será aplicada a um conjunto de objetos conhecidos dos quais possuímos os seus modelos geométricos e dimensões e alguns pontos com coordenadas conhecidas. A partir daí, será feito o reconhecimento e o posicionamento destes objetos em um modelo virtual tridimensional guiado pelas informações de calibração da câmera que obteve a imagem. O posicionamento de cada objeto do modelo inclui definir uma translação no eixo X e outra no eixo Y e uma rotação no eixo Z do objeto descrito em um sistema de coordenadas locais.

Nesta metodologia, utilizamos a técnica de calibração de câmera para recuperar informações referentes à câmera que gerou a imagem, possibilitando a construção de *templates*, representando os objetos do modelo 3D. Estes *templates* são construídos baseados nestas informações da câmera para, em seguida, serem utilizados para procurar na imagem, com a técnica de *template matching*, as coordenadas dos vértices dos objetos do modelo.

4.1 Etapas do processo de reconstrução

Conforme a Figura 4.1 ilustra, a metodologia proposta para realizar a reconstrução tridimensional de uma cena que representa um modelo pode ser feita em etapas dentre as quais: aquisição, calibração, detecção de bordas e vértices, construção de *templates*, *matching* e visualização 3D.

O método tem como entrada uma imagem adquirida por uma câmera digital, representando o modelo 3D do mundo real. Apenas uma imagem é necessária. Com esta imagem adquirida, são realizadas duas etapas: A etapa de detecção de bordas e de vértices e a etapa de calibração.

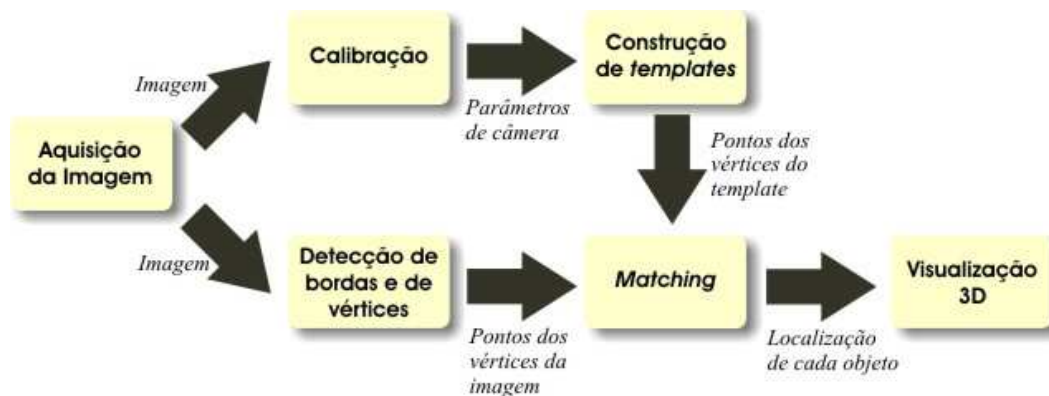


Figura 4.1: Etapas do processo de reconstrução 3D

A etapa de detecção de bordas e de vértices é realizada para obtenção dos possíveis vértices dos sólidos geométricos na imagem adquirida. Para encontrar estes vértices, primeiramente é realizado um pré-processamento para detectar as bordas utilizando-se o filtro de Canny, diminuindo a quantidade de informação na imagem necessária para obtenção dos vértices. Logo após a detecção de bordas, são encontrados os conjuntos de coordenadas $v_i = (x, y)$ dos possíveis vértices,

através do algoritmo de Harris. Estes vértices formam o conjunto de pontos com os quais os *templates* construídos serão comparados, durante a etapa de *matching*.

Já na etapa de calibração, o usuário primeiramente intervém no processo fazendo o levantamento de um conjunto de pontos $p_i = (x, y)$ da imagem e seus correspondentes $P_i = (x, y, z)$ no mundo real, formando um conjunto de pares de coordenadas cena-imagem. Esta etapa da metodologia tem como resultado os parâmetros intrínsecos e extrínsecos da câmera. Os parâmetros da câmera serão utilizados para a construção dos *templates* que, por sua vez, serão utilizados para localizar os vértices de cada objeto do modelo na imagem, durante a etapa de *matching*.

A construção dos *templates* que permitirão a detecção dos possíveis vértices dos objetos na imagem é feita na etapa de construção de *templates*. Para cada componente ou objeto do modelo, é criado um *template* 2D com a transformação das coordenadas 3D dos vértices de cada um destes objetos para 2D, quando estes vértices estiverem posicionados em algumas posições pré-determinadas. Esta transformação utiliza os parâmetros da câmera que foi calibrada.

De posse dos pontos ou coordenadas dos vértices da imagem, gerados pela etapa de detecção de bordas e de vértices, e dos pontos ou coordenadas dos vértices do *template* 2D que representam cada objeto do modelo, é aplicada a técnica de *template matching*, que realiza o cálculo dos pontos que possuem as menores distâncias entre os vértices do *template* e os vértices encontrados na imagem. Estes pontos constituem as possíveis coordenadas dos vértices do objeto que está sendo procurado.

Na última fase do processo, é feita a reconstrução 3D dos objetos a partir dos pontos ou coordenadas encontrados na etapa anterior de *matching* que representam a localização de cada objeto na imagem. Estas coordenadas bidimensionais são transformadas em coordenadas tridimensionais aplicando-se a perspectiva inversa. Com apenas 3 coordenadas é possível fazer a reconstrução tridimensional. Em seguida, com a geração do mundo virtual tridimensional que representa o modelo do mundo real pode ser feita a visualização dos objetos encontrados de outros ângulos, permitindo ainda a interação com este mundo virtual.

4.1.1 Aquisição

Como parte inicial do processo de reconstrução aqui proposto, é feita a aquisição da imagem utilizando-se uma câmera digital. As imagens obtidas são de uma cena do mundo real construída especialmente para a aplicação da metodologia. A cena representa uma categoria de objetos bem definidos do mundo real. Esta categoria de objetos está constituída de cubos e paralelepípedos de diversos tamanhos feitos de cartolina, dispostos em um plano feito com isopor.

As imagens adquiridas devem conter o mínimo de ruído para não interferir no processo de detecção de vértices. Como é impossível obter uma imagem livre de qualquer ruído, as imagens devem ser capturadas num ambiente onde a iluminação não interfira, gerando sombras indesejáveis. Como a identificação dos objetos do modelo na imagem é feita por vértices, estes objetos devem aparecer na imagem com as linhas das suas arestas bem definidas. Para isso, as arestas dos objetos de cartolinas foram realçadas.

4.1.2 Detecção de bordas e dos vértices

A etapa de detecção dos vértices dos objetos na imagem é necessária para localizar os possíveis vértices dos objetos presentes na imagem. Estes vértices serão utilizados como entrada na etapa *template matching*. Como foi dito anteriormente, para encontrar os vértices presente na imagem, primeiro é necessário detectar as bordas usando o filtro de Canny. Só depois é aplicado o método de Harris.

Os vértices encontrados nem sempre correspondem aos vértices reais dos objetos do modelo procurados na imagem. Isso deve ao fato da existência de ruídos na imagem que prejudicam a localização destes pontos. Além disso, tanto os filtros de Canny quanto o método de Harris utilizam limiares fornecidos como parâmetros de entrada que podem variar dependendo da imagem que está sendo analisada.

4.1.3 Calibração

Para que a cena do mundo seja reconstruída, é necessário que se obtenha as informações da câmera que gerou a imagem. Com estas informações, é possível inserir novos objetos na cena tridimensional. A etapa de calibração do método

proposto neste trabalho é realizada com o usuário selecionando na imagem um ponto $p(x, y)$. Depois ele informa um ponto $P(X, Y, Z)$ do mundo real (representado pelo modelo da maquete construída), relacionado com a coordenada p , formando uma coordenada cena-imagem $C(X, Y, Z, x, y)$ que associa os dois sistemas de coordenadas. Isto é feito até que se tenha pelo menos 5 coordenadas cena-imagem. Após a formação do conjunto de coordenadas cena-imagem, é realizada a calibração propriamente dita com a utilização do método de Tsai, tendo como resultado os parâmetros intrínsecos e extrínsecos da câmera. Estes parâmetros serão utilizados para construção de templates para cada um dos objetos da classe na etapa mostrada a seguir.

4.1.4 Construção de *templates*

Para solucionar o problema na obtenção da localização dos objetos presente na imagem, o método proposto neste trabalho realiza a etapa de construção de *templates*, que servirão para efetuar a localização dos objetos dentro da imagem durante a etapa de *matching*.

Um *template* representa o modelo de um determinado objeto. Para efetuar a construção dos *templates 2D* que representarão os objetos em 3D do mundo real, são necessárias as informações da câmera que geraram a imagem, pois um novo posicionamento desta câmera gera imagens diferentes.

Com os parâmetros da câmera que geraram a imagem, é possível fazer uma mudança de referencial de um ponto do mundo real para o mundo da imagem para que os pontos dos objetos em 3D sejam planificados. Para obter-se o *template* do objeto, ainda é necessário encontrar os pontos mais externos que constituem a fronteira do objeto. Isto pode ser realizado com o algoritmo de fecho convexo. O processo de criação de um *template* é ilustrado na Figura 4.2.

Como são conhecidas as dimensões de cada objeto existente, no caso de sólidos geométricos, posicionamos os objetos em alguns locais do plano do modelo em estudo, onde se conhece as coordenadas, e obtemos as coordenadas de cada vértice no mundo real. Na Figura 4.3, por exemplo, um cubo foi transladado para a origem.

Na fase de teste, verificou-se que apenas posicionar os objetos na origem não seria o suficiente para a construção de um *template* que pudesse representar um

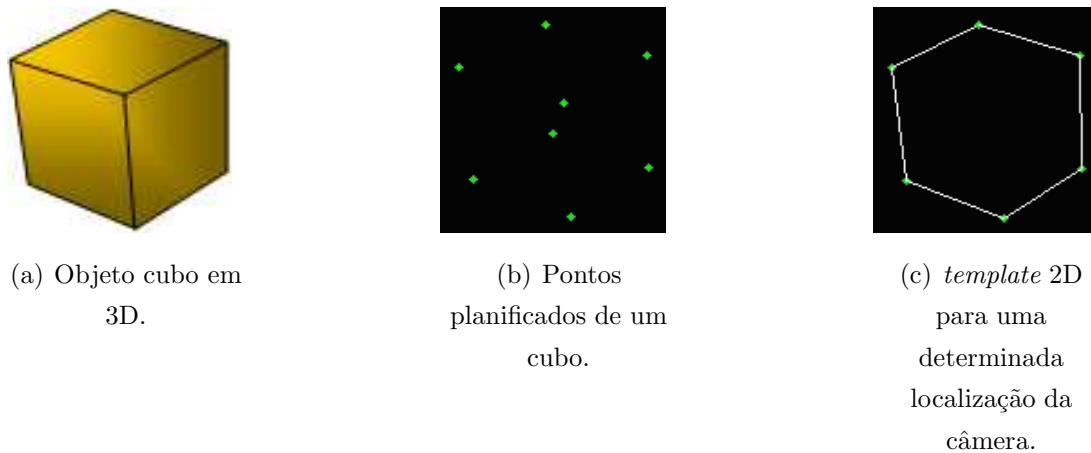
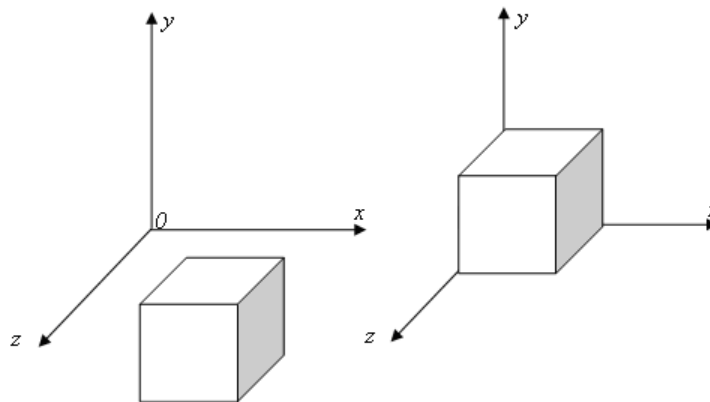
Figura 4.2: Processo de criação do *template* com fecho convexo.

Figura 4.3: Translação de um cubo para a origem.

determinado objeto, pois à medida que a posição da câmera se modifica, tem-se diferentes tamanhos de *templates* para um mesmo objeto. Então, para solucionar este problema, para cada momento em que forem recuperados os parâmetros da câmera, são criados cinco *templates*, com cada um sendo transladado ao longo do plano do modelo. As posições destes objetos foram assim distribuídas a fim se obter os melhores *templates* que representem um determinado tipo de objeto da classe.

A Figura 4.4 ilustra um cubo sendo transladado nas posições $c_1 \dots c_5$, para geração dos cinco *templates*.

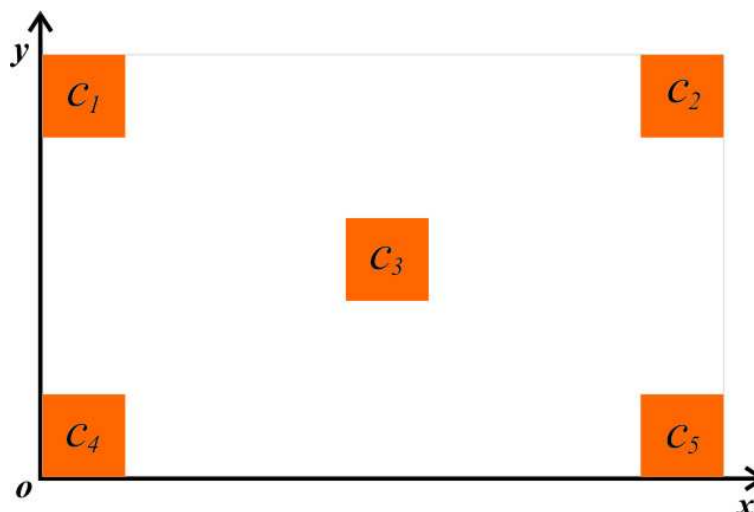


Figura 4.4: Posicionamento de um cubo ao longo do plano do modelo, para geração de 5 *templates* de diferentes tamanhos.

4.1.5 *Matching*

O *matching* é baseado na idéia de calcular sucessivas distâncias. Um algoritmo foi desenvolvido para encontrar o local de *matching* (lugar onde as duas imagens estão mais parecidas se comparadas *pixel a pixel*), fundamentando sua busca por distâncias entre pontos, vértices dos sólidos geométricos, em vez de comparar toda a imagem *pixel a pixel*, baseando-se na técnica da Distância de Hausdorff (Huttenlocher *et al.* 1993).

São extraídos apenas os vértices visíveis dos objetos na imagem base e os vértices do *template*, a partir da construção dos *templates*. Depois, é fixado um ponto na imagem onde será posicionado um ponto do *template*; este ponto dará um coeficiente de translação, que é para onde o *template* foi movimentado sobre os eixos x e y da imagem base. Com esse coeficiente, são calculados os novos pontos do *template*, apenas adicionando este valor ao de suas antigas coordenadas, já que não se pode trabalhar no eixo do *template*, por ser este menor que as dimensões da imagem.

Logo após essa mudança nas coordenadas dos pontos no *template* (a partir do coeficiente de translação), é calculada, para cada novo ponto do *template* (estes apenas transladados sobre a imagem) a menor distância dele para algum ponto dos vértices da imagem, com um detalhe: que não se exceda pontos muito distantes

deste para não haver uma queda no desempenho do algoritmo.

Isto é feito a partir de um limiar calculado sobre a maior distância entre dois pontos do *template*. Feito isto para todos os pontos do *template*, são guardadas as menores distâncias de cada ponto em uma lista. O mesmo processo é feito para um novo ponto do *template* sendo fixado no mesmo ponto fixado anteriormente em um vértice da imagem.

Terminada a etapa de fixação dos pontos do *template*, o algoritmo volta para o passo de fixar um ponto na imagem e o faz para todos seus pontos, a fim de achar um local na imagem com as menores distâncias entre os pontos do *template* e o dos vértices da imagem base adquirida.

Sempre que é realizada uma etapa para busca das menores distâncias, é escolhida a distância que torna a média entre as distâncias dos pontos a menor. E por fim, encontra-se a menor distância que será possivelmente um local de *matching*.

Como o algoritmo de *template matching* utilizado é baseado no cálculo de distâncias sucessivas, então sempre serão encontrados coordenadas que podem representar ou não um objeto. Por isso, mesmo que não se tenha um determinado objeto presente na imagem, mesmo assim será retornado algum valor pelo algoritmo. Outro caso que deve ser levado em consideração é o da existência de mais de um objeto do mesmo tipo na imagem, pois isso faz com que apenas um dos objetos sejam encontrados e será aquele com a menor distância.

4.1.6 Visualização tridimensional

Com a localização dos objetos obtida pela etapa de *matching*, a obtenção das coordenadas de cada ponto encontrado no espaço tridimensional é feita calculando-se a perspectiva inversa destes pontos utilizando os parâmetros da câmera que foram obtidos na etapa de calibração. A reconstrução de um sólido geométrico do modelo em estudo é efetivada com a definição das coordenadas de pelo menos três vértices distintos. Isto faz com que o processo finalize e permita a visualização destes objetos de outros ângulos.

4.2 Ambiente Integrado de Modelagem

Para auxiliar no processo de reconstrução, foi desenvolvido um ambiente de modelagem 3D utilizando o *Fox-ToolKit* (Fox-Toolkit 2007) para a construção das janelas, *OpenSceneGraph* (Osfield and Burns 2006) para a visualização dos objetos em 3D e a biblioteca *OpenCV* para toda etapa de processamento de imagens (Intel 2006).

Este ambiente de modelagem permite ao usuário realizar a reconstrução tridimensional do modelo de duas formas. Na primeira, é realizada a calibração da câmera com o fornecimento pelo usuário dos pontos de referência (conjuntos de pares de coordenadas cena-imagem) e dos pontos do objeto e, na segunda forma, o usuário só precisa informar os pontos de referência.

Ambas as formas têm como entrada uma imagem. Após carregar dentro do ambiente a imagem contendo os objetos do modelo, representando a cena no mundo real, é realizado o processo de captura dos pontos de referência. Isto é realizado pelo usuário que vai clicando na janela que contém a imagem. Após o clique nesta imagem, é aberta uma janela de captura de pontos de referência (Figura 4.5), onde o usuário relaciona uma coordenada da imagem com as coordenadas do mundo real (conjunto de coordenadas cena-imagem). Cada ponto clicado na imagem é marcado e recebe um *label* que é exibido ao lado dele, conforme ilustra a Figura 4.6.

Com os pontos capturados, o usuário seleciona a opção para calibrar a câmera que vai ter como resultado os parâmetros da câmera que geraram a imagem aberta.

4.2.1 Localização dos pontos dos objetos pelo usuário

Após a etapa de calibração, é realizada a reconstrução dos objetos, com o usuário identificando que componente deseja reconstruir na cena virtual, fornecendo os pontos dos objetos (Carvalho *et al.* 1998), utilizando uma barra de ferramentas com os objetos componentes do modelo (Figura 4.7).

Antes de realizar a reconstrução dos objetos, já é possível posicionar o plano do modelo com ajuda das informações dos parâmetros de câmera. Através de duas janelas, conforme a Figura 4.8 mostra, é possível visualizar a cena de duas maneiras: uma com a cena sobreposta pela imagem (Figura 4.8(a)) e outra com

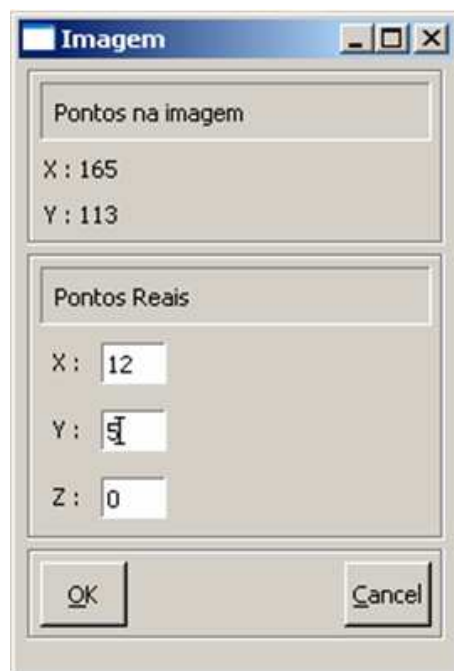


Figura 4.5: Janela de captura de pontos de referência cena-imagem.

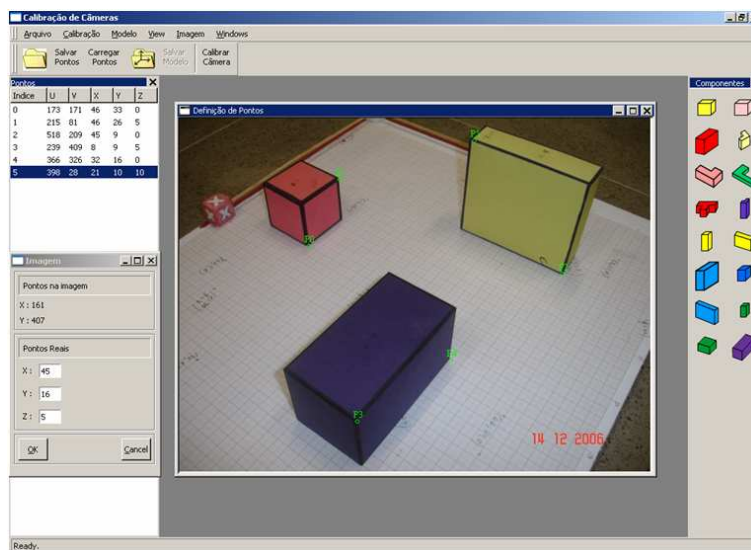


Figura 4.6: Etapa de fornecimento do conjunto de pontos de referência para efetuar a calibração.

a cena que pode ser observada de outros ângulos (Figura 4.8(b)).

À medida que é selecionado um componente, este é exibido em modo *wireframe*, conforme mostra a Figura 4.2, para que o usuário possa informar

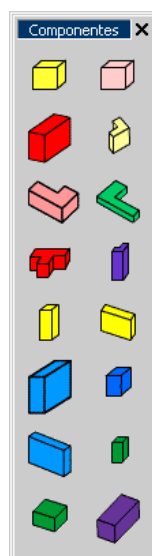


Figura 4.7: Barra de ferramentas com os objetos componentes do modelo.



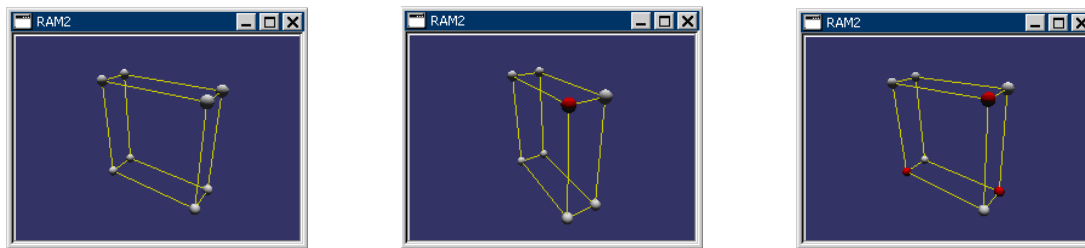
(a) Cena 3D com a imagem da cena sobreposta.

(b) Cena 3D com o plano do modelo já posicionado.

Figura 4.8: Janelas de Visualização 3D.

qual o vértice do objeto corresponde a que ponto na imagem, a fim de realizar a perspectiva inversa deste ponto e obter suas coordenadas do espaço tridimensional (Figura 4.9).

É necessário que sejam fornecidos pelo menos três coordenadas de vértices distintos de cada objeto para que estes sejam reconstruídos. Na Figura 4.10, é possível visualizar um objeto reconstruído.



(a) Componente com todos os seus vértices.

(b) Vértice selecionado.

(c) Três vértices selecionados.

Figura 4.9: Modelo *wireframe* de um dos objetos componentes do modelo.

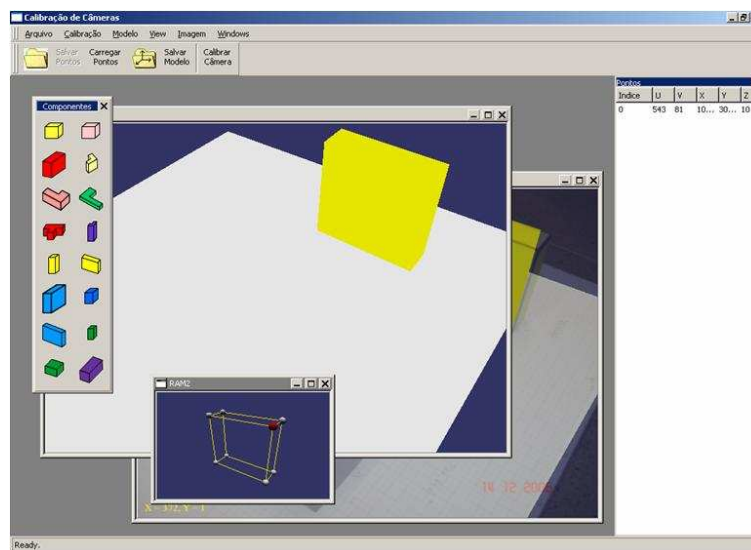


Figura 4.10: Fornecimento dos pontos do objeto e reconstrução de um componente selecionado.

4.2.2 Localização automática dos pontos dos objetos

Para localizar automaticamente os pontos dos objetos dentro da imagem e diminuir a intervenção do usuário no processo de reconstrução do modelo, é necessário, ao contrário do método anterior, utilizar mais 3 novas etapas, conforme foi descrito na seção anterior. São elas: etapa de Detecção de bordas e de vértices, de construção de *templates* e a etapa de *matching*.

Então, logo após a etapa de calibração, é selecionada dentro do ambiente de modelagem a opção detectar vértices da imagem. Como resultado do procedimento, é exibido a janela ilustrada na Figura 4.11, onde os vértices

encontrados pelo método de Harris são destacados por pontos na cor verde.

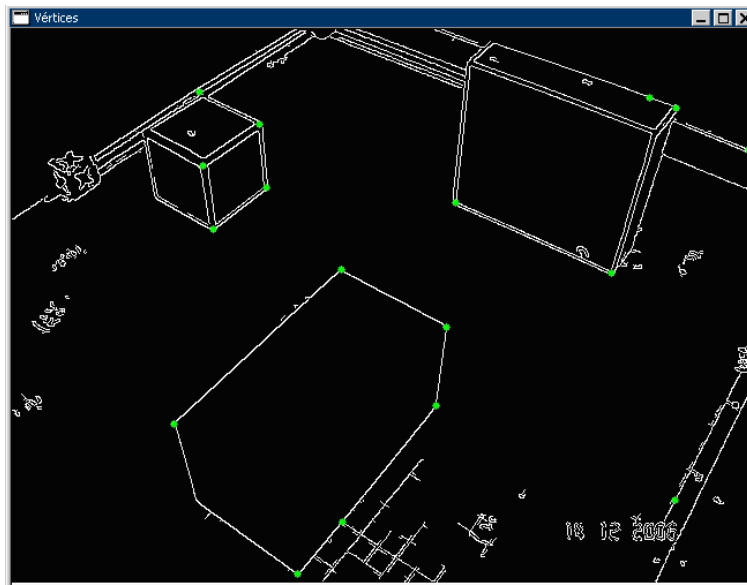


Figura 4.11: Etapa de detecção dos vértices dos objetos nas bordas da imagem.

Após a etapa de detecção dos vértices, é feita a reconstrução dos objetos com a intervenção do usuário. Este seleciona o objeto componente do modelo que deseja acrescentar à cena virtual na barra de ferramentas. À medida que é selecionado um objeto, é construído um *template* que representa este objeto, utilizando-se informações da câmera que gerou a imagem (parâmetros de câmera). Com este *template*, num processo automático, é identificado na imagem o local de *matching* que possua os vértices suficientes (pelo menos três) para construir o objeto selecionado (pontos do objeto). Na Figura 4.12, são ilustrados dois locais de *matching* referentes ao dois tipos de objetos com os pontos dos seus vértices em vermelho.

Na Figura 4.13, é possível visualizar o resultado final do processo de reconstrução do modelo através das duas janelas de visualização.

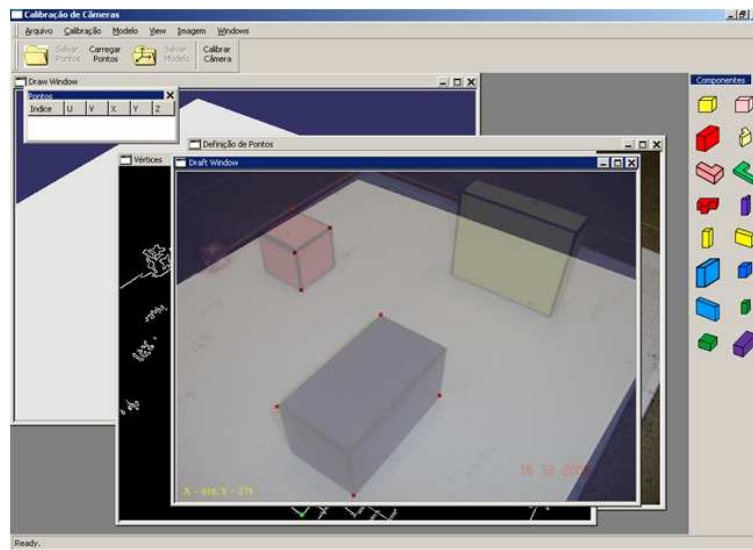


Figura 4.12: Obtenção dos locais de *matching*.

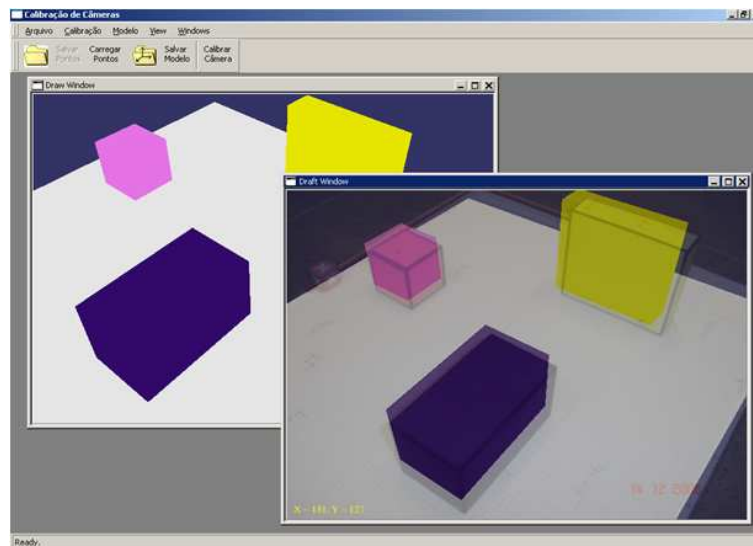


Figura 4.13: Reconstrução finalizada.

CAPÍTULO 5

CONCLUSÃO

A complexidade no desenvolvimento de software para visualização de cenas tridimensionais faz com que aumente o interesse na área de Modelagem Baseada em Imagem. Através do uso de diversas técnicas da área de Processamento de imagens em conjunto com área de Visão computacional, foi possível realizar neste trabalho um processo que cobre as diversas e complexas etapas (aquisição da imagem, pré-processamento, segmentação, representação/descrição e reconhecimento/interpretação, reconstrução e visualização) que possibilitam a reconstrução 3D, a partir de imagens 2D.

Ao contrário da Modelagem Baseada em Geometria, a Modelagem Baseada em Imagem diminui muito o tempo necessário para a construção de objetos em ambientes tridimensionais virtuais. Isto reduz a quantidade de recursos financeiros investidos no desenvolvimento de aplicativos, tornando-se uma tendência em muitas corporações que até então não utilizavam softwares para visualização tridimensional, devido aos altos custos de projeto, construção e manutenção.

No levantamento bibliográfico feito nesta pesquisa, vários trabalhos sobre Modelagem Baseada em Imagem foram vistos. Muitos destes trabalhos focam-se na busca por pontos dos objetos na imagem, como vértices e arestas, em geral apresentando ainda inúmeras etapas realizadas com a intervenção do usuário.

A principal característica do método proposto nesta pesquisa é a obtenção de forma automática dos pontos dos objetos na imagem adquirida através do uso da técnica de *template matching*. Mas, para que esta técnica obtivesse bons resultados, foi necessária a construção de *templates* utilizando-se os parâmetros

da câmera calibrada. Isto permitiu que tivéssemos um *template* diferente para cada nova posição da câmera. Consideramos esta a principal contribuição deste trabalho.

Para diminuir o tempo de processamento necessário para obtenção do local de *matching*, que constituem os pontos dos objetos, o uso do *template matching* foi melhorado com a realização de um pré-processamento nas imagens, identificando antes bordas e vértices para um melhor desempenho. A qualidade na identificação destes vértices na imagem depende diretamente da existência de poucos ruídos que podem influenciar na obtenção de vértices insuficientes para identificar um objeto na imagem ou na obtenção de vértices que não façam parte de nenhum objeto.

Em alguns casos, a obtenção do local de *matching* teve resultados que não permitiram fazer a reconstrução dos objetos da forma esperada. Isto compreende os casos em que existem mais de um objeto de um mesmo tipo. Por exemplo, se existirem dois cubos na imagem, só será possível identificar um deles, ou seja, aquele cujos vértices possuem as menores distâncias. Outro ponto que deve ser levado em consideração, para melhoria no processo de reconstrução, é o da obtenção da informação quanto à existência ou não de um determinado tipo de objeto na imagem que possui a visualização do modelo; em outros casos, objetos oclusos não puderam ser reconstruídos por falta de definição dos três vértices necessários para realizar a reconstrução de sólidos geométricos. Assim, prevê-se a necessidade de utilização de imagens com outros pontos de vista.

Os resultados podem ser considerados satisfatórios, mesmo com erros inerentes ao próprio processo de calibração e ao fornecimento dos pontos dos objetos, mas que resultam em uma precisão adequada.

Como extensão deste trabalho, pretende-se aperfeiçoar o método de reconhecimento dos pontos dos objetos e de encontrar uma maneira de solucionar os casos citados em que a reconstrução não obtém um resultado esperado. Pretende-se também incorporar o método de Zhang ao ambiente de modelagem para que o usuário selecione qual método será aplicado, durante a etapa de calibração. Na etapa de detecção de bordas e de vértices, serão feitos mais testes para encontrar limiares para a execução do filtro de Canny e do método de Harris de forma a melhorar a detecção dos vértices. Finalmente, pretende-se fazer aplicação desta técnica a uma classe de modelos do mundo real que apresentem

características semelhantes ao modelo ao qual a técnica foi aplicada, tais como subestações de distribuição de energia elétrica, conjuntos arquitetônicos, prédios, usinas hidrelétricas, usinas petrolíferas, dentre outros.

Referências Bibliográficas

- Azevedo, Teresa C. S., João M. R. S. Tavares and Mário A. P. Vaz (2005). Obtenção da forma 3d de objectos usando metodologias de reconstrução de estruturas a partir do movimento. *Congreso de Métodos Numéricos en Ingeniería*.
- Bardsley, Daniel (2004). A correlation based stereo vision system for face recognition applications. dissertação. university of nottingham.
- Carvalho, P.C.P. (2002). Visualização e modelagem baseada em imagens. *Com Ciência: Revista Eletrônica de Jornalismo Científico*.
- Carvalho, P.C.P., F. Szenberg and M. Gattass (1998). Image-based modeling using a two-step camera calibration method. In: *Proceedings of International Symposium on Computer Graphics, Image Processing and Vision*. IEEE Computer Society. New Orleans, LA, USA. pp. 388–395.
- Corseuil, Eduardo T. L. (2003). Buscando o uso operacional de realidade virtual em grandes modelos de engenharia. *Anais do VI Symposium on Virtual Reality*.
- Debevec, Paul (1996). Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In: *SIGGRAPH Conference Proceedings*. IEEE Computer Society. New Orleans, LA, USA. pp. 11–20.
- Deriche, R. and G. Giraudon (1993). A computational approach for corner and vertex detection. *The International Journal of Computer Vision* **10**(2), 101–124.

- do Vale, Giovane M., Edinéia A. dos S. Galvanin and Aluir P. D. Poz (2004). O detector de canny-edp: Uma combinação entre as teorias de canny e de difusão anisotrópica não linear. *Revista Brasileira de Cartografia* **2**(56), 156–168.
- El-Hakim, Sabry F., J.-Angelo Beraldin, Michel Picard and Guy Godin (2004). Detailed 3d reconstruction of large-scale heritage sites with integrated techniques. *IEEE Computer Graphics and Applications* **24**(3), 21–29.
- Faugeras, O. (1993). *Three Dimensional Computer Vision: A Geometric Viewpoint*. The MIT Press.
- Figueiredo, L.H. and P.C.P. Carvalho (1991). *Introdução à Geometria Computacional*. IMPA, SBM. Rio de Janeiro, RJ.
- Fox-Toolkit (2007). Fox development toolkit. *Disponível em: <http://www.fox-toolkit.org>. Acesso em: janeiro 2007.*
- Gonzalez, R. and R. E. Woods (1992). *Digital Image Processing*. Addison-Wesley Publishing Company.
- Gruen, A., F. Remondino and L. Zhang (2005). Modeling and visualization of landscape and objects using multi-resolution image data. *CIPA 2005 XX International Symposium*.
- Harris, C. and M. Stephens (1988). A combined corner and edge detection. In: *Proceedings of The Fourth Alvey Vision Conference*. pp. 147–151.
- Heckbert, P. S. (1986). Survey of texture mapping. *IEEE Comput. Graph. Appl.* **6**(11), 56–67.
- Huttenlocher, Daniel P., Gregory A. Klanderman and William Rucklidge (1993). Comparing images using the hausdorff distance. *IEEE Trans. Pattern Anal. Mach. Intell.* **15**(9), 850–863.
- Intel (2006). Intel open source computer vision library (opencv). *Disponível em: <http://www.intel.com/research/mrl/research/opencv>. Acesso em: dezembro 2006.*

- Jahne, Bernd (2002). *Digital Image Processing*. Springer Verlag.
- Kho, Kristen (2006). 3d photography. Technical report. Department of Computer Science and Engineering, University of California, San Diego. Disponível em: <http://www-cse.ucsd.edu/classes/wi07/cse190-a/reports/kmkho.pdf>. Acesso em: outubro 2006.
- Kirner, C. and M.S. Pinho (2006). Uma introdução à realidade virtual. Technical report. Grupo de Pesquisa em Realidade Virtual, Departamento de Computação, Universidade Federal de São Carlos - UFSCar. Disponível em: <http://www.dc.ufscar.br/grv/tutrv/tutrv.htm>. Acesso em: setembro 2006.
- Neoh, Hong Shan and Asher Hazanchuk (2004). Adaptive edge detection for real-time video processing using fpgas. *Global Signal Processing*.
- Osfield, R. and D. Burns (2006). Openscenegraph. *OSG Community*. Disponível em: <http://www.openscenegraph.org>. Acesso em: janeiro 2006.
- Osório, Fernando S. (2004). Ambientes virtuais interativos e inteligentes: Fundamentos, implementação e aplicações práticas. *Anais do XXIV Congresso da Sociedade Brasileira de Computação* **2**, 239–288.
- Sherman, William R. and Alan B. Craig (2003). *Understanding Virtual Reality: Interface, Application, and Design*. Morgan Kaufman. San Francisco.
- Sonka, Milan, Vaclav Hlavac and Roger Boyle (1994). *Image Processing, Analysis and Machine Vision*. 1nd ed.. Chapman & Hall.
- Szenberg, F. (2001). Acompanhamento de Cenas com Calibração Automática de Câmeras. PhD thesis. PUC-Rio.
- Tsai, Roger Y. (1987). A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation* pp. 323–344.
- Velho, Luis, Paulo C. P. Carvalho, Asla Sá and E. Medeiros (2005). *Fotografia 3D*. IMPA, SBM. Rio de Janeiro, RJ.

- Vázquez, Pere-Pau, Miquel Feixas, Mateu Sbert and Wolfgang Heidrich (2003). Automatic view selection using viewpoint entropy and its application to image-based modelling. *Computer Graphics forum* **22**(4), 689–700.
- Williams, Jeff and Won-Sook Lee (2006). Interactive virtual simulation for multiple camera placement. In: *Proceedings of the IEEE International Workshop on Haptic Audio Visual Environments and their Applications (HAVE 2006)*.
- Zhang, Zhengyou (2000). A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(11), 1330–1334.
- Zhang, Zhengyou (2001). Image-based modeling of objects and human faces. In: *Proceedings of Conference Videometrics and Optical Methods for 3D Shape Measurement (SPIE)*. Vol. 4309. San Jose, California, USA. pp. 21–26.