

UNIVERSIDADE ESTADUAL DE SANTA CRUZ  
DEPARTAMENTO DE CIÊNCIAS EXATAS E TECNOLÓGICAS  
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Ivan Cezanne Ismerim Bezerra de Meneses Seara

*Reconstrução de Objetos 3D a partir de Imagens 2D*

Ilhéus– Bahia

2014

Ivan Cezanne Ismerim Bezerra de Meneses Seara

*Reconstrução de Objetos 3D a partir de Imagens 2D*

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação da UESC, como requisito para a obtenção parcial do grau de BACHAREL em Ciência da Computação.

**Orientador: César Alberto Bravo Pariente**

**Doutor**

Ilhéus– Bahia

2014

Ivan Cezanne Ismerim Bezerra de Meneses Seara

## *Reconstrução de Objetos 3D a partir de Imagens 2D*

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação da UESC, como requisito para a obtenção parcial do grau de BACHAREL em Ciência da Computação.

Ilhéus, 26/11/2014.

### **BANCA EXAMINADORA**

---

César Alberto Bravo Pariente - Doutor

UESC/DCET

(Professor Orientador)

---

Vânia Cordeiro da Silva - Doutora

UESC/DCET

---

Marcelo Ossamu Honda - Doutor

UESC/DCET

*À família, aos amigos e aos mestres, dedico.*

## **Agradecimentos**

Agradeço à UESC, DCET, COLCIC e seus funcionários, magníficos reitores, diretores e coordenadores pelo empenho e sucesso no funcionamento e provimento do curso.

Agradeço ao NBCGIB pelo apoio infra-estrutural durante a execução deste trabalho.

Agradeço, imensamente, aos professores do curso de ciência da computação, contribuidores de conhecimento e experiência da maior valia.

Agradeço ao meu orientador, pelo apoio, esclarecimentos e coordenação; sem os quais este trabalho não se tornaria o que veio a ser.

Agradeço, ainda, aos amigos da TecnoJr e CACIC pelo trabalho, esforço e sucessos compartilhados durante estes anos de graduação.

Por fim, manifesto minha eterna gratidão à minha família - base primordial do meu ser, origem de toda a força, motivação e inspiração, tão fundamentais nesta jornada que agora se encerra.

## Resumo

O processo de formação de imagens consiste em projetar regiões visíveis de uma cena em um plano bidimensional, desta forma consegue-se registrar um ambiente tridimensional em um outro bidimensional. Na bibliografia sobre computação gráfica encontram-se vários métodos desta captura, de forma digital. Neste trabalho propõe-se uma técnica capaz de criar uma nuvem de pontos tridimensionais que represente um objeto ora apresentado por uma única imagem bidimensional. Utilizando imagens de apelo histórico à computação gráfica e à arte, apresenta-se uma forma de identificação de um sistema de coordenadas de referência particular a cada cena representada na imagens, de forma que seja possível o estudo analítico da constituição espacial dos objetos de interesse na cena. Referenciadas as cenas, pôde-se obter uma transformação projetiva que implemente uma correspondência entre pontos 2D de uma imagem com pontos 3D de um sistema de referências. Por fim, obteve-se um software que permite um referenciaamento aproximado da cena, implicando em uma reconstrução coesa, porém, com certas limitações.

Palavras-chaves: Visão Computacional, Processamento Digital de Imagens, Reconstrução 3D

*“Há um ditado que ensina ‘o gênio é uma grande paciência’; sem pretender ser gênio, teimei em ser um grande paciente. As invenções são, sobretudo, o resultado de um trabalho teimoso, em que não deve haver lugar para o esmorecimento”.*

*Alberto Santos Dumont*

# Sumário

<b>Lista de Tabelas</b>	x
<b>Lista de Figuras</b>	xi
<b>1 Introdução</b>	1
1.1 Justificativa . . . . .	1
1.2 Problemática . . . . .	2
1.3 Objetivos . . . . .	3
<b>2 Estado da Arte</b>	4
2.1 Múltiplas Visões e Geometria Epipolar . . . . .	4
2.1.1 Cinema 3D através de Estereoscopia . . . . .	5
2.1.2 Um Abordagem Mais Simples da Estereoscopia . . . . .	5
2.2 Calibração de Câmera e Geometria Projetiva . . . . .	6
<b>3 Materiais e Métodos</b>	7
3.1 Recursos Computacionais . . . . .	7
3.2 Conjunto de Dados . . . . .	8
3.3 Aferição da corretude . . . . .	10
<b>4 Fundamentação Teórica</b>	12
4.1 Projeções . . . . .	12
4.1.1 Projeção Paralela . . . . .	12
4.1.2 Projeção Cônica . . . . .	13
4.2 Câmera Virtual . . . . .	14

4.3	Calibração de câmera . . . . .	15
<b>5</b>	<b>A Técnica de Reconstrução</b>	<b>18</b>
<b>6</b>	<b>Aplicativos Desenvolvidos</b>	<b>22</b>
6.1	Aplicativo de detecção de retas . . . . .	22
6.2	Aplicativo de Projeção 3D . . . . .	23
6.3	Aplicativo de Reconstrução . . . . .	25
<b>7</b>	<b>Resultados, Discussões e Trabalhos Futuros</b>	<b>27</b>
7.1	Testes validadores . . . . .	27
7.2	Resultados . . . . .	30
7.3	Discussões . . . . .	34
7.4	Trabalhos Futuros . . . . .	35
	<b>Referências Bibliográficas</b>	<b>37</b>

## **Lista de Tabelas**

7.1	Dados da reconstrução da cena de validação 1 . . . . .	28
7.2	Dados da reconstrução da cena de validação 2 . . . . .	28
7.3	Dados da reconstrução da cena de validação 3 . . . . .	28
7.4	Erros geométricos médios e desvios-padrão dos testes validadores . . . . .	29
7.5	Dados da reconstrução do <i>VW Beetle</i> de Sutherland . . . . .	32
7.6	Dados da reconstrução da Litografia de Dürer . . . . .	33

# **Lista de Figuras**

2.1	Triangulação utilizada na geometria epipolar . . . . .	5
3.1	O <i>VW Beetle</i> de Sutherland . . . . .	9
3.2	O Bule de Leite de Newell . . . . .	9
3.3	A litografia de Dürer em (BRUNELLESHI'S..., 1998) . . . . .	10
3.4	Cenas para validação da técnica de reconstrução . . . . .	11
4.1	Projeção paralela em (POTTMANN et al., 2007) . . . . .	13
4.2	Projeção cônica em (POTTMANN et al., 2007) . . . . .	14
5.1	Os pontos $P_i$ pertencem à reta de projeção de $P$ , mas suas coordenadas $Z$ são diferentes da coordenada $Z$ do ponto $P$ . . . . .	19
5.2	É perceptível que os pontos demarcados com um "X" têm a mesma coordenada $Z$ no espaço . . . . .	19
6.1	O programa de regressão de rasterização com uma aproximação de exemplo	23
6.2	Aplicativo de projeção 3D com a famosa malha Suzanne . . . . .	24
6.3	Opções adicionais do aplicativo de projeção 3D de auxílio à visualização .	24
6.4	Tela inicial do aplicativo de reconstrução . . . . .	25
6.5	Utilização da interface para a demarcação de pontos de referência e interesse	25
6.6	<i>Prompt</i> com dados necessários à reconstrução . . . . .	26
6.7	<i>Prompt</i> para escolha manual da coordenada Z . . . . .	26
7.1	Panoramas das reconstruções de validação . . . . .	29
7.2	Reconstruções validadoras (faces inseridas manualmente) . . . . .	30
7.3	Panorama da reconstrução do <i>VW Beetle</i> de Sutherland . . . . .	31

7.4	A reconstrução do <i>VW Beetle</i> de Sutherland (Faces inseridas manualmente)	31
7.5	Panorama da reconstrução da litografia do Dürer	32
7.6	A reconstrução da cena da litografia de Dürer (Faces inseridas manualmente)	33
7.7	Referenciamento na imagem do Bule de Newell	34

# 1 Introdução

## 1.1 Justificativa

O processo fotográfico, surgido no século XIX, determinou um marco na forma de registrar a realidade humana, sendo capaz de capturar a luz proveniente do ambiente e convertê-la em regiões semelhantes sobre uma superfície manuseável chamada de fotografia. Tal processo tem sido motivo de estudo de diversas áreas do conhecimento e, hoje, já é possível executar tal processo, outrora baseado em películas fotossensíveis, de forma digital e computacional.

Uma das áreas de estudo das imagens digitais é a visão computacional - campo da ciência da computação que discute acerca da interpretação e extração de conhecimento de imagens. Desde a década de 1960, a visão computacional apresenta trabalhos interessantes acerca de algoritmos de emulação de visão humana e interpretação de imagens, vide (KUBERT; SZABO; GIULIERI, 1968). Derivado disto, hoje é possível a construção de ferramentas úteis para a compreensão via *software* de cenas retratadas em imagens.

Fato é que, o processo de formação de uma imagem digital funciona baseado em um processo de projeção de regiões visíveis de um ambiente tridimensional - o universo real - em uma superfície plana bidimensional - a imagem. Facilmente se percebe que tal técnica consiste em reduzir em uma dimensão o meio de representação de objetos de uma cena. Apesar de avanços tecnológicos neste processo permitirem a reprodução fiel das propriedades da luz, facilitando a noção do observador da imagem acerca dos aspectos físicos tridimensionais (dimensões, distâncias etc.) da cena retratada, os dados que representam tais aspectos foram truncados. Tais dados tridimensionais truncados podem apresentar utilidade para áreas da sociedade que têm nas imagens armazenadas sua fonte de conteúdo. Áreas como conservação e restauro de patrimônio arquitetônico, serviços de GPS e mapas e investigação de cenários reais trabalham constantemente com representações bidimensionais de objetos tridimensionais (edifícios, avenidas urbanas, cômodos de residências mobiliados etc.) onde o sucesso de suas tarefas depende da compreensão espacial do seu objeto de estudo.

De tal forma, a existência de um processo digital reverso ao processo da projeção, capaz de reconstruir objetos 3D a partir de imagens 2D, representa um avanço significativo na capacidade de se representar a realidade em forma digital.

## 1.2 Problemática

O *VW Beetle* de Sutherland entrou para a história da computação gráfica por ser o primeiro objeto real digitalizado em três dimensões. Em um processo liderado pelo professor Ivan Edward Sutherland, um grupo de alunos demarcou manualmente, com giz, sobre a carroceria de um veículo Volkswagen Beetle 1967, pontos de interesse para a construção do modelo 3D do carro (MUSEUM, 2014a).

Tendo a experiência sido bem sucedida, um modelo tridimensional foi obtido. No entanto a manutenção dos dados deste modelo é nebulosa, não se sabe ao certo onde obter uma versão correta deste modelo ao passo que já se questiona se os seus dados originais estão preservados para fins de acesso público.

No entanto, a própria Universidade de Utah - sede do experimento - mantém as imagens deste modelo, sob a curadoria do *Computer History Museum*, além de existirem registros individuais com parâmetros úteis a este trabalho (vide capítulo 3). Assim, há a possibilidade de se restaurar os dados tridimensionais deste modelo 3D digital icônico.

Em um caso similar, o Bule de Leite de Newell é, originalmente, parte integrante de um famoso modelo tridimensional chamado *Teaset* de autoria do pesquisador Martin Edward Newell. O modelo *Teaset* pode ser encontrado facilmente em repositórios na *internet*, no entanto, em nenhuma das fontes consta o bule de leite. Sabe-se que tal objeto era parte integrante do conjunto *Teaset* por abordagens e imagens de trabalhos do próprio autor (NEWELL, 1975).

Por tal circunstância, percebe-se que o Bule de Leite de Newell também se trata de um elemento de apelo histórico ao campo de estudo deste trabalho cujo conhecimento é interessante e não tem seus dados tridimensionais publicados.

Em última instância, há uma litografia datada do século XVII de autoria do artista Albrecht Dürer que representa uma cena onde os personagens exemplificam a implementação da projeção perspectiva (vide seção 4.1 sobre as projeções). Tal cena, por ser uma documentação pioneira do estudo da perspectiva, apresenta uma oportunidade

de trabalhar sob dados antigos e fornecer uma previsão de como ampliar este trabalho de conclusão de curso para a reconstrução 3D de cenários com mais de um objeto.

### 1.3 Objetivos

Em face da problemática disposta, este trabalho visa, em âmbito específico, obter um conjunto de pontos tridimensionais que represente o *VW Beetle* de Sutherland, o Bule de Leite de Newell e alguns elementos da cena de Dürer, de forma que seja possível a visualização destes objetos em um software de visualização 3D.

Para a obtenção de sucesso no objetivo geral, os seguinte objetivos específicos foram estabelecidos:

- definir um método de identificação de sistemas de coordenadas de referências nas imagens estudadas;
- estabelecer uma correspondência entre pontos das imagens com pontos tridimensionais; e
- inferir informações de padronização deste objetos, tais como simetrias e repetições.

## 2 Estado da Arte

Com o uso de *hardwares* elaborados (câmeras sincronizadas ou câmeras de percepção volumétrica), definição estrita de objeto de reconstrução e distintas vertentes geométricas, a reconstrução 3D já é uma prática difundida; sendo possível encontrar um grande número de artigos acadêmicos sobre o assunto, além de trabalhos de empresas de entretenimento famosas que implementam reconstrução 3D, cada um de uma forma particular.

Neste capítulo é exposta uma breve coleção de exemplos representativos desta difusão da reconstrução a fim de configurar o cenário onde este trabalho de conclusão de curso se encaixa.

### 2.1 Múltiplas Visões e Geometria Epipolar

Uma abordagem muito comum para a implementação de um sistema de reconstrução 3D é o uso de diferentes pontos de vista sobre um mesmo objeto. A aquisição de dados de um objeto sob diferentes pontos de vista pode ser feita através de sistemas com múltiplos dispositivos de captura (câmeras) devidamente sincronizados e calibrados ou com imagens já obtidas, desde que seja conhecido um certo número de dados tridimensionais e dados de correspondência. Câmeras utilizadas para obtenção de múltiplas visões de um objeto são conhecidas como câmeras estéreo e tal técnica baseada em múltiplas visões, de estereoscopia.

Observando relações de correspondência entre pontos de distintas imagens, o uso da geometria epipolar fornece meios de concluir uma localização tridimensional a partir de dados bidimensionais. Segundo Aguiar (2010), a geometria epipolar utiliza a técnica de triangulação entre os centros de projeção - que são pontos tridimensionais - de duas câmeras e o ponto de interseção - que também é tridimensional e pode ser chamado de epipolo - entre as retas formadas pelos centros de projeção das câmeras e o ponto da imagem em análise (AGUIAR, 2010).

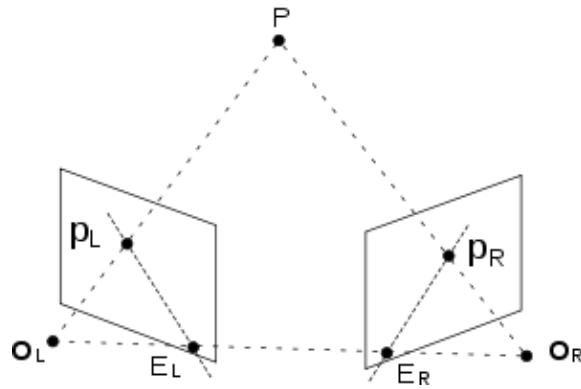


Figura 2.1: Triangulação utilizada na geometria epipolar

### 2.1.1 Cinema 3D através de Estereoscopia

Em (HEINZLE et al., 2011) vê-se a sugestão de um complexo sistema motorizado de câmeras de vídeo com um ajuste automático de parâmetros de calibração para maior precisão na determinação de características tridimensionais obtidas através do sistema de câmeras estéreo proposto.

Esta abordagem é efetiva para os autores dada a sua finalidade cinematográfica; e já que filmes em 3D são cada vez mais comuns nas salas de cinema em todo mundo, o investimento feito para obter a precisão que este trabalho alcança é facilmente compensado.

Em um contexto mais amplo, que não suporta uma infra-estrutura complexa de uma filmagem dos estúdios de cinema internacionais ou demanda um dinamismo maior, o uso de um sistema motorizado de câmeras em sincronia é uma estratégia pouco eficiente, apesar da sua grande eficácia.

### 2.1.2 Um Abordagem Mais Simples da Estereoscopia

Assim como Neves & Tsuzuki (2010), pode-se utilizar uma abordagem mais enxuta da reconstrução 3D através de estereoscopia, sob a premissa de que as dimensões do objeto a ser reconstruído são conhecidas. (NEVES; TSZUKI, 2010) Tal premissa, mesmo válida para o trabalho citado, reduz drasticamente o escopo da técnica sugerida.

Usando um único par de imagens estereoscópicas estáticas e as dimensões do objeto estudado, ainda é preciso efetuar alguns processos de normalização a ajuste algébrico para que condições de convergência, como o epipolo, possam ser obtidas. Tais

empecilhos, inclusive, são decorrentes de uma análise do objeto em termos de coordenadas absolutas, o que é irrelevante para a análise de um objeto isolado cuja reconstrução não faz parte de um cenário igualmente importante.

Assim, a reconstrução 3D de Neves & Tszuki (2010) ainda não se adapta completamente ao objetivo deste trabalho.

## 2.2 Calibração de Câmera e Geometria Projetiva

Como definem Gomes & Velho (2008), um conceito conhecido na computação gráfica é o de câmera virtual - um modelo matemático computacional que representa o funcionamento de um dispositivo de captura (GOMES; VELHO, 2008a).

A técnica de calibração de câmera consiste em estimar os parâmetros de uma câmera virtual capaz de produzir uma imagem. Uma vez que imagens são produzidas através de projeções, com o uso da geometria projetiva é possível haver dedução, sob certas hipóteses, de dados 3D após a análise de uma única imagem 2D.

Em (LEE; FENG; GOOCH, 2008), após um processo de obtenção dos parâmetros de uma câmera, é possível transformar polígonos pertencentes ao plano da imagem em polígonos equivalentes pertencentes a planos tridimensionais distintos do plano da imagem. Infelizmente, esta referência não era conhecida tão cedo quanto desejado, impedindo que este trabalho pudesse ter maior embasamento e robustez baseado em tal referência. No entanto, trabalhos parecidos sobre calibração de câmera para reconstrução 3D, que, estes sim, serviram como embasamento inicial, estão presentes em (CARVALHO; SZENBERG; GATTASS, 1998) e (SZENBERG, 2001)

## 3 Materiais e Métodos

Para a concretização deste trabalho foram necessários materiais tanto de ordem computacional, para processamento e implementação de software, tanto de ordem de dados, principalmente para teste das hipóteses encontradas, verificação de sucesso e embasamento teórico.

Os materiais em forma de dados consistiram, basicamente, em imagens de objetos isolados e, de preferência, com elementos dispensáveis como plano de fundo e texturas padronizados e sistema ortogonal de coordenadas de referência, a fim de facilitar o processamento dos dados pelo computador e o funcionamento dos algoritmos desenvolvidos. Consegiu-se um número suficiente, para testes, de imagens que correspondiam a este padrão desejado através de repositórios *on-line*.

A demanda por recursos computacionais de alta eficiência foi relativamente baixa, portanto, pôde-se utilizar o computador pessoal do autor sem prejuízo de tempo. Além disto, foram utilizadas ferramentas computacionais de terceiros, como bibliotecas de linguagens de programação e ambientes de edição de código-fonte, sob licença acadêmica gratuita, portanto, não acarretando em custos financeiros para a execução do trabalho.

### 3.1 Recursos Computacionais

Não havendo grande volume de dados ou alta complexidade no processamento dos dados, foi utilizado o computador pessoal *Notebook Mobile Sim+ 6220* do autor do trabalho, com a seguinte configuração:

- processador Intel®Core®i3 330M (2,13 GHz, 3MB Cache);
- memória RAM Kingston®PC3-8500 (4MB, DDR3);
- chipset gráfico Intel®HM55;
- sistema operacional Windows 7®.

Para fins de construção de programas de computador, otimização do rendimento dos códigos-fonte em relação ao *hardware* disponível, confecção de testes e análise de resultados, foram usadas as seguintes ferramentas computacionais:

- ambiente gráfico para programação em linguagem C# Microsoft Visual Studio 2010 v4.5.50938 RTMRel®;
- biblioteca de métodos algébricos em C# ALGLIB Free Edition v3.8.2;
- software de visualização 3D Meshlab v1.3.3;
- software de modelagem 3D Blender v2.70.

## 3.2 Conjunto de Dados

Os dados utilizados como entrada nos algoritmos e programas desenvolvidos neste trabalho consistem em imagens digitais representando objetos apresentados no capítulo 1, o *VW Beetle* de Sutherland (figura 3.1), o Bule de Leite de Newell (figura 3.2) e uma litografia digitalizada de Dürer (figura 3.3). Todas as imagens utilizadas neste trabalho foram obtidas respeitando os direitos autorais e de reprodução.

Quanto ao *VW Beetle* de Sutherland, conseguiu-se obter uma imagem com sistemas ortogonais de coordenadas de referência graficamente evidentes (figura 3.1c), no entanto todas as imagens apresentavam-se em projeção em perspectiva, distorcendo as linhas do sistema de referência. Todas as imagens referentes a este objeto apresentavam regiões desprezíveis, como plano de fundo, facilmente diferenciáveis através de cor. Em formato ‘JPG’, as imagens possuíam resolução de 72 dpi, 24 bits para a intensidade de cor e dimensões de 970 *pixels* para a largura e 646 *pixels* para altura.

As imagens do Bule de Leite de Newell, por este ser um objeto sem registro oficial dos seus dados, são menos elaboradas, com o sistema de coordenadas de referência mais nebuloso, mas, apresentando projeção paralela isométrica. Tais imagens são em branco e preto, em formato ‘PNG’ com 32 bits para intensidade de cor, resolução de 95 dpi, 609 *pixels* de largura e 582 *pixels* de altura.

A litografia de Dürer, digitalizada e disponibilizada por terceiros, possui um sistema de referências implícito em objetos da cena e, como se pode prever, está representada através da projeção em perspectiva. Em formato ‘JPG’ com 96 dpi de resolução,

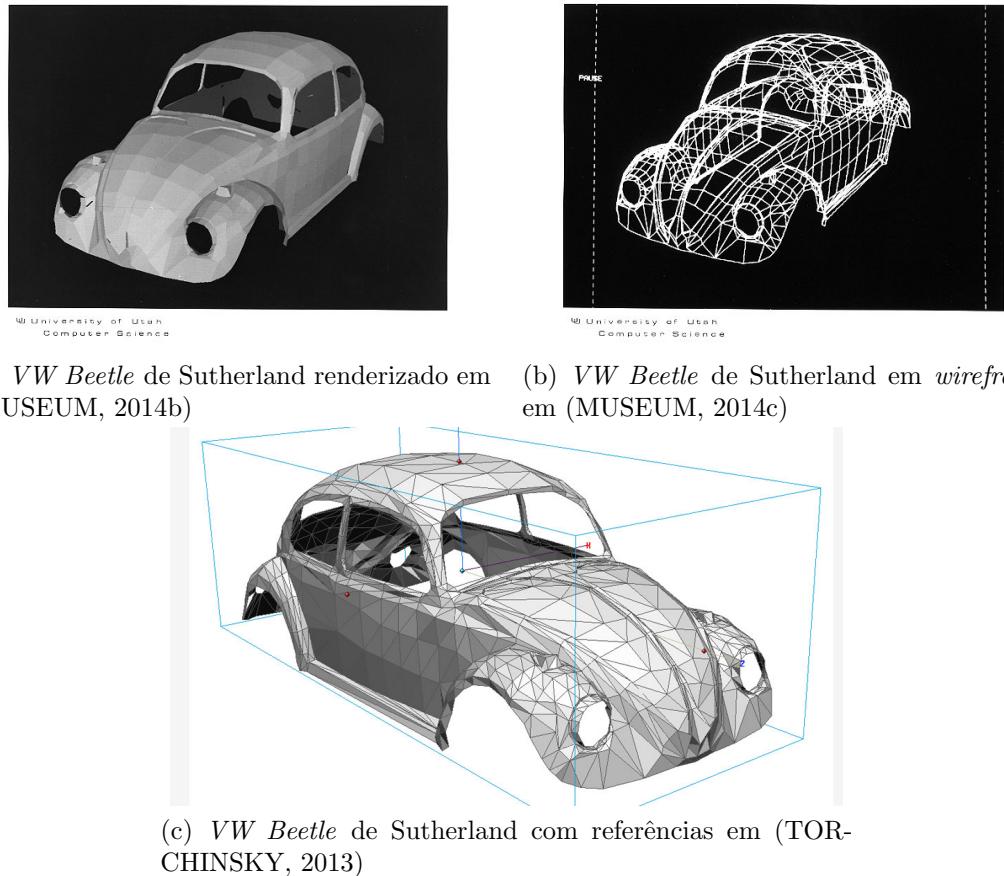


Figura 3.1: O VW Beetle de Sutherland

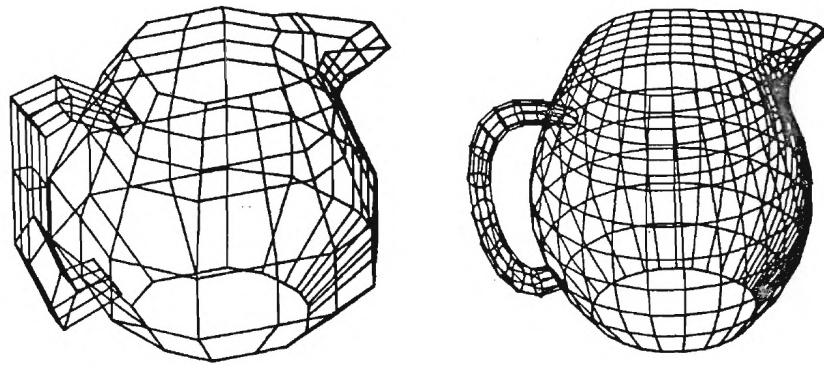


Figura 3.2: O Bule de Leite de Newell

24 bits de profundidade de cor e dimensões de 782 e 600 *pixels* para largura e altura, respectivamente.

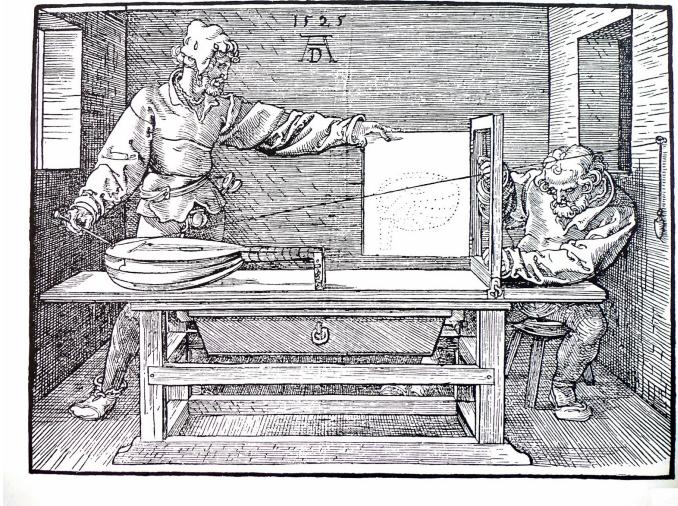


Figura 3.3: A litografia de Dürer em (BRUNELLESCHI'S..., 1998)

### 3.3 Aferição da corretude

Como o objetivo deste trabalho é reconstruir objetos cujas informações tridimensionais são inexistentes, é impossível fazer uma validação direta dos resultados obtidos, isto é, não há como comparar os modelos tridimensionais obtidos neste trabalho com um modelo ”controle” dos mesmos objetos estudados.

De tal forma, a corretude dos modelos obtidos foi obtida de forma implícita, através da validação da técnica de reconstrução deste trabalho. Utilizando cenas específicas confeccionadas pelo autor (figura 3.4) através de software Blender®, será estimado um erro geométrico  $E$  (equação 3.1) entre os pontos  $P$  de coordenadas conhecidas dos modelos e os pontos  $P'$  de coordenadas obtidas pela técnica desenvolvida neste trabalho.

$$E = \sqrt{(x_{P_i} - x_{P'_i})^2 + (y_{P_i} - y_{P'_i})^2 + (z_{P_i} - z_{P'_i})^2} \quad (3.1)$$

Com a cena de validação 1 (figura 3.4a) deseja-se a reconstrução dos vértices de um hexaedro de dimensão unitária; com a cena de validação 2 (figura 3.4b) almejam-se os vértices de uma pirâmide regular equilátera de dimensão unitária; por fim, com a cena de validação 3, buscam-se os vértices de um cubo de dimensão 0.5.

Na cena de validação 1 (figura 3.4a), um cubo (hexaedro) teve duas de suas faces removidas, a fim facilitar a demarcação de pontos de referência, originando a figura vista. No caso da figura 3.4b, há um quadrado de cor azul paralelo ao plano da base

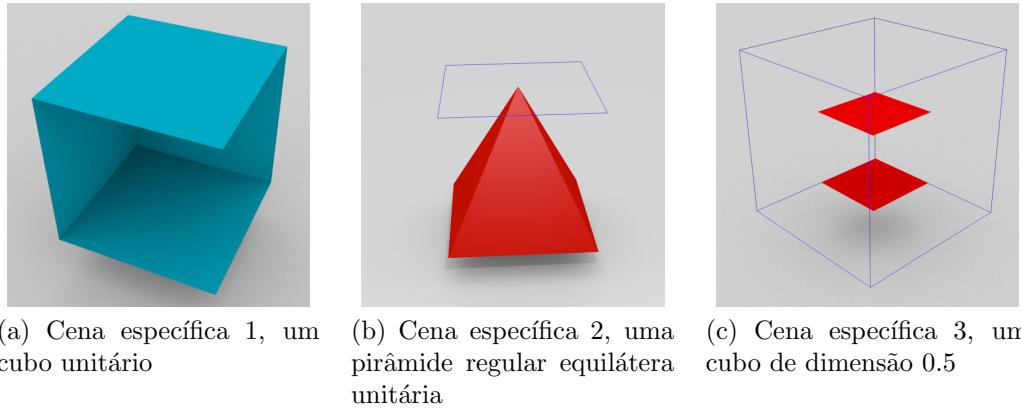


Figura 3.4: Cenas para validação da técnica de reconstrução

da pirâmide, tal quadrado é um auxílio, inserido pelo autor, para a demarcação dos pontos de referência e também será desconsiderado na reconstrução. A figura 3.4c, por sua vez, apresenta as arestas (em azul) de um cubo unitário, estas são um auxílio para a demarcação de referências; note que nesta cena também foram removidas faces do cubo para facilitar demarcações.

Todas as imagens das cenas de validação têm resolução de 72 dpi e 540 *pixels* de largura e altura.

## 4 Fundamentação Teórica

Neste trabalho de conclusão de curso, são usados alguns conceitos e técnicas da computação gráfica de forma adaptada ao objetivo proposto. Nas próximas seções são explicados os conceitos utilizados e suas respectivas aplicações no trabalho desenvolvido.

### 4.1 Projeções

Inicialmente, projeções são representações de objetos sob as restrições de um determinado ambiente. Em geometria analítica usam-se projeções para a descrição de vetores (as componentes do vetor são as projeções deste em cada uma das bases do sistema ordenado); no cotidiano, sombras geradas pelo nosso corpo no chão, por exemplo, é um caso específico de projeção (representação) do nosso corpo (objeto) na superfície plana do chão (ambiente) (POTTMANN et al., 2007).

Em computação gráfica, as projeções servem para representar objetos tridimensionais em dispositivos de visualização bidimensionais. Dependendo da utilização da visualização, diferentes formas de projetar o objeto tridimensional podem ser usadas (POTTMANN et al., 2007); as duas principais são a projeção paralela e a projeção cônica.

Independente do tipo de projeção, segundo Carvalho & Velho (2005), com apenas uma imagem não se pode estabelecer uma relação unívoca entre pontos no plano de projeção e pontos no espaço (CARVALHO; VELHO, 2005). No entanto, relações de isometria (seção 4.1.1) ou proporcionalidade (seção 4.1.2) podem incutir restrições aos pontos no espaço de forma a reduzir tal multiplicidade.

#### 4.1.1 Projeção Paralela

A projeção gráfica pode ser feita através de linhas paralelas entre si que ligam pontos do objeto ao plano de projeção (figura 4.1); esta projeção, segundo Pottmann *et al.* (2007), é chamada de projeção paralela (POTTMANN et al., 2007). Neste caso, conforme Gomes & Velho (2008), são mantidas as proporções entre as dimensões do objeto estudado

(isometria) (GOMES; VELHO, 2008a).

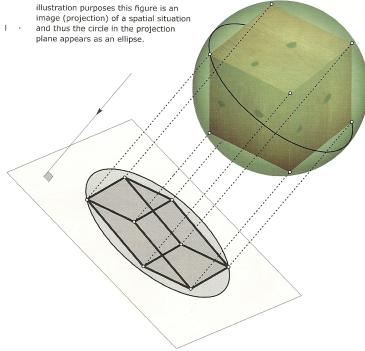


Figura 4.1: Projeção paralela em (POTTMANN et al., 2007)

A isometria é uma característica interessante deste tipo de projeção pois todas as condições de paralelismo são mantidas.

A transformação de um ponto 3D  $P = (X_P, Y_P, Z_P)$  em um ponto 2D  $P'$  sob a projeção paralela é dada pela equação 4.1, em coordenadas baseadas no Sistema de Coordenadas da Câmera (CARVALHO; VELHO, 2005). Como pode-se notar, a projeção de  $P$  independe da sua distância -  $Z_P$  - ao plano de projeção e isto causa a multiplicidade projetiva.

$$P' = (X_P, Y_P) \quad (4.1)$$

### 4.1.2 Projeção Cônica

Com o intuito de representar um objeto de forma similar à realidade, é aplicada a projeção cônica (ou perspectiva) (POTTMANN et al., 2007). Tal projeção provoca uma deformação linear (equação 4.2), proporcional à distância  $Z_P$  ao plano de projeção, na posição planar do ponto  $P = (X_P, Y_P, Z_P)$ , que também está expresso em coordenadas do Sistema de Coordenadas da Câmera (CARVALHO; VELHO, 2005). Isto se dá pois a projeção  $P'$  de  $P$  é a interseção da reta que une o ponto 3D  $e$  ao ponto  $P$  (figura 4.2), sendo  $e$  o centro de projeção, que funciona de forma análoga ao olho de um observador humano (POTTMANN et al., 2007).

$$P' = \left( \frac{X_P}{Z_P}, \frac{Y_P}{Z_P} \right) \quad (4.2)$$

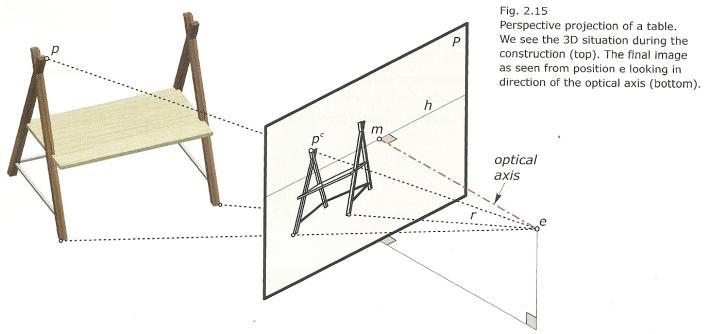


Figura 4.2: Projeção cônica em (POTTMANN et al., 2007)

A deformação provocada pela projeção cônica impossibilita a existência da isometria, no entanto, o processo de calibração de câmera virtual (seções 4.2 e 4.3) possibilita relações entre pontos 2D e 3D (CARVALHO; SZENBERG; GATTASS, 1998).

## 4.2 Câmera Virtual

A operação básica de uma câmera virtual é a projeção (GOMES; VELHO, 2008a). A fim de sintetizar uma imagem de um objeto digital com três dimensões, programas de visualização 3D implementam um modelo matemático que utiliza as técnicas discutidas anteriormente (seção 4.1) para a transformação de 3D para 2D.

Neste trabalho optou-se por desenvolver um modelo de câmera que projeta apenas em perspectiva, dada a ausência, inicialmente, de bibliografia sobre modelos de câmeras de projeção paralela. No entanto, na seção 7.3 é exposta uma possibilidade de retificação desta limitação.

Para um modelo altamente fiel à câmera real, devem ser feitas correções acerca de distorções de lentes (AGUIAR, 2010) além de que uma câmera digital real, em termos geométricos, precisa converter as coordenadas de um objeto sob um sistema de referências global para um sistema de coordenadas em *pixels* passando por um sistema de coordenadas de câmera e de imagem, processos chamados de Transformações de Câmera (CARVALHO; VELHO, 2005). Para este trabalho, não há distorção de lentes nas imagens (uma vez que são artificiais).

Em (CARVALHO; VELHO, 2005) é descrita cada transformação entre cada sistema de coordenadas. No entanto, neste trabalho todas as transformações de câmera foram agrupadas, como em (CARVALHO; SZENBERG; GATTASS, 1998) e (GOMES;

VELHO, 2008a), uma vez que apenas interessa a transformação final e não os parâmetros explícitos da câmera. Assim, o modelo final de câmera  $C$  utilizado foi o apresentado na equação 4.3.

$$C = \begin{pmatrix} q_{1,1} & q_{1,2} & q_{1,3} & q_{1,4} \\ q_{2,1} & q_{2,2} & q_{2,3} & q_{2,4} \\ q_{3,1} & q_{3,2} & q_{3,3} & q_{3,4} \end{pmatrix} \quad (4.3)$$

De forma que, aplicando  $C$  a um ponto 3D, em coordenadas homogêneas (GOMES; VELHO, 2008b) sob o Sistema de Coordenadas do Mundo (CARVALHO; VELHO, 2005), obter-se-á diretamente o respectivo ponto 2D em termos de *pixel* na imagem.

### 4.3 Calibração de câmera

Costumeiramente, o modelo de câmera virtual é construído a partir dos parâmetros (intrínsecos e extrínsecos) (GOMES; VELHO, 2008a) da câmera; pode-se dizer que esta é a forma direta de especificação da câmera.

Neste trabalho, um dos empecilhos era o desconhecimento destes atributos da câmera, mas sem tal conhecimento não há como construir a transformação 4.3. No entanto, com uma correspondência previamente conhecida entre pontos da imagem e pontos 3D, o modelo da câmera pode ser estimado através de um processo chamado de Calibração de Câmera (costuma-se chamar, também, de *Resectioning*, já que calibração pode envolver, ainda, ajuste cromático). (CARVALHO; SZENBERG; GATTASS, 1998), (SZENBERG, 2001) e (LANGER, 2010b) aplicam apenas a calibração geométrica e usam o termo "Calibração".

Como dito na seção 4.2, a aplicação do modelo de câmera em pontos 3D  $P = (X, Y, Z)$  leva a pontos 2D  $P' = (u, v)$  da imagem; de forma que, segundo (CARVALHO; SZENBERG; GATTASS, 1998), têm-se para cada correspondência de pontos 3D com 2D as equações 4.4 e 4.5.

$$u = \left( \frac{q_{1,1} \cdot X + q_{1,2} \cdot Y + q_{1,3} \cdot Z + q_{1,4}}{q_{3,1} \cdot X + q_{3,2} \cdot Y + q_{3,3} \cdot Z + q_{3,4}} \right) \quad (4.4)$$

$$v = \begin{pmatrix} q_{2,1} \cdot X + q_{2,2} \cdot Y + q_{2,3} \cdot Z + q_{2,4} \\ q_{3,1} \cdot X + q_{3,2} \cdot Y + q_{3,3} \cdot Z + q_{3,4} \end{pmatrix} \quad (4.5)$$

Como a matriz  $C$  da câmera (equação 4.3) tem 12 graus de liberdade, são necessários, no mínimo, 6 correspondências, com pontos 3D linearmente independentes, para que se monte um sistema dado por 4.6 possível de ser resolvido (LANGER, 2010b).

$$\begin{pmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & -u_1 X_1 & -u_1 Y_1 & -u_1 Z_1 & -u_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -v_1 X_1 & -v_1 Y_1 & -v_1 Z_1 & -v_1 \\ \vdots & \vdots \\ X_n & Y_n & Z_n & 1 & 0 & 0 & 0 & -u_n X_n & -u_n Y_n & -u_n Z_n & -u_n \\ 0 & 0 & 0 & 0 & X_n & Y_n & Z_n & 1 & -v_n X_n & -v_n Y_n & -v_n Z_n & -v_n \end{pmatrix} \cdot \begin{pmatrix} q_{1,1} \\ q_{1,2} \\ q_{1,3} \\ q_{1,4} \\ q_{2,1} \\ q_{2,2} \\ q_{2,3} \\ q_{2,4} \\ q_{3,1} \\ q_{3,2} \\ q_{3,3} \\ q_{3,4} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (4.6)$$

O problema 4.6 é, costumeiramente, impreciso devido a erros na coleta dos valores dos pontos, neste trabalho, isto acontece devido a imprecisão do *mouse* ao demarcar *pixels* (vide capítulo 5). Assim, usa-se um aproximação pelo método dos mínimos quadrados.

Pelo método dos mínimos quadrados, busca-se o melhor valor para a matriz  $C$  solucionar o sistema 4.6 dada uma restrição que impeça de se chegar à solução trivial ( $C = 0$ ). Em (CARVALHO; SZENBERG; GATTASS, 1998) buscou-se uma restrição linear. Segundo Langer (2010), todavia, por trabalhar em coordenadas homogêneas,  $C$  é independente de escala ( $\lambda C \cong C$ ), podendo-se adotar a restrição  $|C| = 1$  (LANGER, 2010b).

Por conseguinte, este é um problema de minimização solucionável por multiplicadores de Lagrange, onde se chega à expressão 4.7. Langer (2010) observa que a

solução  $Q$  é um autovetor de  $A^T A$  e, para a maior minimização,  $Q$  deve ser o autovetor gerado pelo menor autovalor de  $A^T A$ . (LANGER, 2010a)

$$A^T A.Q = \lambda.Q \quad (4.7)$$

Por fim, a organização dos valores de  $Q$  na forma de  $C$  completa o processo de calibração da câmera; o que permite a busca de pontos 3D correspondentes a pontos 2D da imagem (capítulo 5).

## 5 A Técnica de Reconstrução

Em posse do referencial teórico descrito no capítulo 4, a reconstrução de um objeto se deu através da indicação de pontos de interesse na imagem e da implementação de hipóteses de restrição geométrica, baseadas na transformação de câmera encontrada, a fim de eliminar a multiplicidade inerente às correspondência projetiva (seção 4.1).

Inicialmente, é necessário descobrir qual matriz de transformação projetiva é capaz de gerar tal imagem do objeto. Tal conhecimento permite conhecer a posição 2D, no sistema de coordenadas da imagem e sob seu mesmo ponto de vista, que corresponde a qualquer ponto 3D do espaço. Como discutido na seção 4.3, 6 correspondências entre pontos 2D e 3D, conhecidas previamente, são o bastante para fornecer uma aproximação da transformação desejada; no entanto, neste trabalho são usadas 8 correspondências, obrigatoriamente entre os pontos 3D  $P = \{(x, y, z) | 0 \leq x, y, z \leq 1\}$ .

Tal restrição tem o intuito de simplificar a entrada de dados do usuário no aplicativo de reconstrução (seção 6.3) e garantir a satisfação da condição de independência linear para a boa execução do método de calibração (seção 4.3).

Após o processo de calibração já é possível descobrir se um ponto 3D tem sua projeção coincidente com um ponto de interesse na imagem, assim, por Força-Bruta, busca-se quais pontos 3D, no espaço, correspondem, projetivamente, aos pontos de interesse 2D da imagem.

No entanto, resta, ainda, o problema da multiplicidade de pontos; sabendo-se que isto se dá devido a formação da imagem ser feita através da interseção entre a reta de projeção (reta que liga o ponto 3D retratado ao centro de projeção da câmera) e o plano de projeção (figura 4.2).

Analizando o cenário que contém o objeto, nota-se que a reta de projeção contém vários pontos 3D, mas cada um destes tem um valor exclusivo da coordenada  $Z$  (figura 5.1); isto é provado pela geometria analítica, uma vez que a intersecção entre uma reta (a reta de projeção, no caso) e um plano (no caso, um plano  $Z = k$ , com um  $k$  qualquer) é um ponto.

Se, ao indicar o ponto de interesse, fosse fornecido o seu valor da coordenada

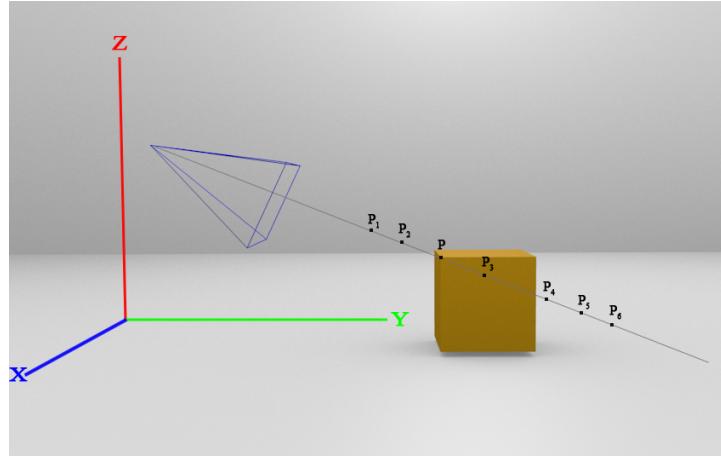


Figura 5.1: Os pontos  $P_i$  pertencem à reta de projeção de  $P$ , mas suas coordenadas  $Z$  são diferentes da coordenada  $Z$  do ponto  $P$

$Z$ , determinar seu correspondente 3D no espaço seria uma tarefa banal com uma demanda exaustiva por dados do usuário. Em viés, sem informações além da transformação projetiva e coordenadas 2D do ponto de interesse não há formas de determinar sua coordenada  $Z$  no espaço (CARVALHO; VELHO, 2005).

No entanto, devido ao apelo visual da imagem, é fácil determinar se um ponto da imagem tem seu correspondente 3D com coordenada  $Z$  igual, ou aceitavelmente igual, à coordenada  $Z$  do correspondente 3D de outros pontos da imagem (figura 5.2). De forma que, ao fornecer os pontos de interesse para a reconstrução, indicar quais pontos de interesse da imagem têm seu correspondente 3D em uma mesma coordenada  $Z$  é simples e provê subterfúgios para a eliminação da multiplicidade projetiva.

Neste trabalho, para cada grupamento é atribuído um identificador numérico inteiro, onde, quanto menor o identificador, menor o valor da coordenada  $Z$  do grupamento.

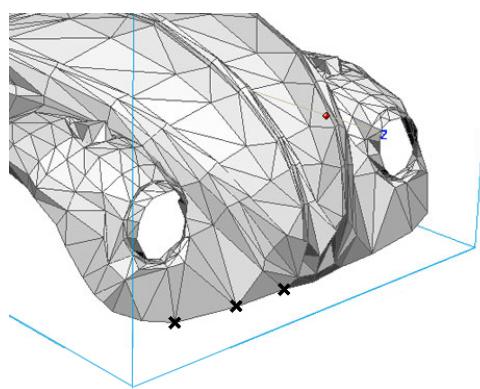


Figura 5.2: É perceptível que os pontos demarcados com um "X" têm a mesma coordenada  $Z$  no espaço

Assim, de posse de um grupamento de pontos da imagem com mesma coordenada  $Z$ , iterativamente, são descartados valores para a coordenada  $Z$  do grupamento que, pela transformação projetiva encontrada para a imagem, não possui um ponto possível de ser o correspondente 3D de algum dos pontos de interesse do grupamento em questão.

Tal eliminação de multiplicidade por interseção é eficaz dado o seu embasamento geométrico citado anteriormente. No entanto, assim como no estabelecimento das correspondências (seção 4.3), há imprecisão na demarcação dos pontos de interesse na imagem. Então, é compreensível que seja adotada uma tolerância projetiva. A tolerância projetiva diz respeito a quanto longe do ponto demarcado como interessante à reconstrução um ponto 3D ainda pode projetar para ser considerado como correspondente a este; tolerâncias grandes levam à imprecisão de reconstrução, ao passo que tolerâncias pequenas levam à dificuldade em achar correspondentes. O valor da tolerância projetiva é definido pelo usuário no momento da reconstrução.

Tal tolerância pode causar a não convergência do algoritmo de eliminação de coordenadas  $Z$  por interseção para um único valor; neste caso, o valor da coordenada  $Z$  adotado é dado por um critério de convergência arbitrário. Neste trabalho foram usados os critérios:

- adotar o valor mais próximo ao valor anteriormente definido (começando do zero);  
ou
- adotar o valor mais próximo à média dos valores possíveis dada a eliminação por intersecção.

Ainda em consequência da tolerância projetiva, um número finito de pontos com a mesma coordenada  $Z$  projetam próximo o suficiente do ponto de interesse. Assim, como correspondente 3D definitivo do ponto de interesse em questão é adotado o baricentro destes pontos adicionais.

Não obstante, é necessário que se determine um espaço finito de busca destes correspondentes 3D, já que há pontos 3D que projetam em um ponto de interesse em infinitas posições do espaço. Para limitar a busca, então, são usados os mesmos pontos 3D aplicados na correspondência para calibração, isto é, os pontos  $P = \{(x, y, z) | 0 \leq x, y, z \leq 1\}$ . Como resultado, os pontos 3D reconstruídos terão suas coordenadas entre 0 e 1.

Dentro do espaço de busca um quesito ainda importante é a distância entre dois

valores consecutivos de uma coordenada (i.e. incremento) durante a iteração de busca. Incrementos menores levam à uma maior precisão da reconstrução, enquanto incrementos maiores tornam o algoritmo mais ágil. O valor do incremento é definido no início da reconstrução.

Em suma, é importante ressaltar que esta técnica reconstrói apenas pontos 3D, arestas e planos podem ser adicionados à nuvem de pontos encontrada em um processo de pós-tratamento.

## 6 Aplicativos Desenvolvidos

Ao decorrer deste trabalho, fora o aplicativo que implementa a técnica de reconstrução descrita nesta monografia, foi sendo necessário que técnicas adjuntas à estudada aqui também fossem implementadas em programas a fim de facilitar comprovações de resultados e automatizações de tarefas auxiliares ao projeto; além de agregar determinada riqueza de esforço e de ferramentas a este trabalho de conclusão de curso.

A seguir são apresentados os exemplos mais representativos dos aplicativos desenvolvidos ao longo do projeto.

### 6.1 Aplicativo de detecção de retas

Tanto para auxiliar na detecção automática dos segmentos de reta do sistema de coordenadas de referência quanto dos pontos (interseção de segmentos de reta) de interesse à reconstrução. Foi desenvolvido um aplicativo para a obtenção de parâmetros analíticos de retas apresentadas em uma região de uma imagem digital.

As linhas inferidas como interessantes, por serem representadas através da rasterização - processo de conversão da representação vetorial para a matricial (AZEVEDO; CONCI, 2003) - precisaram sofrer um processo de regressão linear a fim de se obter uma aproximação dos parâmetros geométricos analíticos de sua formação. A regressão linear executada para a obtenção dos coeficientes angular e linear das retas de interesse se deu através do método dos mínimos quadrados.

O MÉTODOS DOS MÍNIMOS QUADRADOS é uma técnica de regressão linear que busca ajustar corretamente coeficientes de uma curva algébrica de forma que estes minimizem o somatório do quadrado da diferença entre os valores dos dados reais e os valores estimados, minimizando o erro de aproximação.

Com uma interface apropriada, pôde-se carregar uma imagem e, através da demarcação manual de uma região de interesse na mesma (demarcada por um retângulo vermelho), pontos da possível linha foram detectados pela sua cor preta e submetidos ao método dos mínimos quadrados (figura 6.1).

Após a aproximação são exibidos, na parte superior da janela, os coeficientes encontrados e é plotada, sobre a imagem, a reta encontrada.

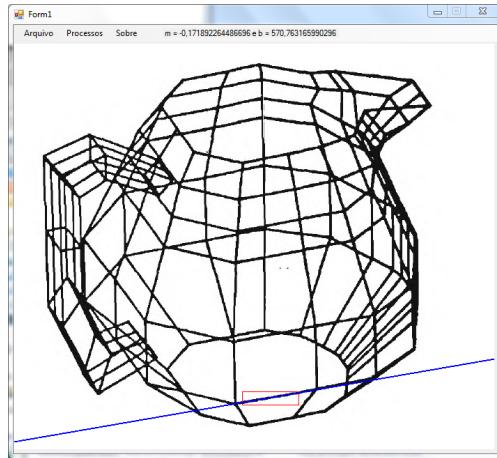


Figura 6.1: O programa de regressão de rasterização com uma aproximação de exemplo

Com a plotagem da reta encontrada pelo programa sobre a região de interesse na imagem, tomou-se como comprovada a veracidade dos dados encontrados pelo aplicativo.

Com esta técnica se obteve uma possibilidade de automatização da busca de elementos de interesse nas imagens estudadas.

Este aplicativo está disponível em ([https://github.com/ivancezanne/TCC/tree/master/App\\_MinimosQuadrados](https://github.com/ivancezanne/TCC/tree/master/App_MinimosQuadrados)).

## 6.2 Aplicativo de Projeção 3D

Tendo em vista que o resultado deste trabalho é gerar malhas 3D, um aplicativo (figura 6.2), baseado na descrição presente em (HUGHES et al., 2013) e capaz de interpretar arquivos PLY (BOURKE, 2014), foi desenvolvido. Mesmo apresentando uma limitação quanto a definições de usuário em relação a cores e normais de faces, a utilidade deste aplicativo no contexto deste trabalho foi notável.

Além da projeção da malha 3D descrita em um arquivo PLY de entrada, este aplicativo possibilita manipulações geométricas na malha (translação e rotação, apenas) e algoritmos prontos de normalização (alteração proporcional das dimensões para o máximo de 1 unidade de medida) e centralização (translação do baricentro da malha à origem do espaço 3D).

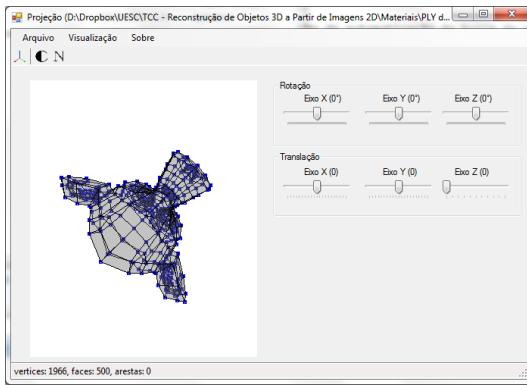
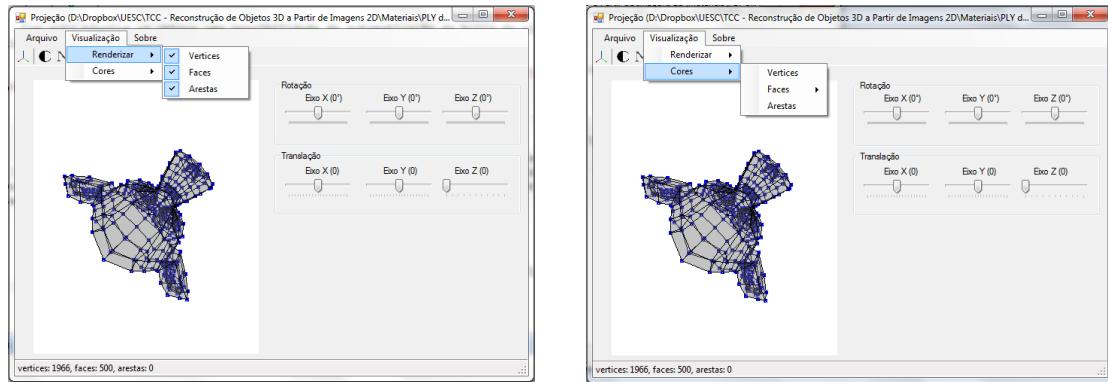


Figura 6.2: Aplicativo de projeção 3D com a famosa malha Suzanne

Como auxílios visuais o aplicativo também permite escolher quais elementos da malha deverão ser projetados à visualização (figura 6.3a) e alterar as cores dos elementos que compõem a malha (figura 6.3b).



(a) Menu onde se escolhe quais elementos da malha serão renderizados

(b) Menu onde se pode alterar as cores dos elementos da malha

Figura 6.3: Opções adicionais do aplicativo de projeção 3D de auxílio à visualização

Em testes com nuvens de pontos pós-tratadas (após a inserção de faces), notou-se uma limitação do componente gráfico da linguagem C# usado na renderização da projeção - o *picturebox*. Tal limitação impede que faces parcialmente fora da área de visualização sejam renderizadas corretamente, causando uma distorção nos polígonos planares correspondentes às faces parcialmente visíveis. Todavia, uma vez que vértices não apresentavam tal problema, as funcionalidades implementadas neste aplicativo o fizeram preferível, na visualização de resultados brutos deste trabalho, em relação ao *software* MeshLab.

Este aplicativo pode ser encontrado em [https://github.com/ivancezanne/TCC/tree/master/App\\_Projecao](https://github.com/ivancezanne/TCC/tree/master/App_Projecao).

## 6.3 Aplicativo de Reconstrução

A técnica de reconstrução deste trabalho (capítulo 5) foi implementada através de um *aplicativo* (figura 6.4) que, com forte auxílio visual, busca otimizar a reconstrução almejada.

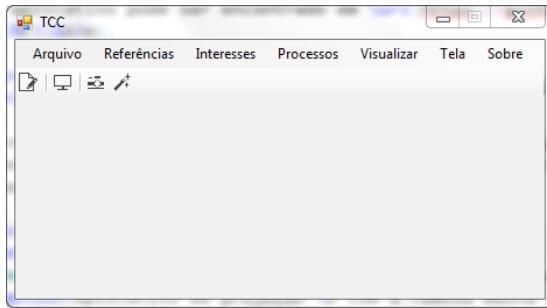


Figura 6.4: Tela inicial do aplicativo de reconstrução

Inicialmente, o aplicativo necessita que uma imagem seja aberta em sua interface para que as demais funcionalidades sejam disponibilizadas. Com uma imagem carregada (que já é aberta com um conjunto de referências arbitrário), a demarcação de pontos de referência e de interesse (capítulo 5) pode ser feita através de *clicks* na região desejada da imagem (figura 6.5).

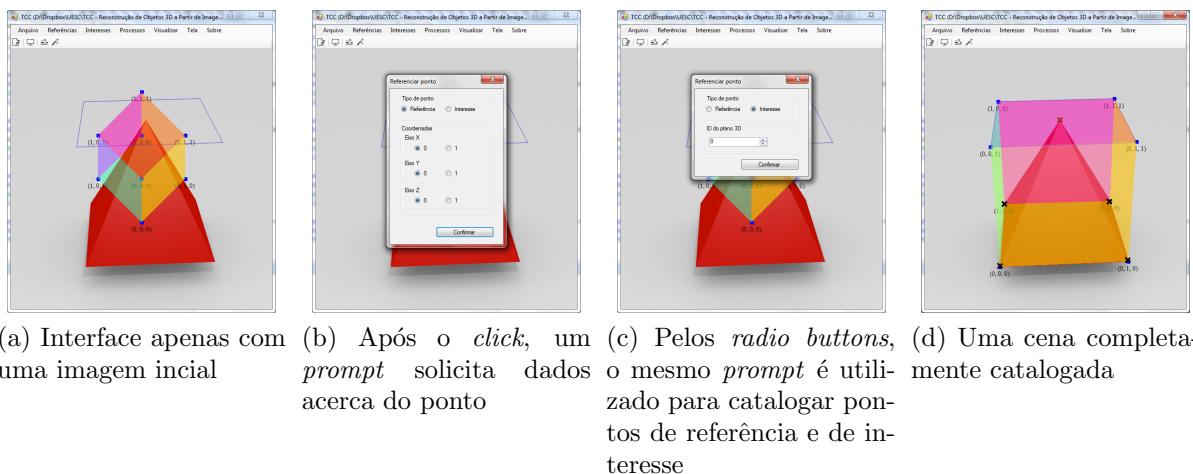


Figura 6.5: Utilização da interface para a demarcação de pontos de referência e interesse

Após o completo fornecimento de dados para a reconstrução, é necessário que se execute a calibração de câmera e só então a reconstrução será possível de ser executada. Para a implementação da calibração (seção 4.3) foi utilizada o método de cálculo de autovalores e autovetores, baseado no algoritmo QR, da biblioteca ALGLIB Free Edition v3.8.2 para linguagem C#.

Ao início da reconstrução, os dados descritos no capítulo 5 (tolerância projetiva, incremento e critério de convergência) são requisitados pelo aplicativo e seu fornecimento é obrigatório ao usuário (figura 6.6).

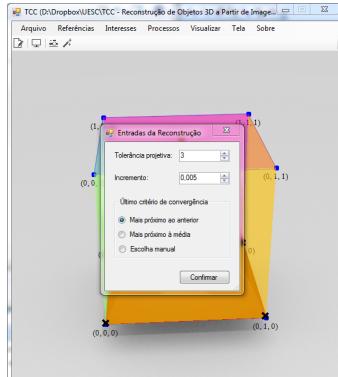


Figura 6.6: *Prompt* com dados necessários à reconstrução

Em adição aos critérios arbitrários de convergência descritos no capítulo 5, o aplicativo oferece a possibilidade de escolha manual, pelo usuário, do valor da coordenada Z dentre os que o algoritmo já encontrou como possíveis (figura 6.7)

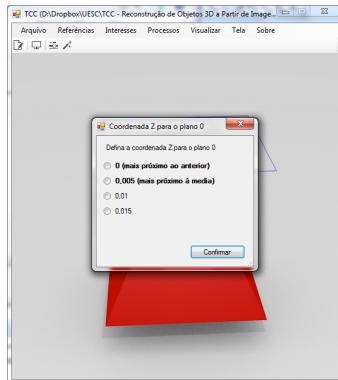


Figura 6.7: *Prompt* para escolha manual da coordenada Z

Em posse disto a técnica de reconstrução proposta neste trabalho pode ser executada por qualquer usuário interessado através de um interface adequada. Em adição, ferramentas adicionais como visualização de aproximação após calibração de câmera; salvamento e carregamento de pontos de referência e de interesse através de arquivos; e captura de tela de trabalho foram implementadas tornando o uso do aplicativo mais completo e eficiente.

Este aplicativo de reconstrução está disponível em [https://github.com/ivancezanne/TCC/tree/master/App\\_Reconstrucao](https://github.com/ivancezanne/TCC/tree/master/App_Reconstrucao).

# 7 Resultados, Discussões e Trabalhos Futuros

Em posse da técnica descrita no capítulo 5 e dos *aplicativos* expostos no capítulo 6, foi possível obter resultados com certo grau de corretude e com grande potencial de melhora em sua exatidão.

A seguir são apresentados testes, resultados e discussões acerca das atividades desenvolvidas neste trabalho.

## 7.1 Testes validadores

Como descrito no capítulo 3, a corretude da técnica desenvolvida, e consequente validade dos resultados obtidos, foi aferida através de cenas específicas (figura 3.4) cujos dados foram catalogados previamente.

Após alguns testes preliminares, com valores para a tolerância projetiva igual a 3 e incremento igual a 0.002, todas as cenas de validação obtiveram seus mais adequados resultados; dependo, individualmente, apenas do critério arbitrário de convergência.

Para a cena de validação 1 (figura 3.4a), cuja montagem de dados está expressa na figura 7.1a, o melhor critério de convergência foi o de valor de coordenada  $Z$  mais próximo ao anteriormente analisado. Demorando, em média, seis minutos e dez segundos para concluir, a reconstrução da cena de validação 1 retornou os dados da tabela 7.1.

Na cena de validação 2 (figura 3.4b), cuja montagem de dados é a da figura 7.1b, o critério de convergência mais eficaz, por sua vez, foi o de valor de coordenada  $Z$  mais próximo à média dos valores possíveis. Com uma demora média para conclusão de quatro minutos, a reconstrução da cena 2 retornou os dados da tabela 7.2.

Na cena de validação 3 (figura 3.4c) - montagem da figura 7.1c, novamente, o critério de escolher coordenadas  $Z$  cujo valor é mais próximo ao plano analisado anteriormente foi o melhor. Em média, eram gastos seis minutos e vinte segundos para a reconstrução da cena de validação 3 retornar os dados da tabela 7.3.

Tabela 7.1: Dados da reconstrução da cena de validação 1

Obtido			Desejado			Erro geométrico ( $E$ )
$X$	$Y$	$Z$	$X$	$Y$	$Z$	
0.006250000	0.007035714	0.000000000	0	0	0	0.009410833
0.996142857	0.003428571	0.000000000	1	0	0	0.005160683
0.992000000	0.990393443	0.000000000	1	1	0	0.012501438
0.005571429	0.994571429	0.000000000	0	1	0	0.007778831
0.995111111	0.005407407	0.990000000	1	0	1	0.012375027
0.997000000	0.998000000	0.990000000	1	1	1	0.010630146
0.010571429	0.017492063	0.990000000	0	0	1	0.022753624
0.009538462	0.995461538	0.990000000	0	1	1	0.014545786

Tabela 7.2: Dados da reconstrução da cena de validação 2

Obtido			Desejado			Erro geométrico ( $E$ )
$X$	$Y$	$Z$	$X$	$Y$	$Z$	
0.007000000	0.005000000	0.008000000	0.0	0.0	0.0	0.011747340
0.011866667	0.993520000	0.008000000	0.0	1.0	0.0	0.015710130
0.967567100	0.989264069	0.008000000	1.0	1.0	0.0	0.035087793
0.982966292	0.004876404	0.008000000	1.0	0.0	0.0	0.019440332
0.766722973	0.517574324	0.904000000	0.5	0.5	1.0	0.284017607

Tabela 7.3: Dados da reconstrução da cena de validação 3

Obtido			Desejado			Erro geométrico ( $E$ )
$X$	$Y$	$Z$	$X$	$Y$	$Z$	
0.507521739	0.512478261	0.000000000	0.5	0.5	0.0	0.014569954
0.504075949	0.992075949	0.000000000	0.5	1.0	0.0	0.008910889
0.989418182	0.511109091	0.000000000	1.0	0.5	0.0	0.015342320
0.987270270	0.981310811	0.000000000	1.0	1.0	0.0	0.022612647
0.521545455	0.521545455	0.492000000	0.5	0.5	0.5	0.031502591
0.511000000	0.995750000	0.492000000	0.5	1.0	0.5	0.014250000
0.997200000	0.516400000	0.492000000	1.0	0.5	0.5	0.018460769
0.996181818	0.995454545	0.492000000	1.0	1.0	0.5	0.009961911

Tabela 7.4: Erros geométricos médios e desvios-padrão dos testes validadores

Teste validador	Erro geométrico médio	Desvio padrão do erro geométrico
Cena de validação 1	0.011894546	0.005286092
Cena de validação 2	0.073200641	0.118182296
Cena de validação 3	0.016951385	0.007320478
Geral	0.028417650	0.059034919

Com o baixo erro geométrico médio (da ordem de  $10^{-2}$ ) e desvio padrão, evidenciados na tabela 7.4, o desempenho da reconstrução é, além de consideravelmente correto, constante.

À exceção da reconstrução errônea, a ser comentada posteriormente, do vértice da pirâmide presente na cena de validação 2, cujo desvio do erro geométrico à média geral de erros é cerca de quatro vezes maior que o desvio-padrão geral (o que o configura como um *outlier*), nota-se grande uniformidade na reconstrução.

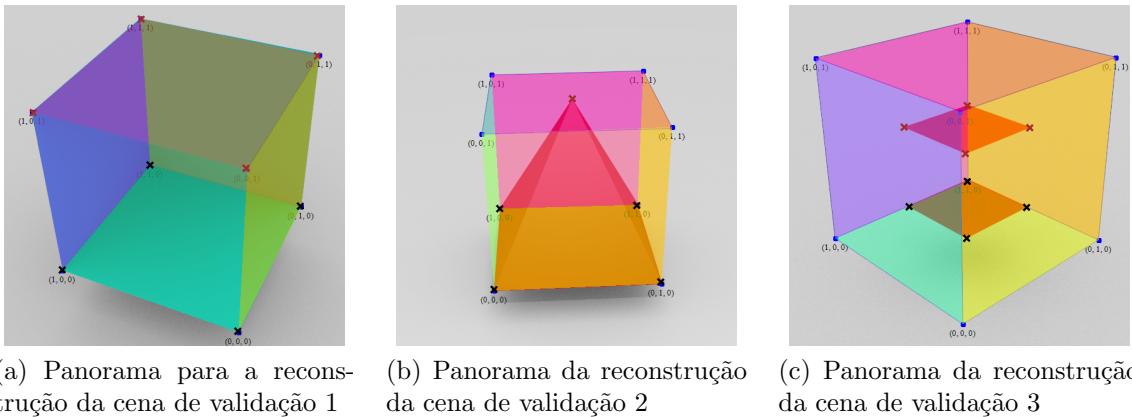


Figura 7.1: Panoramas das reconstruções de validação

Nota-se que, mesmo com certo erro numérico evidenciado pelos erros geométricos ( $E$ ), a geometria dos objetos reconstruídos é mantida (figura 7.2); resguardando, ainda, certa proporção que tem sua exatidão afetada pelas aproximações (aproximação durante calibração e tolerância projetiva) efetuadas pela técnica de reconstrução.

O caso menos bem-sucedido é o da cena 2, onde o vértice da pirâmide, dada a arbitrariedade do critério de escolha do valor da coordenada  $Z$ , é reconstruído muito abaixo da sua posição real causando um deslocamento das coordenadas  $X$  e  $Y$  e, como consequência, a deformação da pirâmide reconstruída.

Através da escolha manual, disponibilizada pelo aplicativo de reconstrução, de valores para as coordenadas  $Z$ , há um erro geométrico médio menor entre os pontos,

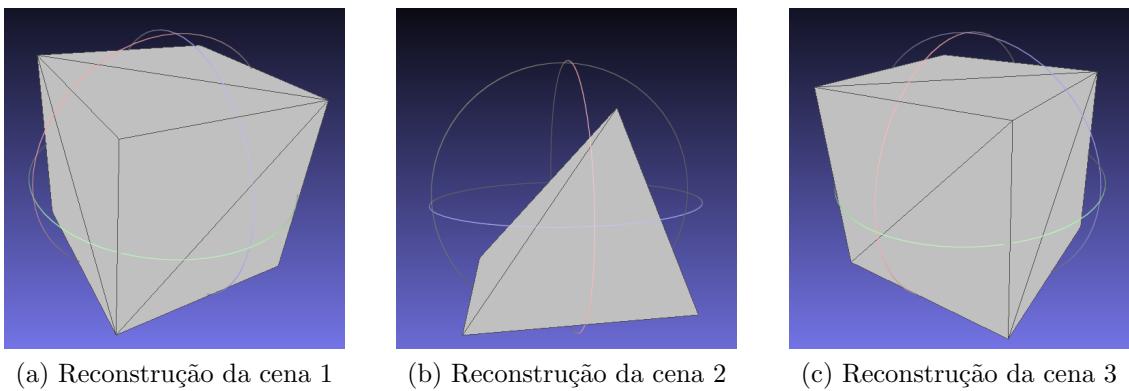


Figura 7.2: Reconstruções validadoras (faces inseridas manualmente)

fazendo com que a geometria dos objetos reconstruídos, principalmente da pirâmide da cena 2, seja mais verossímil. Todavia, a escolha manual é uma estratégia notavelmente brusca e, por isso, resolveu-se não discuti-la nesta monografia.

Todos os dados e resultados dos testes validadores, inclusive os correspondentes à escolha manual, estão disponíveis em [⟨https://github.com/ivancezanne/TCC/tree/master/App\\_Reconstrucao/Testes⟩](https://github.com/ivancezanne/TCC/tree/master/App_Reconstrucao/Testes).

## 7.2 Resultados

Após os testes validadores, nos quais a técnica de reconstrução se mostrou factível, buscou-se a reconstrução das imagens descritas na seção 3.2. Nem todas as imagens foram possíveis de serem reconstruídas, no entanto, pôde-se obter certo resultado com as imagens do *VW Beetle* de Sutherland e da litografia de Dürer.

Utilizando a preparação exposta na figura 7.3, uma tolerância projetiva igual a 2, um incremento de 0.002 e escolha de coordenada  $Z$  mais próxima ao último valor analisado, a reconstrução dos pontos de interesse demarcados para a imagem do *VW Beetle* de Sutherland encontram-se na figura 7.4.

Pela figura 7.4 nota-se que há certas deformações, uma vez que os segmentos de faces deveriam ser retilíneos quando vistos frontalmente e há um deslocamento no sentido do eixo  $Y$ . Todavia, nota-se o distanciamento central entre os segmentos de faces, o que é correto, além de esperado comportamento curvilíneo em relação ao eixo  $Z$ .

De tal forma, conclui-se que a deformação é causada por uma leve imprecisão na demarcação dos pontos de um mesmo plano e um erro que ainda persiste na deter-

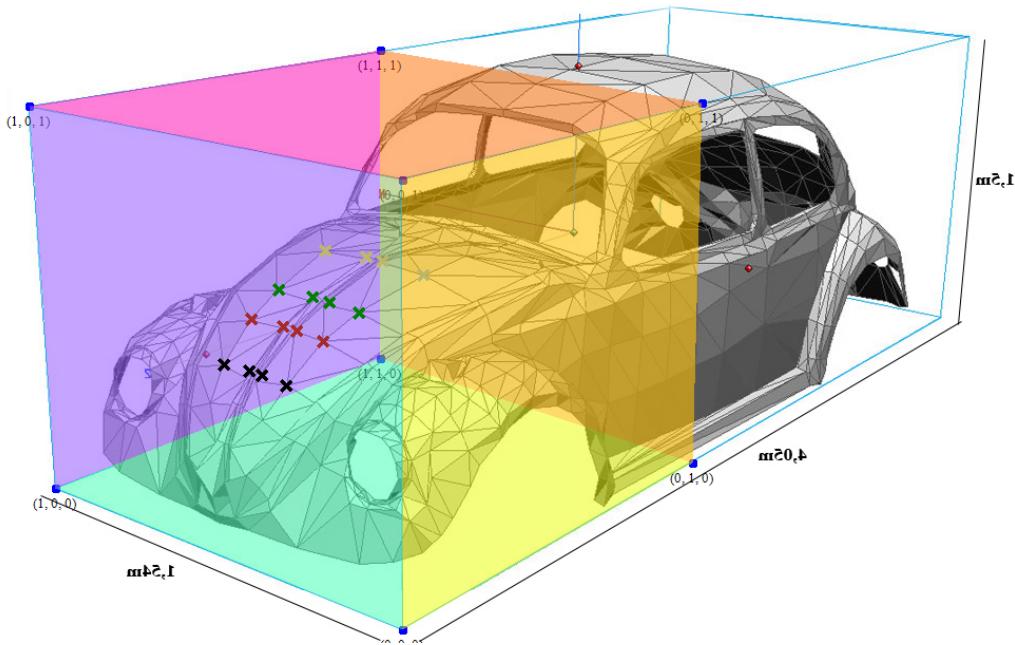
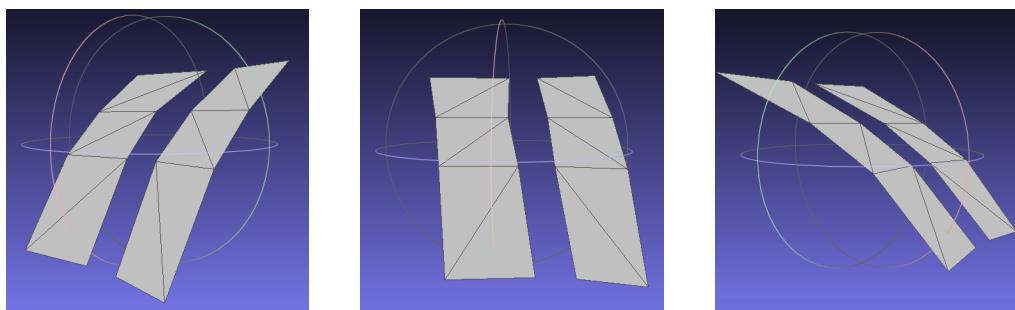


Figura 7.3: Panorama da reconstrução do *VW Beetle* de Sutherland



(a) Visão da lateral esquerda do *VW Beetle* de Sutherland reconstruído  
 (b) Visão frontal da reconstrução do *VW Beetle* de Sutherland  
 (c) Visão da lateral direita do *VW Beetle* de Sutherland reconstruído

Figura 7.4: A reconstrução do *VW Beetle* de Sutherland (Faces inseridas manualmente)

minação das coordenadas  $Z$ .

O deslocamento no eixo  $Y$  é causado por haver demarcação de pontos de um mesmo plano em alturas distintas na imagem; no ponto de vista da imagem, pontos de um mesmo plano no espaço têm sua coordenada  $Y$  influenciada majoritariamente pela posição vertical do ponto de interesse na imagem. Já o erro de não paralelismo, por sua vez, ocorre devido a arbitrariedade dos critérios finais de convergência, que seguem uma tendência de crescimento para a coordenada  $Z$  diferente da tendência de crescimento real, fazendo com que pontos de um plano, por estarem em uma altura incorreta, se desloquem nos sentidos dos eixos  $X$  e  $Y$ .

Para a litografia de Dürer, a melhor configuração foi com tolerância projetiva

Tabela 7.5: Dados da reconstrução do *VW Beetle* de Sutherland

<i>X</i>	<i>Y</i>	<i>Z</i>
0.793081081081082	0.471351351351352	0.1680000000000000
0.886315789473685	0.483947368421053	0.1680000000000000
0.929076923076924	0.483692307692308	0.1680000000000000
0.998000000000001	0.4710000000000000	0.1680000000000000
0.764600000000001	0.5626000000000000	0.2740000000000000
0.864844444444445	0.576177777777778	0.2740000000000000
0.91142222222223	0.576177777777778	0.2740000000000000
0.998000000000001	0.5600000000000000	0.2740000000000000
0.737574468085107	0.654255319148936	0.3360000000000000
0.846000000000001	0.6660000000000000	0.3360000000000000
0.905625000000001	0.668083333333334	0.3360000000000000
0.998000000000001	0.6500000000000000	0.3360000000000000
0.683440000000000	0.8208400000000000	0.4140000000000000
0.837473684210527	0.840210526315791	0.4140000000000000
0.891864406779662	0.840813559322035	0.4140000000000000
0.997500000000001	0.8060000000000001	0.4140000000000000

igual a 3, incremento de 0.002 e escolha da coordenada *Z* mais próxima à média dos valores possíveis. Assim, com os dados para reconstrução expostos na figura 7.5, obtiveram-se os resultados da figura 7.6

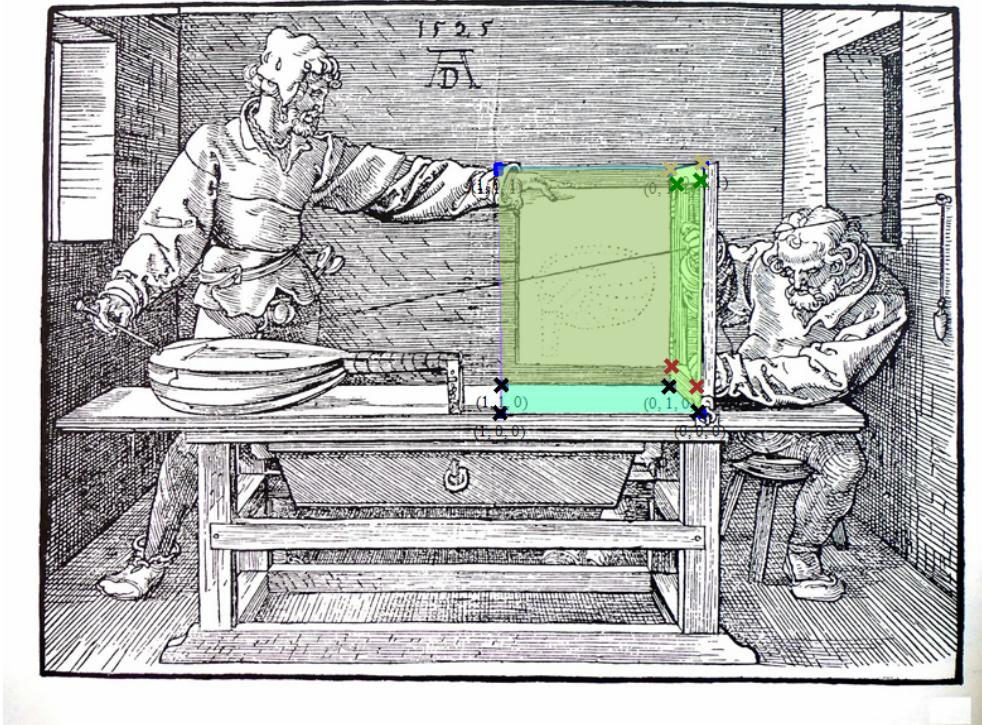


Figura 7.5: Panorama da reconstrução da litografia do Dürer

No caso da litografia do Dürer, o ponto de vista da imagem corrobora para

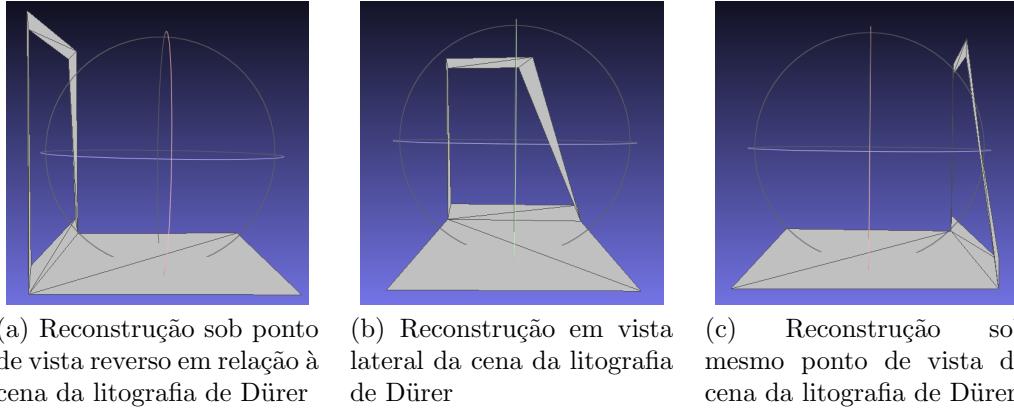


Figura 7.6: A reconstrução da cena da litografia de Dürer (Faces inseridas manualmente)

Tabela 7.6: Dados da reconstrução da Litografia de Dürer

<i>X</i>	<i>Y</i>	<i>Z</i>
0.00830218687872761	0.04745526838966200	0.006000000000000000
0.9960000000000000200	0.028000000000000000	0.006000000000000000
0.9900000000000000300	0.8860000000000000100	0.006000000000000000
0.02913911007025730	0.86632693208430800	0.006000000000000000
0.01040092699884120	0.06723754345307070	0.102000000000000000
0.02020642648490740	0.84466309639727500	0.102000000000000000
0.05101063273487950	0.47641345651037100	0.9360000000000000100
0.00647956403269755	0.04270299727520430	0.9360000000000000100
0.00461463414634147	0.03157073170731700	0.9960000000000000100
0.07145343800758070	0.55953627504060700	0.9960000000000000100

uma melhor convergência de valores para coordenada *Z*. Todavia, a demarcação das referências e pontos de interesse é muito difícil dada a pequena região e ausência de dados claros sobre dimensões. Assim, o resultado da reconstrução sofre distorções causadas por má catalogação de dados iniciais.

A imagem do Bule de Newell foi dispensada pois, dada a necessidade de demarcação de pontos de referência, esta não possibilitava um bom referenciamento (vide figura 7.7) devido à ausência de pontos interessantes nessa região e o fato de sua projeção ser paralela, o que causaria a má calibração de câmera (que atende ao modelo em perspectiva).

Os dados e resultados destas reconstruções estão disponíveis em [https://github.com/ivancezanne/TCC/tree/master/App\\_Reconstrucao/Resultados](https://github.com/ivancezanne/TCC/tree/master/App_Reconstrucao/Resultados).

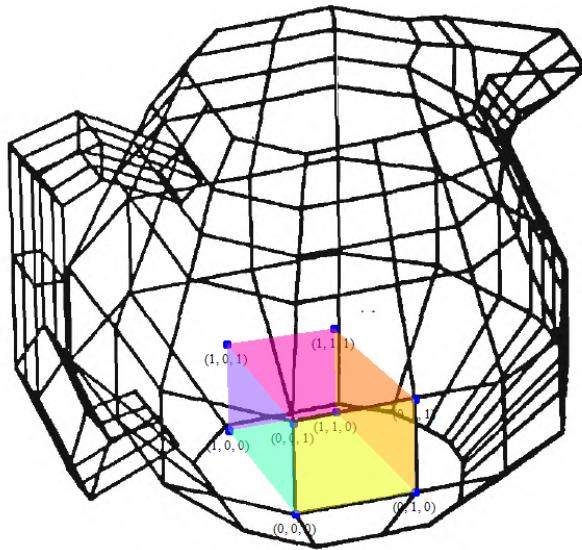


Figura 7.7: Referenciamento na imagem do Bule de Newell

## 7.3 Discussões

O resultado gerado pelo *aplicativo* de reconstrução - disponibilizado em arquivos de extensão PLY - pode, portanto, ser lido pelos principais programas de modelagem 3D; além disso, os *aplicativos* anexos ao de reconstrução desempenham um bom auxílio a este último, o que faz deste trabalho um pacote de ferramentas computacionais direcionadas à reconstrução 3D.

Por outro lado, metodologia e valores de referenciamento e região de busca fixados, além de um único modelo de câmera configuram-se como limitações à técnica de reconstrução que, ao fim, causam certa imprecisão ou restrição nos resultados objetivados. A delimitação de uma área de busca fixa, por exemplo, impede que um conjunto maior de reconstruções seja executado; a metodologia de demarcação - por *clicks* - é costumeiramente imprecisa; e a estratégia de eliminação da multiplicidade projetiva não forá tão eficaz quanto o necessário.

De qualquer forma, a exemplo da litografia do Dürer e da imagem do *VW Beetle* de Sutherland, viu-se que as imagens não propiciaram uma boa determinação de pontos de referência e interesse, uma vez que medidas eram inexistentes ou não correspondiam às medidas que o software requisitava; e, a exemplo da imagem do Bule de Newell, a projeção paralela impede o bom funcionamento da calibração de câmera necessária à reconstrução.

Como visto na subseção 4.1.1, a projeção paralela mantém aspectos isométricos

ao manter as coordenadas  $X$  e  $Y$ , independente da coordenada  $Z$ . Desta forma, sugere-se que um modelo de câmera que ignore a profundidade (equação 7.1) possa permitir que a técnica deste trabalho aja corretamente em imagens isométricas. Todavia, esta é uma hipótese ainda a ser verificada.

$$C = \begin{pmatrix} q_{1,1} & q_{1,2} & q_{1,3} & q_{1,4} \\ q_{2,1} & q_{2,2} & q_{2,3} & q_{2,4} \end{pmatrix} \quad (7.1)$$

Embora tenha-se notado que as imagens utilizadas para a reconstrução neste trabalho (*VW Beetle* de Sutherland, Bule de Newell e Litografia do Dürer) não permitem uma reconstrução perfeitamente verossímil dos seus objetos, pode-se concluir que o aprimoramento de uma técnica baseada na descrita na seção 6.1 é uma promissora abordagem para uma reconstrução mais fiel dos objetos retratados. Neste caso, restrições algébricas inerentes ao elemento geométrico detectado podem se aliar às restrições projetivas e resolver, com maior fidelidade, o problema da multiplicidade projetiva. Neste ínterim, condições de simetria e repetições seriam, também, melhor aplicáveis.

Todavia, é importante notar a importância deste trabalho na busca por uma expansão no rol de técnicas de reconstrução 3D; já que, como discutido no capítulo 2, as reconstruções mais eficazes atualmente só são possíveis através de várias imagens do objeto de interesse.

Por fim, nota-se que a técnica de reconstrução desenvolvida neste trabalho gera resultados coesos, cuja eficiência é constante e mensurável, e satisfaz uma necessidade inicial de obtenção de dados tridimensionais.

## 7.4 Trabalhos Futuros

No futuro, a conclusão da avaliação do modelo de câmera isométrico e consequente integração ao software de reconstrução aqui exposto é uma possibilidade. Por outra via, a já descrita integração da técnica de reconstrução com a de detecção de regiões descritas parametricamente prevê uma grande evolução desta discussão.

Por fim, deseja-se focar o desenvolvimento desta técnica em um escopo mais estrito - modelos arquitetônicos (de forma similar a (LEE; FENG; GOOCH, 2008)) - pois acredita-se que com um escopo mais restrito, o conhecimento prévio do objeto de interesse

será mais rico e possibilitará um estudo de restauro e documentação patrimonial robusto.

## Referências Bibliográficas

- AGUIAR, E. de. Preliminary techniques. In: *Animation and Performance Capture Using Digitized Models*. New York: Springer, 2010. p. 9–18.
- AZEVEDO, E.; CONCI, A. *Computação Gráfica: Teoria e Prática*. Rio de Janeiro: Elsevier, 2003.
- BOURKE, P. *PLY - Polygon File Format*. 2014. Disponível em <<http://paulbourke.net/dataformats/ply/>> Acesso em 10 Nov 2014.
- BRUNELLESCHI'S Peepshow & the Origins of Perspective. 1998. Disponível em <<https://www.dartmouth.edu/~matc/math5.geometry/unit11/unit11.html>> Acesso em: 24 Set 2014.
- CARVALHO, P. C.; VELHO, L. *Fotografia 3D*. Rio de Janeiro, 2005. 174 p.
- CARVALHO, P. C. P.; SZENBERG, F.; GATTASS, M. Image-based modeling using a two step camera calibration method. In: SIBGRAPI'98. *Proceedings of International Symposium on Computer Graphics, Image Processing and Vision*. [S.l.], 1998. p. 388–395.
- GOMES, J.; VELHO, L. Câmera virtual. In: *Fundamentos da Computação Gráfica*. Rio de Janeiro: IMPA, 2008. p. 343–371.
- GOMES, J.; VELHO, L. Geometria. In: *Fundamentos da Computação Gráfica*. Rio de Janeiro: IMPA, 2008. p. 21–52.
- HEINZLE, S. et al. Computational stereo camera system with programmable control loop. In: *ACM SIGGRAPH 2011 Papers*. New York, NY, USA: ACM, 2011. (SIGGRAPH '11), p. 94:1–94:10.
- HUGHES, J. F. et al. *An Ancient Renderer Made Modern*. 3. ed. [S.l.]: Addison-Wesley Professional, 2013. 61-80 p.
- KUBERT, B. R.; SZABO, J.; GIULIERI, S. The perspective representation of functions of two variables. *J. ACM*, ACM, New York, NY, USA, v. 15, n. 2, p. 193–204, abr. 1968.
- LANGER, M. *Lecture 18: Least Squares, Singular Value Decomposition*. 2010. Disponível em: <<http://www.cim.mcgill.ca/~langer/558/18-leastsquares.pdf>> Acesso em 18 Set 2014.
- LANGER, M. *Lecture 19: Camera Calibration*. 2010. Disponível em: <<http://www.cim.mcgill.ca/~langer/558/19-cameracalibration.pdf>> Acesso em 18 Set 2014.
- LEE, S.; FENG, D.; GOOCH, B. Automatic construction of 3d models from architectural line drawings. In: *Proceedings of the 2008 symposium on Interactive 3D graphics and games*. [S.l.: s.n.], 2008. p. 123–130.
- MUSEUM, C. H. *Mapping Sutherland's Volkswagen*. 2014. Disponível em <<http://www.computerhistory.org/revolution/computer-graphics-music-and-art/15/206/560>> Acesso em: 14 Ago 2014.

- MUSEUM, C. H. *Rendering of Sutherland's Volkswagen*. 2014. Disponível em <<http://www.computerhistory.org/revolution/computer-graphics-music-and-art/15/206/559>> Acesso em: 14 Ago 2014.
- MUSEUM, C. H. *Wireframe model of Sutherland's Volkswagen*. 2014. Disponível em <<http://www.computerhistory.org/revolution/computer-graphics-music-and-art/15/206/558>> Acesso em: 14 Ago 2014.
- NEVES, A. C. das; TSZUKI, M. de S. G. *Reconstrução 3D a Partir de Sequência de Imagens*. Monografia (Graduação) — Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos, Escola Politécnica da Universidade de São Paulo, 2010.
- NEWELL, M. E. The utilization of procedure models in digital image synthesis. Supported by Advanced Research Projects Agency of the Department of Defense. 1975.
- POTTMANN, H. et al. Projections. In: *Architectural Geometry*. [S.l.]: Bentley Institute Press, 2007. p. 23–65.
- SZENBERG, F. *Acompanhamento de Cenas Com Calibração Automática de Câmeras*. 144 p. Tese (Doutorado) — Departamento de Informática, Pontifícia Universidade Católica do Rio, 2001.
- TORCHINSKY, J. *The First Real Object Ever 3D Scanned And Rendered Was A VW Beetle*. 2013. Disponível em: <<http://floost.com/jalopnik-post-the-first-real-object-ever-3d-scanned-and-rendered-was-a-vw-beetle-2706861>> Acesso em: 14 ago 2014.