



SALA INTELIGENTE UTILIZANDO GEOMETRIA DE MÚLTIPLAS IMAGENS

Daniel de Carvalho Cayres Pinto

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientador: José Gabriel Rodríguez Carneiro
Gomes

Rio de Janeiro
Dezembro de 2013

SALA INTELIGENTE UTILIZANDO GEOMETRIA DE MÚLTIPLAS
IMAGENS

Daniel de Carvalho Cayres Pinto

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

Prof. José Gabriel Rodríguez Carneiro Gomes, Ph.D.

Prof. Mariane Rembold Petraglia, Ph.D.

Prof. Ricardo Guerra Marroquim, D.Sc.

RIO DE JANEIRO, RJ – BRASIL
DEZEMBRO DE 2013

Pinto, Daniel de Carvalho Cayres

Sala Inteligente Utilizando Geometria de Múltiplas Imagens/Daniel de Carvalho Cayres Pinto. – Rio de Janeiro: UFRJ/COPPE, 2013.

XI, 52 p.: il.; 29, 7cm.

Orientador: José Gabriel Rodríguez Carneiro Gomes

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2013.

Referências Bibliográficas: p. 45.

1. Sala inteligente. 2. Geometria projetiva. 3. Tensor trifocal. I. Gomes, José Gabriel Rodríguez Carneiro. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

À minha família.

Agradecimentos

Agradeço a todos os professores do Programa de Engenharia Elétrica da COPPE e ao meu orientador acadêmico, professor José Gabriel Rodríguez Carneiro Gomes pelos ensinamentos.

Agradeço à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo suporte financeiro.

Agradeço a toda minha família, à Ana Cecilia e aos meus amigos Diego Santos Wanderley e Carlos Vinícius Caldas Campos pelo apoio.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

SALA INTELIGENTE UTILIZANDO GEOMETRIA DE MÚLTIPLAS IMAGENS

Daniel de Carvalho Cayres Pinto

Dezembro/2013

Orientador: José Gabriel Rodríguez Carneiro Gomes

Programa: Engenharia Elétrica

Este trabalho apresenta uma sala inteligente simplificada na qual o sistema de visão computacional da sala formado por três câmeras detecta quando um usuário aponta para um objeto entre um conjunto de objetos presentes no ambiente. A determinação de qual objeto foi selecionado se dá a partir de uma reconstrução tridimensional da cena realizada através de técnicas de geometria de múltiplas imagens.

A reconstrução 3D dos objetos da sala com duas câmeras foi realizada a partir da obtenção da matriz fundamental, que, em Geometria Projetiva, é a matriz que engloba as principais relações entre duas imagens de uma mesma cena capturadas por diferentes câmeras, ou por uma mesma câmera em diferentes ângulos. Para a reconstrução com três câmeras foi utilizado o tensor trifocal, que possui papel semelhante ao da matriz fundamental, mas relacionando três imagens da cena.

Do tensor trifocal e da matriz fundamental foram extraídas as matrizes das câmeras, que representam a operação de projeção realizada por suas respectivas câmeras. A partir dessas matrizes foram realizadas as reconstruções tridimensionais dos objetos da sala e do instrumento utilizado para apontar para eles, possibilitando assim a verificação do objeto apontado pelo usuário da sala.

Os resultados obtidos usando tanto duas quanto três câmeras foram satisfatórios, não havendo erro na detecção dos objetos durante os testes. Não houve diferenças significativas entre as reconstruções realizadas a partir de duas e três imagens.

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

SMART ROOM USING MULTIPLE-IMAGE GEOMETRY

Daniel de Carvalho Cayres Pinto

December/2013

Advisor: José Gabriel Rodríguez Carneiro Gomes

Department: Electrical Engineering

This work presents a simplified smart room in which a computer vision system based on three cameras detects which object a user points to, among a set of objects in the environment. The determination of which object was selected is carried out from the tridimensional reconstruction of the scene, which is accomplished by means of multiple-view geometry techniques.

To reconstruct the room objects from two cameras, the fundamental matrix must be obtained. In Projective Geometry, this is the matrix which represents the main relationships between two images from the same scene, which are captured by different cameras or by a single camera from different angles. To reconstruct room objects from three cameras, the trifocal tensor was used. It plays a similar role to the fundamental matrix one, but relating three images from the scene.

From the trifocal tensor and the fundamental matrix, the camera matrices were extracted. These matrices represent the projection operations performed by their respective cameras. Tridimensional room object and pointing device reconstructions were obtained from these matrices, thus allowing the determination of the object pointed to by the room user.

The results obtained by using either two or three cameras were satisfactory, in the sense that no object detection error occurred during the tests. There was no significant difference between reconstructions performed from two or three cameras.

Sumário

Lista de Figuras	x
Lista de Tabelas	xi
1 Introdução	1
1.1 Motivação	1
1.2 Objetivos	2
1.3 Metodologia	2
1.4 Organização	3
2 Modelagem e Reconstrução 3D	5
2.1 Modelagem da Sala Inteligente	5
2.1.1 Pré-Processamento	6
2.2 Geometria Projetiva	6
2.3 Modelo de Câmera	7
2.4 Reconstrução 3D de um ponto	9
3 Usando Duas Câmeras	10
3.1 Geometria Epipolar	10
3.2 Matriz Fundamental	11
3.2.1 Cálculo da Matriz Fundamental a Partir das Matrizes das Câmeras	11
3.2.2 Cálculo da Matriz Fundamental a Partir da Correspondência de Pontos	12
3.2.3 Normalização	13
3.3 Matriz Essencial	14
3.4 Obtendo as Matrizes das Câmeras	15
3.4.1 Obtendo as Matrizes das Câmeras através da Matriz Funda- mental	15
3.4.2 Obtendo as Matrizes das Câmeras através da Matriz Essencial	16
3.5 Algoritmo	18

4 Usando Três Câmeras	20
4.1 Tensor Trifocal	20
4.2 Calculando o Tensor Trifocal	24
4.2.1 Corrigindo as Características Geométricas do Tensor Trifocal .	25
4.2.2 Normalização	25
4.3 Extraindo as Matrizes Fundamentais do Tensor Trifocal	26
4.4 Obtendo as Matrizes das Câmeras	27
4.5 Algoritmo	28
5 Sistema Completo	31
5.1 Visão Geral do Sistema	31
5.2 Rotinas do Programa	32
5.2.1 Captura de Imagens	33
5.2.2 Calibração	34
5.2.3 Correspondência de Pontos	35
5.2.4 Reconstrução	35
6 Resultados	37
7 Conclusões	43
Referências Bibliográficas	45
A Códigos em MATLAB	46
A.1 Obtendo a Matriz de Calibração	46
A.2 Obtendo as Matrizes das Câmeras Usando a Matriz Fundamental . .	47
A.3 Obtendo as Matrizes das Câmeras Usando o Tensor Trifocal	49

Lista de Figuras

2.1	Configuração da sala inteligente.	5
2.2	Pré-processamento das imagens utilizadas para a reconstrução.	6
2.3	Projeção de um ponto \mathbf{X} no espaço no ponto \mathbf{x} no plano focal de uma câmera.	8
2.4	<i>Grid</i> de calibração.	8
3.1	Geometria epipolar.	11
3.2	Mapeamento de um ponto na imagem na primeira câmera em uma reta epipolar na imagem da segunda.	12
4.1	Relação entre as imagens projetadas de uma reta no espaço.	21
4.2	Relação entre duas imagens \mathbf{x} e \mathbf{x}'' de um ponto e a imagem \mathbf{I}' de uma reta que passa por ele.	23
4.3	Relação entre três imagens de um ponto.	23
5.1	Diagrama do <i>hardware</i> do trabalho.	31
5.2	Interface do aplicativo.	32
5.3	Resultado de uma reconstrução 3D.	32
5.4	Diagrama de entradas e saídas do aplicativo.	33
5.5	Rotinas do programa.	33
5.6	Procedimento de captura de imagens.	34
5.7	Rotina de marcação de pontos de referência para calibração.	34
5.8	Processo de correspondência de pontos entre imagens.	35
5.9	Rotina de reconstrução 3D das cenas.	35
6.1	Configuração das câmeras.	37
6.2	Pontos utilizados para calibração (a) e correspondência de pontos (b).	38

Lista de Tabelas

6.1	Imagens dos conjuntos 1, 2, 3 e 4, utilizados nas reconstruções das cenas.	39
6.2	Imagens dos conjuntos 5, 6, 7 e 8, utilizados nas reconstruções das cenas.	40
6.3	Resultados das reconstruções das cenas a partir dos conjuntos de imagens 1, 2, 3 e 4.	41
6.4	Resultados das reconstruções das cenas a partir dos conjuntos de imagens 5, 6, 7 e 8.	42

Capítulo 1

Introdução

A interação das pessoas com os equipamentos eletrônicos vem sofrendo importantes transformações. Os dispositivos de entrada tradicionais, como teclados e mouses, têm sido substituídos por novas tecnologias que buscam tornar os comandos dos usuários mais simples e intuitivos. A essa nova forma de interação homem-máquina deu-se o nome de interface natural com o usuário.

Diversas áreas da Engenharia Eletrônica estão envolvidas nos sistemas de interface natural. O processamento de áudio é utilizado para o reconhecimento de voz, o processamento de imagens e a visão computacional, para reconhecimento facial e de gestos, por exemplo, e vários tipos de sensores são usados na detecção de movimentos e de toque e na medição de distâncias, entre outras aplicações.

Embora as técnicas de interação natural sejam utilizadas com mais frequência em dispositivos móveis (telefones celulares e *tablets*), computadores pessoais, videogames e outros equipamentos eletrônicos isoladamente, há a possibilidade de expandir sua utilização para ambientes inteiros de forma integrada. Uma sala inteligente é um ambiente no qual um sistema captura informações visuais, de áudio, de toque e quaisquer outras informações relevantes através de sensores e as processa, interpretando os possíveis comandos enviados pelas pessoas presentes na sala e, como resposta, realiza as ações correspondentes aos comandos [1].

1.1 Motivação

A visão computacional desempenha um papel fundamental na sala inteligente. De uma única câmera é possível desenvolver muitas funcionalidades como reconhecimento de pessoas, expressões faciais e gestos. Em geral, é útil utilizar múltiplas câmeras para que as imagens cubram o ambiente por inteiro.

A utilização de múltiplas imagens traz também outra importante vantagem, que é explorada neste trabalho. Ao utilizar-se mais de uma imagem é possível realizar uma reconstrução tridimensional de todos os objetos e pessoas na sala. Em [2] é

visto um exemplo de aplicação de múltiplas câmeras em uma sala inteligente para o reconhecimento de comandos gestuais.

Foram desenvolvidos, no Laboratório de Processamento Analógico e Digital de Sinais (PADS/COPPE/UFRJ), trabalhos anteriores envolvendo a utilização de pares de câmeras para dimensionamento [3], [4] e estimação de movimento [5]. No atual trabalho, procura-se investigar os efeitos da inclusão de uma câmera ao sistema. Portanto, serão investigados os detalhes teóricos da geometria de múltiplas imagens envolvendo três câmeras. O trabalho é valioso, além de sua parte teórica, porque a montagem de um sistema completo (Capítulo 5) permitiu a aquisição de conhecimento empírico sobre sistemas de três câmeras, como por exemplo as considerações que são apresentadas no final do Capítulo 6.

A reconstrução 3D também cria novas possibilidades para a interação por gestos e permite que se crie e movimente modelos dos objetos e pessoas existentes, ou mesmo virtuais, ao que se dá o nome de realidade aumentada.

1.2 Objetivos

O objetivo deste trabalho é construir uma sala inteligente simplificada na qual, dadas imagens de um conjunto de objetos simples (retangulares) e um cilindro fino (apontador) manuseado por um usuário da sala para apontar para um dos objetos, o sistema é capaz de reconhecer para qual objeto esse usuário está apontando.

É também feita uma comparação qualitativa a respeito das diferenças obtidas quando são utilizadas duas ou três câmeras para a aquisição das imagens e reconstrução 3D da cena.

1.3 Metodologia

Foram utilizadas, para a aquisição das imagens, três *webcams* com resolução de 1280 pixels de largura por 800 pixels de altura. Por simplicidade na segmentação e modelagem, foram utilizados como objetos retângulos de cores diferentes. Dessa forma, os objetos puderam ser modelados apenas pelas coordenadas de seus vértices, obtidas a partir de uma segmentação baseada em cor e de detecções de bordas e retas.

A reconstrução tridimensional do ambiente e a modelagem matemática das câmeras foram feitas com base na Geometria Projetiva, ferramenta matemática mais natural para a modelagem de projeções [6], [7].

Para atender ao objetivo principal do sistema, de detectar o objeto apontado, foram utilizadas técnicas que extraem informações a respeito do sistema de câmeras através da comparação das imagens de pontos correspondentes em cada câmera, sem a necessidade da utilização de um referencial global para o sistema de coordenadas

3D. Para duas câmeras, as relações entre suas imagens são representadas por uma matriz denominada matriz fundamental. Para três câmeras, a entidade matemática que relaciona suas imagens é o tensor trifocal. Da matriz fundamental e do tensor trifocal é possível obter as matrizes das câmeras, que representam sua operação de projeção, e a partir delas realizar a reconstrução.

Embora seja possível criar reconstruções 3D das cenas e determinar os objetos apontados sem a utilização de um referencial para a calibração das câmeras, as reconstruções obtidas dessa maneira apresentam distorções projetivas que, embora não interfiram no resultado, inviabilizam uma análise qualitativa da reconstrução. Então, para remover as distorções projetivas, as câmeras foram calibradas com auxílio de um *grid* de calibração com coordenadas 3D conhecidas. A calibração das câmeras fornece uma matriz de parâmetros internos que pode ser utilizada para corrigir as matrizes das câmeras obtidas anteriormente, permitindo uma reconstrução euclidiana que pode ser analisada qualitativamente.

1.4 Organização

O trabalho se inicia com a modelagem da sala inteligente, no Capítulo 2. Na Seção 2.1 o funcionamento do sistema e a descrição da sala inteligente são detalhados. Nela é definido como serão representados os objetos da sala e qual é o pré-processamento realizado para se extrair das imagens informações que podem ser utilizadas nos algoritmos apresentados posteriormente. A ferramenta matemática básica para esses algoritmos é a Geometria Projetiva, descrita de forma sucinta na Seção 2.2. Nessa seção são apresentados as principais definições, propriedades e resultados que serão utilizados ao longo do trabalho. Em seguida, na Seção 2.3, define-se o modelo de câmera a ser utilizado. Nesse modelo, baseado na Geometria Projetiva, a operação de projeção é representada pela matriz da câmera. A Seção 2.4 mostra como é realizada a reconstrução 3D de um ponto a partir de múltiplas imagens capturadas por diferentes câmeras, ou de diferentes ângulos conhecendo as matrizes das câmeras e as coordenadas 2D das imagens do ponto.

O Capítulo 3 traz a teoria necessária para a realização da reconstrução tridimensional de um ponto usando as imagens de uma mesma cena capturadas por duas câmeras. Após uma breve exposição sobre Geometria Epipolar na Seção 3.1, as relações entre essas imagens são expressas pela matriz fundamental, apresentada na Seção 3.2. Nessa seção são apresentadas as principais propriedades dessa matriz, é explicado como ela relaciona pontos e retas nas duas imagens e são descritos os métodos para obtê-la a partir das duas matrizes de câmera, ou através de um procedimento de correspondência de pontos entre as duas imagens. Uma forma especial da matriz fundamental, chamada de matriz essencial, é introduzida na Seção 3.3.

A vantagem da utilização da matriz essencial em vez da fundamental fica clara na Seção 3.4, que mostra como obter as matrizes das duas câmeras a partir da matriz fundamental ou da essencial. Essa seção mostra que as matrizes das câmeras obtidas através da matriz fundamental apresentam uma ambiguidade projetiva, que não existe quando são obtidas utilizando a matriz essencial. A reconstrução a partir de matrizes de câmeras obtidas a menos de uma ambiguidade projetiva apresenta uma distorção, chamada de distorção projetiva, que torna mais difícil a análise qualitativa do resultado da reconstrução, portanto para a avaliação dos resultados são utilizadas as reconstruções obtidas a partir da matriz essencial. Por fim, na Seção 3.5, exibe-se o algoritmo de reconstrução com duas câmeras utilizado neste trabalho.

De forma semelhante ao caso de duas câmeras, o Capítulo 4 mostra que também é possível obter relações entre as imagens de três câmeras. A Seção 4.1 apresenta o tensor trifocal, um arranjo de 27 números na forma $3 \times 3 \times 3$, que engloba as relações existentes entre três imagens de uma mesma cena, de forma semelhante à matriz fundamental para o caso de duas imagens.

Uma forma de obter o tensor trifocal a partir de um procedimento de correspondência de pontos entre três imagens é explicada na Seção 4.2. Na Seção 4.3 é explicado como obter as matrizes fundamentais que relacionam a segunda câmera e a terceira câmera com a primeira e a Seção 4.4 mostra como se pode recuperar as matrizes das câmeras usando essas matrizes fundamentais, permitindo a reconstrução 3D da cena. O algoritmo utilizado para a reconstrução com três câmeras pode ser visto na Seção 4.5.

No Capítulo 5, é apresentado o aplicativo desenvolvido neste trabalho para facilitar as tarefas de calibração, correspondência de pontos e reconstrução. A Seção 5.1 apresenta uma visão geral do sistema, especificando o *hardware* de captura de imagens e apresentando a interface gráfica do programa. Todas as rotinas necessárias para se chegar a uma reconstrução 3D a partir de um conjunto de três câmeras estão disponíveis na interface para serem executadas. Cada uma dessas rotinas é detalhada na Seção 5.2.

Os resultados obtidos neste trabalho são expostos no Capítulo 6. Nesse capítulo é feita uma comparação qualitativa entre os resultados obtidos utilizando duas câmeras e os obtidos usando três câmeras. Por fim, no Capítulo 7, são apresentadas as conclusões e sugeridos possíveis trabalhos a serem realizados futuramente.

Capítulo 2

Modelagem e Reconstrução 3D

Neste capítulo são apresentados conceitos básicos utilizados nos demais capítulos. Inicialmente são definidos os componentes da sala inteligente e é explicado o processamento inicial realizado nas imagens para se obter os dados a serem utilizados como entrada para as rotinas de reconstrução 3D.

A Geometria Projetiva, que tem suas principais definições apresentadas na Seção 2.2, é a ferramenta matemática utilizada para a modelagem das câmeras detalhada na Seção 2.3 e nos algoritmos utilizados na reconstrução tridimensional, explicados na Seção 2.4.

2.1 Modelagem da Sala Inteligente

A sala inteligente deste trabalho, como mostrado na Figura 2.1, é composta por quatro objetos, três câmeras e alguns *grids* de calibração cuja utilização é explicada nas próximas seções. Para facilitar a segmentação da imagem, que não é parte do escopo deste trabalho, os objetos utilizados são retângulos de cores diferentes. Para apontar para os retângulos, o usuário do sistema deve utilizar um apontador, que também é identificado pela sua cor.

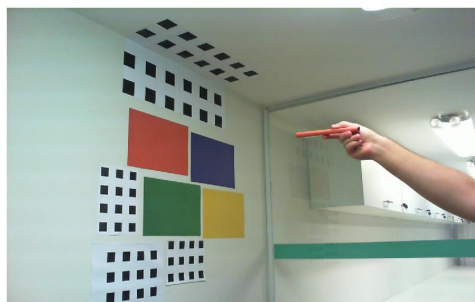


Figura 2.1: Configuração da sala inteligente.

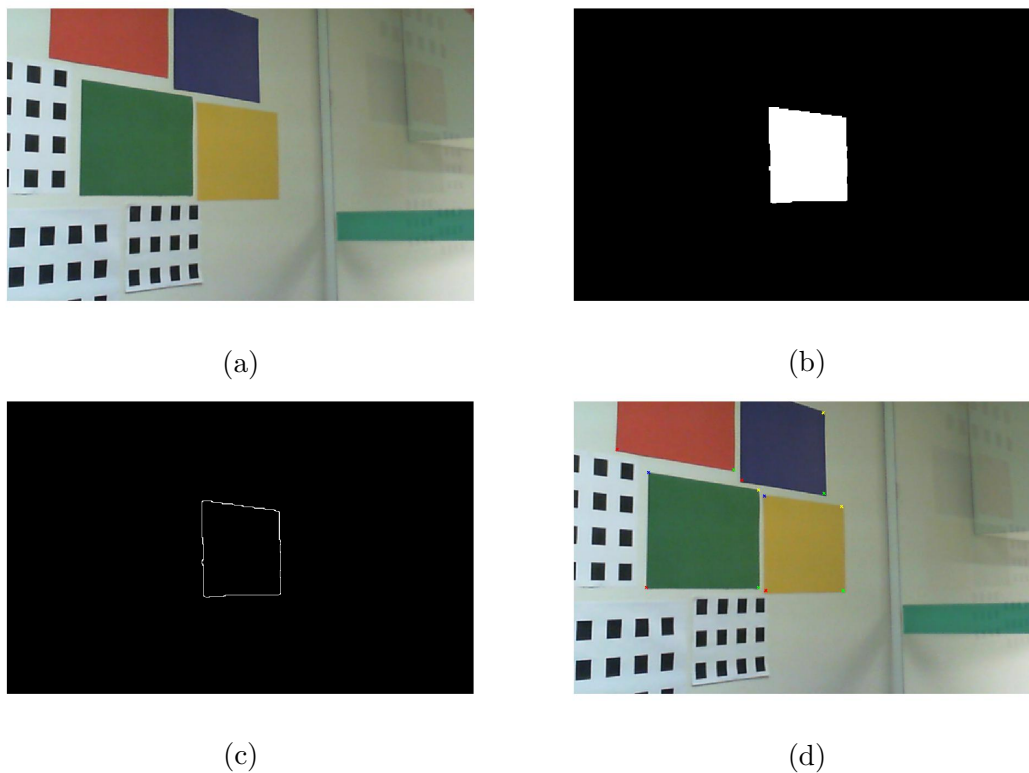


Figura 2.2: Pré-processamento das imagens utilizadas para a reconstrução.

2.1.1 Pré-Processamento

As rotinas de calibração e reconstrução 3D utilizam como entrada as coordenadas dos vértices dos retângulos, das extremidades do apontador e dos pontos de referência nos *grids* de calibração nas imagens capturadas pelas três câmeras. Neste trabalho, os objetos e o apontador foram isolados utilizando processamento baseado em cor, como mostrado nas Figuras 2.2(a) e 2.2(b). Para o cálculo dos vértices dos objetos é realizada uma detecção de bordas (Figura 2.2(c)), seguida por uma detecção de retas. Para cada lado do retângulo é encontrada uma reta e as interseções das retas são os quatro vértices do objeto destacados na Figura 2.2(d). As coordenadas dos pontos de referência para calibração são obtidas manualmente, clicando em cada um dos pontos em todas as imagens de calibração.

2.2 Geometria Projetiva

Neste trabalho, os espaços projetivos de interesse são o de três dimensões e duas dimensões. Um ponto em um espaço projetivo de dimensão N é representado por um vetor de tamanho $N + 1$. Portanto, pontos nos espaços de duas e três dimensões

são representados respectivamente pelos vetores:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (2.1)$$

e

$$\mathbf{X} = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}. \quad (2.2)$$

Dois vetores de tamanho N , $\mathbf{y} = [y_1, y_2, \dots, y_N]^T$ e $\mathbf{z} = [z_1, z_2, \dots, z_N]^T$ representam o mesmo ponto quando $y_n/y_N = z_n/z_N$, $n = 1, 2, \dots, N$. Neste caso, diz-se que \mathbf{y} e \mathbf{z} possuem coordenadas homogêneas.

As retas, no espaço projetivo de dimensão 2 (chamado de plano projetivo), são representadas, assim como os pontos, por vetores de 3 elementos. Dados dois pontos no plano projetivo representados pelos vetores \mathbf{x}_1 e \mathbf{x}_2 , a reta \mathbf{l} que passa pelos dois pontos é definida pelo produto vetorial entre eles: $\mathbf{l} = \mathbf{x}_1 \times \mathbf{x}_2$. De forma análoga, o ponto de interseção \mathbf{x} de duas retas \mathbf{l}_1 e \mathbf{l}_2 é definido por $\mathbf{x} = \mathbf{l}_1 \times \mathbf{l}_2$.

A reta \mathbf{l} , no plano projetivo, é formada por todo ponto \mathbf{x} que satisfaz à equação

$$\mathbf{x}^T \mathbf{l} = 0. \quad (2.3)$$

De forma equivalente o ponto \mathbf{x} é definido por todas as retas \mathbf{l} que satisfazem à mesma equação.

2.3 Modelo de Câmera

A Geometria Projetiva é ideal para modelar a projeção do espaço tridimensional para uma imagem bidimensional. O uso de coordenadas homogêneas permite que a projeção, um mapeamento não linear quando utilizam-se coordenadas euclidianas, seja representada por equações lineares [7] como na equação:

$$\mathbf{x} = \mathbf{P}\mathbf{X} \quad (2.4)$$

onde \mathbf{X} é o vetor de coordenadas 3D de um ponto no espaço, \mathbf{x} é o vetor de coordenadas 2D da imagem do ponto (Figura 2.3) e \mathbf{P} é a matriz de projeção, ou matriz da câmera.

A matriz da câmera possui informações acerca dos parâmetros da câmera. Os parâmetros podem ser divididos em dois grupos: o dos parâmetros que dependem das

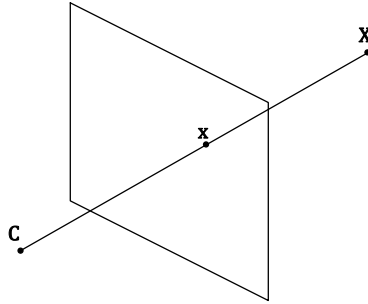


Figura 2.3: Projeção de um ponto \mathbf{X} no espaço no ponto \mathbf{x} no plano focal de uma câmera.

configurações do sistema de coordenadas 3D (parâmetros extrínsecos) e os que dependem das características internas da câmera (parâmetros intrínsecos). É possível decompor a matriz da câmera da seguinte forma:

$$\mathbf{P} = \mathbf{KR} \left[\mathbf{I} \mid \mathbf{C} \right] \quad (2.5)$$

na qual \mathbf{K} é a matriz composta pelos parâmetros intrínsecos ou matriz de calibração, \mathbf{R} é a matriz que contém informações acerca dos parâmetros extrínsecos e \mathbf{C} é a posição do centro da câmera no sistema de coordenadas 3D.

O procedimento de obtenção da matriz da câmera através de um processo de calibração com um *grid* de calibração como referencial global é explicado em [3]. A decomposição da matriz da câmera para obter a matriz de calibração é apresentada em [7] (p. 163, Seção 6.2.4). O código utilizado neste trabalho para obtenção da matriz da câmera e sua respectiva matriz de calibração pode ser visto no Apêndice A.1.

A Figura 2.4 apresenta o *grid* de calibração utilizado neste trabalho. Para se realizar a calibração é necessário escolher no *grid* um ponto como sendo a origem do sistema de coordenadas 3D e medir manualmente as coordenadas de alguns pontos que serão utilizados no algoritmo de calibração. No *grid* da imagem, foram utilizados como pontos de referência para calibração alguns dos vértices dos quadrados pretos.

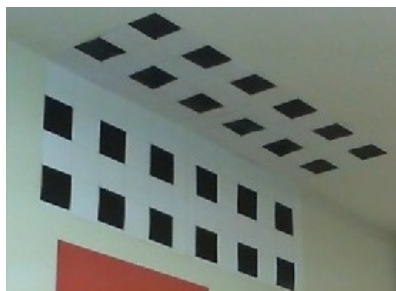


Figura 2.4: *Grid* de calibração.

2.4 Reconstrução 3D de um ponto

Reconstrução 3D é a obtenção das coordenadas euclidianas de um ponto no espaço a partir de, em geral, duas ou mais imagens do ponto.

Dado um ponto representado na Geometria Projetiva pelo vetor \mathbf{X} , cuja imagem capturada por uma câmera i é representada pelo vetor \mathbf{x}_i , sendo \mathbf{m}_i o vetor que representa as coordenadas euclidianas da imagem desse ponto e \mathbf{M} , as coordenadas euclidianas no espaço, os elementos de \mathbf{m}_i e \mathbf{M} podem ser obtidos, respectivamente, dividindo os dois primeiros elementos de \mathbf{x}_i pelo terceiro e os três primeiros elementos de \mathbf{X} pelo quarto.

Reescrevendo a Equação (2.4) para cada câmera i como

$$\mathbf{x}_i = \begin{bmatrix} \mathbf{p}_1^i & p_{14}^i \\ \mathbf{p}_2^i & p_{24}^i \\ \mathbf{p}_3^i & p_{34}^i \end{bmatrix} \mathbf{X} \quad (2.6)$$

onde \mathbf{p}_j^i são os três primeiros elementos da j -ésima linha e p_{4j}^i o quarto elemento da mesma linha, reorganizando as equações e utilizando coordenadas euclidianas, pode-se obter a seguinte equação [6], [3]:

$$\begin{bmatrix} \mathbf{p}_1^{iT} & -m_1^i \mathbf{p}_3^{iT} \\ \mathbf{p}_2^{iT} & -m_2^i \mathbf{p}_3^{iT} \end{bmatrix} \mathbf{M} = \begin{bmatrix} -p_{14}^i + m_1^i p_{34}^i \\ -p_{24}^i + m_2^i p_{34}^i \end{bmatrix}. \quad (2.7)$$

Para N imagens de diferentes câmeras pode-se escrever

$$\begin{bmatrix} \mathbf{p}_1^{1T} & -m_1^1 \mathbf{p}_3^{1T} \\ \mathbf{p}_2^{1T} & -m_2^1 \mathbf{p}_3^{1T} \\ \mathbf{p}_1^{2T} & -m_1^2 \mathbf{p}_3^{2T} \\ \mathbf{p}_2^{2T} & -m_2^2 \mathbf{p}_3^{2T} \\ \vdots & \vdots \\ \mathbf{p}_1^{NT} & -m_1^N \mathbf{p}_3^{NT} \\ \mathbf{p}_2^{NT} & -m_2^N \mathbf{p}_3^{NT} \end{bmatrix} \mathbf{M} = \begin{bmatrix} -p_{14}^1 + m_1^1 p_{34}^1 \\ -p_{24}^1 + m_2^1 p_{34}^1 \\ -p_{14}^2 + m_1^2 p_{34}^2 \\ -p_{24}^2 + m_2^2 p_{34}^2 \\ \vdots \\ -p_{14}^N + m_1^N p_{34}^N \\ -p_{24}^N + m_2^N p_{34}^N \end{bmatrix}, \quad (2.8)$$

ou de forma reduzida

$$\mathbf{A}\mathbf{M} = \mathbf{b}. \quad (2.9)$$

Resolvendo a equação anterior pelo método dos mínimos quadrados [6], [3], tem-se que

$$\mathbf{M} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}. \quad (2.10)$$

Capítulo 3

Usando Duas Câmeras

No Capítulo 2 foi explicado como se pode realizar a reconstrução 3D de um ponto utilizando duas ou mais câmeras calibradas. As matrizes das câmeras obtidas pelo processo de calibração descrito naquele capítulo dependem apenas das características de suas respectivas câmeras e de um *grid* de calibração.

Este capítulo mostra como obter informações sobre duas câmeras utilizando simultaneamente informações das imagens capturadas por elas, sem a utilização de informações sobre as coordenadas 3D de pontos em um *grid* de calibração.

A geometria que descreve as relações entre duas imagens neste trabalho é chamada de Geometria Epipolar e é descrita na Seção 3.1. Em termos de Geometria Epipolar, a principal descrição da relação entre duas câmeras é a matriz fundamental, cujas principais propriedades são apresentadas na Seção 3.2. A Seção 3.3 trata da Matriz Essencial, uma especialização da Matriz Fundamental para um caso especial de coordenadas das imagens. Das matrizes Essencial e Fundamental são extraídas as matrizes das câmeras utilizando-se os métodos explicados na Seção 3.4. Essa seção mostra que a reconstrução realizada a partir da matriz fundamental apresenta uma distorção projetiva que interfere na visualização do resultado e que essa distorção não existe quando se utiliza a matriz essencial. O algoritmo completo usado para a reconstrução com duas câmeras neste trabalho é apresentado na Seção 3.5.

3.1 Geometria Epipolar

A Geometria Epipolar é a geometria que descreve as principais relações entre duas imagens [7]. Ela depende apenas dos parâmetros intrínsecos e da posição relativa entre duas câmeras e não de um sistema de coordenadas global. A Geometria Epipolar surge como resposta ao problema de, dadas as coordenadas de um ponto \mathbf{x} em uma imagem e coordenadas do ponto \mathbf{x}' correspondente ao mesmo ponto \mathbf{X} no espaço em outra imagem, definir as relações entre os dois pontos.

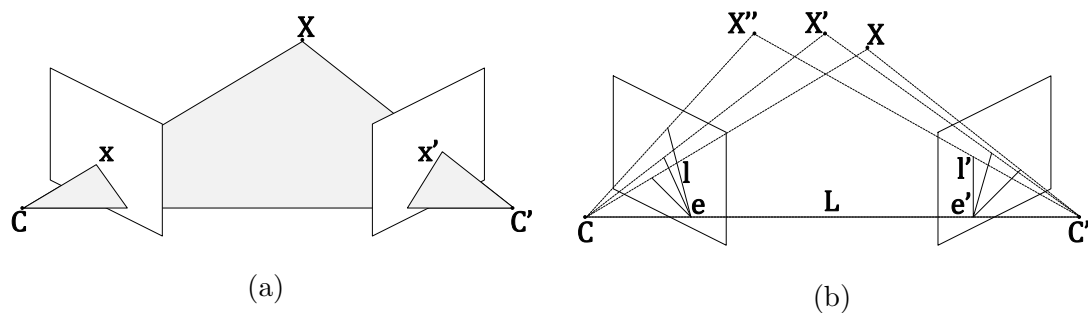


Figura 3.1: Geometria epipolar.

A Figura 3.1(a) deixa claro que os raios retroprojetados (retas definidas pelas junções das imagens \mathbf{x} e \mathbf{x}' com os centros de suas respectivas câmeras \mathbf{C} e \mathbf{C}') se interceptam no ponto \mathbf{X} e são, portanto, coplanares. Variando a posição do ponto \mathbf{X} , obtém-se um conjunto de planos em forma de feixe (Figura 3.1(b)), cuja interseção, ou eixo do feixe, é a reta que une os centros das duas câmeras. Ao plano definido pelos raios retroprojetados denomina-se plano epipolar. As retas de interseção do plano epipolar com os planos focais (\mathbf{l} e \mathbf{l}') são chamadas de retas epipolares. Os pontos de interseção do eixo do feixe de planos com os planos focais \mathbf{e} e \mathbf{e}' são chamados de epípolos. Todas as retas epipolares se interceptam no epípolo.

3.2 Matriz Fundamental

Definidas as entidades geométricas mais importantes na Geometria Epipolar, pode-se descrever as relações matemáticas entre as imagens. Dadas as coordenadas \mathbf{x} da primeira imagem de um ponto \mathbf{X} no espaço, a imagem \mathbf{x}' do ponto na segunda imagem estará obrigatoriamente sobre a interseção do plano epipolar com o plano focal da segunda câmera, ou seja, a reta epipolar da segunda câmera para o ponto \mathbf{x} dado na primeira câmera. Há então um mapeamento de um ponto na primeira imagem para uma linha na segunda, como ilustrado na Figura 3.2, onde a imagem do ponto \mathbf{x}' na segunda câmera, correspondente ao ponto \mathbf{x} na primeira, está localizada necessariamente na reta epipolar \mathbf{l}' . Tal mapeamento é representado pela matriz fundamental.

3.2.1 Cálculo da Matriz Fundamental a Partir das Matrizes das Câmeras

Se as matrizes \mathbf{P} e \mathbf{P}' das duas câmeras são conhecidas, o raio retroprojetado de \mathbf{x} é definido pela equação $\mathbf{X}(\lambda) = \mathbf{P}^+ \mathbf{x} + \lambda \mathbf{C}$, onde \mathbf{P}^+ é a pseudo-inversa da matriz \mathbf{P} . Para definir a reta epipolar \mathbf{l}' basta encontrar a equação da reta que passa pelas imagens de dois pontos da reta retroprojetada de \mathbf{x} . Usando os pontos $\mathbf{X}(0) = \mathbf{P}^+ \mathbf{x}$

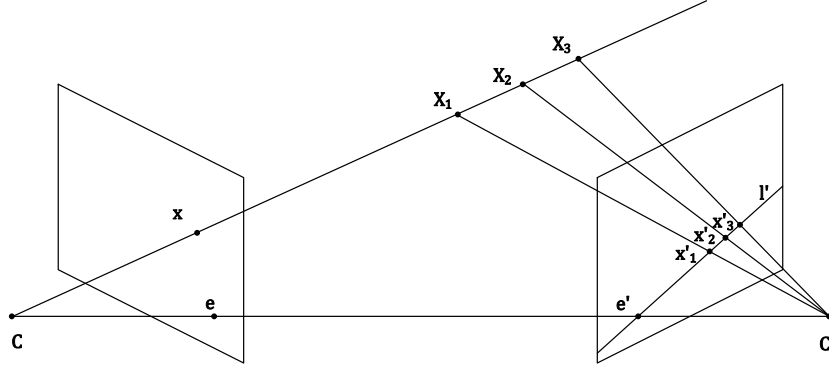


Figura 3.2: Mapeamento de um ponto na imagem na primeira câmera em uma reta epipolar na imagem da segunda.

e $\mathbf{X}(\infty) = \mathbf{C}$, suas imagens na segunda câmera são $\mathbf{P}'\mathbf{P}^+\mathbf{x}$ e $\mathbf{P}'\mathbf{C}$ respectivamente. O segundo ponto é a imagem na segunda câmera do centro da primeira câmera, portanto é o epipolo \mathbf{e}' .

A equação da linha \mathbf{l}' é então a junção de $\mathbf{P}'\mathbf{P}^+\mathbf{x}$ e $\mathbf{P}'\mathbf{C}$:

$$\mathbf{l}' = \mathbf{P}'\mathbf{C} \times \mathbf{P}'\mathbf{P}^+\mathbf{x} = \mathbf{e}' \times \mathbf{P}'\mathbf{P}^+\mathbf{x} = [\mathbf{e}']_{\times} \mathbf{P}'\mathbf{P}^+\mathbf{x}, \quad (3.1)$$

onde $[\mathbf{e}']_{\times}$ é a matriz antissimétrica que tem como vetor nulo o vetor \mathbf{e}' .

E, definindo $\mathbf{F} = \mathbf{e}' \times \mathbf{P}'\mathbf{P}^+$ como a matriz fundamental, que representa a relação entre \mathbf{l}' e \mathbf{x} :

$$\mathbf{l}' = \mathbf{F}\mathbf{x}. \quad (3.2)$$

Também é possível obter uma relação entre dois pontos \mathbf{x} e \mathbf{x}' através da matriz fundamental. Se \mathbf{x}' é a imagem do ponto \mathbf{X} na segunda câmera, ele obrigatoriamente faz parte da reta epipolar \mathbf{l}' . Portanto, $\mathbf{x}'^T \mathbf{l}' = 0$ e substituindo \mathbf{l}' por seu valor na Equação (3.2) tem-se a relação

$$\mathbf{x}'^T \mathbf{F}\mathbf{x} = 0. \quad (3.3)$$

3.2.2 Cálculo da Matriz Fundamental a Partir da Correspondência de Pontos

Na Seção 3.2.1 mostrou-se que a matriz fundamental de um par de câmeras pode ser computada utilizando as matrizes das câmeras. Porém é possível calcular a matriz fundamental utilizando apenas as coordenadas de pontos correspondentes nas duas imagens.

A partir da Equação (3.3), para n correspondências de pontos, onde $\mathbf{x}_i = [x_i, y_i, 1]^T$ e $\mathbf{x}'_i = [x'_i, y'_i, 1]^T$ formam o i -ésimo par de correspondência, pode-se obter a equação

$$\mathbf{A}\mathbf{f} = \begin{pmatrix} x'_1x_1 & x'_1y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_nx_n & x'_ny_n & x'_n & y'_nx_n & y'_ny_n & y'_n & x_n & y_n & 1 \end{pmatrix} \mathbf{f} = \mathbf{0}, \quad (3.4)$$

onde \mathbf{f} é o vetor de elementos de \mathbf{F} ordenados por linha.

A matriz \mathbf{A} precisa ter posto igual a 8 para que haja uma solução única para \mathbf{f} a menos de um fator de escala. Mas como os dados não são exatos devido ao ruído nas coordenadas dos pontos, então o posto de \mathbf{A} deve ser igual a 9 e deve-se encontrar uma solução pelo método dos mínimos quadrados [7], que pode ser calculada através da decomposição em valores singulares. Se a matriz \mathbf{A} tem como decomposição em valores singulares $\mathbf{A} = \mathbf{UDV}^T$, com os elementos da diagonal de \mathbf{D} dispostos em ordem decrescente, o vetor \mathbf{f} que minimiza $\|\mathbf{A}\mathbf{f}\|$ é a última coluna da matriz \mathbf{V} .

Uma propriedade de toda matriz fundamental é que ela deve ser singular. Como a solução encontrada computacionalmente não resultará em uma matriz singular devido às imprecisões de medição, deve-se substituir a matriz encontrada \mathbf{F} pela matriz singular \mathbf{F}' encontrada minimizando a norma $\|\mathbf{F} - \mathbf{F}'\|$ com o determinante de \mathbf{F}' igual a zero [7]. Sendo $\mathbf{F} = \mathbf{UDV}^T$ uma decomposição em valores singulares com os elementos de \mathbf{D} ordenados de forma decrescente, a matriz singular \mathbf{F}' que minimiza $\|\mathbf{F} - \mathbf{F}'\|$ é $\mathbf{F}' = \mathbf{UD}'\mathbf{V}^T$, onde \mathbf{D}' é obtido da matriz \mathbf{D} substituindo o terceiro e menor elemento de sua diagonal por zero [7].

3.2.3 Normalização

Na matriz \mathbf{A} da Equação (3.4) há elementos que, considerando câmeras com resolução *full* HD, podem chegar à ordem de grandeza de 10^6 (x'_ix_j , x'_iy_j , y'_iy_j e y'_ix_j), outros até a ordem de 10^3 (x'_i , x_i , y'_i , y_i) e outros iguais a 1. Ao se calcular \mathbf{f} a partir da decomposição em valores singulares de uma matriz \mathbf{A} cuja ordem de grandeza dos elementos varia muito de acordo com a coluna, o resultado obtido tenderia a negligenciar as colunas de \mathbf{A} com os elementos de menor ordem de grandeza.

Para resolver esse problema, é necessário realizar uma normalização nas coordenadas dos pontos de referência em ambas as imagens. Neste trabalho, a normalização foi realizada de forma que o centroide para cada conjunto de pontos fosse a origem do sistema de coordenadas e o comprimento médio dos vetores que representam os pontos fosse igual a $\sqrt{2}$ [7].

Sendo \mathbf{t} e \mathbf{t}' os vetores de translação para os pontos das imagens das duas câmeras que levam os centroides às origens dos respectivos sistemas de coordenadas e sendo S e S' os fatores de escala que devem ser aplicados aos pontos das duas imagens para que os comprimentos médios dos vetores que representam os pontos em cada imagem sejam iguais a $\sqrt{2}$, e por fim utilizando coordenadas projetivas de forma

que $\mathbf{x}_i^T = [x_i, y_i, 1]$ e $\mathbf{x}'_i{}^T = [x'_i, y'_i, 1]$, a normalização pode ser expressa por:

$$\hat{\mathbf{x}}_i = \begin{bmatrix} 1 & 0 & \mathbf{t} \\ 0 & 1 & \\ 0 & 0 & 1/S \end{bmatrix} \mathbf{x}_i = \mathbf{H}\mathbf{x}_i \quad (3.5)$$

e

$$\hat{\mathbf{x}}'_i = \begin{bmatrix} 1 & 0 & \mathbf{t}' \\ 0 & 1 & \\ 0 & 0 & 1/S' \end{bmatrix} \mathbf{x}'_i = \mathbf{H}'\mathbf{x}'_i. \quad (3.6)$$

3.3 Matriz Essencial

A matriz essencial é o caso particular da matriz fundamental quando as matrizes das câmeras são normalizadas com relação a suas matrizes de calibração. A normalização apresentada nesta seção não tem relação com a normalização explicada na Seção 3.2.3. Enquanto a normalização apresentada anteriormente servia apenas para fins numéricos, a desta seção serve para melhor representação do significado físico das matrizes de câmera. Uma matriz de câmera é dita normalizada quando a uma matriz de câmera genérica na forma $\mathbf{P} = \mathbf{K} \begin{bmatrix} \mathbf{R} & | & \mathbf{t} \end{bmatrix}$ é aplicada a inversa da matriz de calibração \mathbf{K}^{-1} . A matriz normalizada é dada, então, por

$$\mathbf{K}^{-1}\mathbf{P} = \begin{bmatrix} \mathbf{R} & | & \mathbf{t} \end{bmatrix}. \quad (3.7)$$

Dadas as duas matrizes de câmera normalizadas $\mathbf{P} = \begin{bmatrix} \mathbf{I} & | & \mathbf{0} \end{bmatrix}$ e $\mathbf{P}' = \begin{bmatrix} \mathbf{R} & | & \mathbf{t} \end{bmatrix}$, de acordo com a Equação (3.1) a matriz essencial pode ser escrita como

$$\mathbf{E} = [\mathbf{t}]_{\times} \mathbf{R} \quad (3.8)$$

Neste trabalho, a matriz essencial é calculada a partir da matriz fundamental e com conhecimento prévio das matrizes de calibração das câmeras. Da Equação (3.3), segue que para o caso de coordenadas normalizadas é válida a relação

$$\mathbf{x}'^T \mathbf{E} \mathbf{x} = 0, \quad (3.9)$$

cujo lado esquerdo pode ser reescrito como $(\mathbf{x}'^T \mathbf{K}'^T)((\mathbf{K}'^T)^{-1} \mathbf{E} \mathbf{K}^{-1})(\mathbf{K} \mathbf{x})$. Os fatores $(\mathbf{x}'^T \mathbf{K}'^T)$ e $(\mathbf{K} \mathbf{x})$ são o vetor de coordenadas de um ponto na segunda imagem transposto e o vetor de coordenadas do ponto correspondente na primeira imagem, respectivamente, nos sistemas de coordenadas originais (não normalizados). O fator restante $(\mathbf{K}'^T)^{-1} \mathbf{E} \mathbf{K}^{-1}$ deve, de acordo com a Equação (3.3), ser correspondente à matriz fundamental \mathbf{F} . Então, tem-se a relação

$$\mathbf{E} = \mathbf{K}'^T \mathbf{F} \mathbf{K}, \quad (3.10)$$

a qual é utilizada para se obter a matriz essencial a partir da fundamental.

3.4 Obtendo as Matrizes das Câmeras

Obtidas as matrizes fundamental e essencial, é possível, a partir delas, recuperar as matrizes das câmeras e usá-las para realizar a reconstrução da cena. O par de matrizes de câmera obtido utilizando a matriz fundamental é determinado a menos de uma ambiguidade projetiva, como será explicado na Seção 3.4.1.

Essa ambiguidade na determinação das matrizes das câmeras não interfere no objetivo deste trabalho, mas apenas na estrutura dos objetos reconstruídos. Ou seja, os objetos reconstruídos usando essas matrizes apresentam uma distorção projetiva, mas tal distorção não interfere na verificação de qual objeto está sendo selecionado.

Apesar de a ambiguidade projetiva não interferir nos resultados deste trabalho, para melhor visualização da reconstrução da cena a ambiguidade pode ser removida utilizando-se a matriz essencial. As matrizes das câmeras recuperadas através dela não apresentam ambiguidade projetiva, resultando, então, em uma reconstrução dos objetos com menor distorção. Esse procedimento é apresentado na Seção 3.4.2.

3.4.1 Obtendo as Matrizes das Câmeras através da Matriz Fundamental

Para obter um possível par de matrizes de câmera \mathbf{P} e \mathbf{P}' utilizando a matriz fundamental \mathbf{F} , uma possibilidade é partir da Equação (3.3). Substituindo \mathbf{x} por $\mathbf{P}\mathbf{X}$ e \mathbf{x}' por $\mathbf{P}'\mathbf{X}$ nessa equação tem-se que

$$\mathbf{X}^T \mathbf{P}'^T \mathbf{F} \mathbf{P} \mathbf{X} = 0, \quad (3.11)$$

de onde conclui-se que $\mathbf{P}'^T \mathbf{F} \mathbf{P}$ é antissimétrica.

Em [7] está demonstrado (p. 256, resultado 9.13) que um par de matrizes de câmera que respeita essa relação é $\mathbf{P} = [\mathbf{I} \mid \mathbf{0}]$ e $\mathbf{P}' = [\mathbf{S}\mathbf{F} \mid \mathbf{e}']$, onde \mathbf{S} é uma matriz antissimétrica. Substituindo as matrizes diretamente na expressão e desenvolvendo, tem-se que $\mathbf{P}'^T \mathbf{F} \mathbf{P} = [\mathbf{S}\mathbf{F} \mid \mathbf{e}']^T \mathbf{F} [\mathbf{I} \mid \mathbf{0}] = [\mathbf{S}\mathbf{F} \mid \mathbf{e}']^T [\mathbf{F} \mid \mathbf{0}] = \begin{bmatrix} \mathbf{F}^T \mathbf{S}^T \mathbf{F} & \mathbf{0} \\ \mathbf{e}'^T \mathbf{F} & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{F}^T \mathbf{S}^T \mathbf{F} & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}$ que é antissimétrica, provando que a relação é respeitada.

É necessário também que a matriz \mathbf{P}' na forma sugerida seja uma matriz de câmera válida (com posto igual a 3). Para isso basta que $\mathbf{s}^T \mathbf{e}' \neq 0$, onde \mathbf{s} é o vetor

nulo de \mathbf{S} , como provado a seguir. Sendo \mathbf{f}_1 , \mathbf{f}_2 e \mathbf{f}_3 as três colunas de \mathbf{F} , tem-se que $\mathbf{SF} = \begin{bmatrix} \mathbf{s} \times \mathbf{f}_1 & \mathbf{s} \times \mathbf{f}_2 & \mathbf{s} \times \mathbf{f}_3 \end{bmatrix}$ e o espaço gerado por suas colunas é, portanto, o plano perpendicular a \mathbf{s} e tem posto igual a 2. Como se $\mathbf{s}^T \mathbf{e}' \neq 0$, \mathbf{e}' não é perpendicular a \mathbf{s} , então \mathbf{e}' não está no espaço gerado pelas colunas de \mathbf{SF} . Logo $\begin{bmatrix} \mathbf{SF} & | & \mathbf{e}' \end{bmatrix}$ tem posto 3, como requerido.

Uma escolha possível para \mathbf{S} é a matriz antissimétrica cujo vetor nulo é \mathbf{e}' , denotada por $[\mathbf{e}']_{\times}$. Com essa escolha, o par de câmeras utilizado é, então,

$$\mathbf{P} = \begin{bmatrix} \mathbf{I} & | & \mathbf{0} \end{bmatrix} \quad (3.12)$$

e

$$\mathbf{P}' = \begin{bmatrix} [\mathbf{e}']_{\times} \mathbf{F} & | & \mathbf{e}' \end{bmatrix}. \quad (3.13)$$

O par de matrizes de câmera escolhido dessa maneira não é único. Outras escolhas para \mathbf{S} resultariam em diferentes pares de matrizes de câmera compatíveis com a mesma matriz fundamental. Isso pode ser explicado a partir da equação $\mathbf{X}^T \mathbf{P}'^T \mathbf{F} \mathbf{P} \mathbf{X} = \mathbf{0}$. Dada uma matriz \mathbf{H} , 4×4 , substituir o ponto \mathbf{X} por $\mathbf{H}\mathbf{X}$ e as matrizes de câmera \mathbf{P} e \mathbf{P}' por $\mathbf{P}\mathbf{H}^{-1}$ e $\mathbf{P}'\mathbf{H}^{-1}$ não altera o resultado da equação, pois $\mathbf{X}^T \mathbf{H}^T \mathbf{H}^{-1T} \mathbf{P}'^T \mathbf{F} \mathbf{P} \mathbf{H}^{-1} \mathbf{H} \mathbf{X} = \mathbf{X}^T \mathbf{P}'^T \mathbf{F} \mathbf{P} \mathbf{X} = \mathbf{0}$. Então todos os pares de câmeras definidos dessa maneira são compatíveis com a mesma matriz fundamental.

Esse fato afeta a reconstrução 3D de um ponto a partir de um par de câmeras obtido de uma matriz fundamental, uma vez que os pontos \mathbf{x} e \mathbf{x}' nas imagens da primeira e segunda câmera podem ser imagens de diversos pontos $\mathbf{H}\mathbf{X}$ no espaço, dependendo da escolha das matrizes de câmera. A imagem reconstruída apresentará, então, uma distorção projetiva determinada pela matriz \mathbf{H} . Essa distorção pode inviabilizar a visualização da cena reconstruída e deve, portanto, ser eliminada. Esta consideração, associada à consideração que aparece no último parágrafo da Seção 3.4.2, justifica a utilização da matriz essencial ao invés da matriz fundamental.

3.4.2 Obtendo as Matrizes das Câmeras através da Matriz Essencial

A extração das matrizes das câmeras da matriz essencial é realizada através de uma decomposição da matriz essencial que permita a obtenção do vetor de translação \mathbf{t} e da matriz de rotação \mathbf{R} , como na Equação (3.8), que pode ser reescrita como $\mathbf{E} = \mathbf{S}\mathbf{R}$, onde \mathbf{S} é a matriz antissimétrica que tem \mathbf{t} como vetor nulo à direita.

Segundo [7] toda matriz antissimétrica \mathbf{S} pode ser escrita como $\mathbf{S} = \mathbf{U}\mathbf{Z}\mathbf{U}^T$,

onde \mathbf{U} é uma matriz ortogonal e

$$\mathbf{Z} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \quad (3.14)$$

A matriz \mathbf{Z} pode ser escrita, a menos do sinal, como \mathbf{DW} onde

$$\mathbf{W} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ e } \mathbf{D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \quad (3.15)$$

Então tem-se que $\mathbf{SR} = \mathbf{UDWU}^T\mathbf{R}$. Como \mathbf{W} , \mathbf{U} e \mathbf{R} são matrizes ortogonais, o produto $\mathbf{WD}^T\mathbf{R}$ é também ortogonal e, portanto, $\mathbf{SR} = \mathbf{UDV}^T$, onde $\mathbf{V}^T = \mathbf{WU}^T\mathbf{R}$ é uma decomposição em valor singular da matriz \mathbf{SR} .

Então, se a decomposição em valor singular de uma matriz essencial \mathbf{E} seguindo a mesma notação é dada por $\mathbf{E} = \mathbf{UDV}^T$, com

$$\mathbf{D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad (3.16)$$

a matriz \mathbf{S} é recuperada fazendo $\mathbf{S} = \mathbf{UZU}^T$ e há apenas duas possibilidades para a matriz \mathbf{R} : $\mathbf{R} = \mathbf{UW}^T\mathbf{V}^T$ ou $\mathbf{R} = \mathbf{UWV}^T$ [7].

Para provar que existem essas e apenas essas duas formas para a matriz \mathbf{R} , basta escrever $\mathbf{E} = \mathbf{SR} = (\mathbf{UZU}^T)(\mathbf{UXV}^T)$, onde \mathbf{X} é alguma rotação. Tem-se que $(\mathbf{UZU}^T)(\mathbf{UXV}^T) = \mathbf{U}(\mathbf{ZX})\mathbf{V}^T$ e, portanto, $\mathbf{ZX} = \mathbf{D}$. Como \mathbf{X} é uma rotação, então $\mathbf{X} = \mathbf{W}$ ou $\mathbf{X} = \mathbf{W}^T$.

Como \mathbf{t} é o vetor nulo à direita de \mathbf{S} , tem-se que $\mathbf{St} = \mathbf{UZU}^T\mathbf{t} = \mathbf{0}$, que tem como solução, a menos do sinal, $\mathbf{t} = \mathbf{U} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$, ou $\mathbf{t} = \mathbf{u}_3$, a terceira coluna da matriz \mathbf{U} .

Portanto, existem quatro possíveis soluções para a matriz de câmera \mathbf{P}' considerando o sinal de \mathbf{t} e a escolha de \mathbf{R} , $\mathbf{P}' = [\mathbf{UWV}^T \mid \mathbf{u}_3]$, $\mathbf{P}' = [\mathbf{UWV}^T \mid -\mathbf{u}_3]$, $\mathbf{P}' = [\mathbf{UW}^T\mathbf{V}^T \mid \mathbf{u}_3]$ e $\mathbf{P}' = [\mathbf{UW}^T\mathbf{V}^T \mid -\mathbf{u}_3]$. A escolha do sinal de \mathbf{u}_3 implica na inversão do sentido da translação da câmera. A relação entre as duas possibilidades para \mathbf{R} é

$$\mathbf{UW}^T\mathbf{V}^T = (\mathbf{UW}^T\mathbf{W}^T\mathbf{U}^T)\mathbf{UWV}^T = (\mathbf{U} \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{U}^T)\mathbf{UWV}^T, \quad (3.17)$$

que equivale a uma rotação de 180° em torno da terceira coluna de \mathbf{U} (o vetor \mathbf{u}_3), que pode ser interpretada como uma rotação de 180° ao redor da linha base que une o centro das duas câmeras.

Como visto, o par de matrizes de câmera obtido a partir da matriz essencial não apresenta a ambiguidade projetiva observada nas matrizes obtidas através da matriz fundamental. A utilização das matrizes de calibração para a obtenção da matriz essencial serve então para que se possa obter uma reconstrução sem distorção projetiva, permitindo uma visualização adequada da reconstrução das cenas.

3.5 Algoritmo

Neste capítulo é apresentado o algoritmo utilizado neste trabalho para obtenção das matrizes \mathbf{P} e \mathbf{P}' de duas câmeras através da matriz fundamental. O código utilizado no aplicativo do trabalho, baseado neste algoritmo, pode ser visto no Apêndice A.2.

Entradas: N pares de pontos $(\mathbf{x}_i, \mathbf{x}'_i) = ([x_i, y_i], [x'_i, y'_i])$ nas imagens das duas câmeras com cada par correspondendo ao mesmo ponto \mathbf{X}_i no espaço.

1. Normalizar os vetores \mathbf{x}_i e \mathbf{x}'_i de forma que o centroide para cada conjunto de vetores seja a origem do sistema de coordenadas e o comprimento médio dos vetores seja igual a $\sqrt{2}$, obtendo as matrizes de normalização \mathbf{H} e \mathbf{H}' , conforme as Equações (3.5) e (3.6).
2. Calcular a matriz fundamental $\hat{\mathbf{F}}$, a partir da equação $\mathbf{A}\mathbf{f} = \mathbf{0}$, (Equação (3.4)).
 - (a) Encontrar a matriz \mathbf{A} a partir das coordenadas normalizadas.
 - (b) Realizar a decomposição em valores singulares $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$.
 - (c) Fazer $\mathbf{f} = \mathbf{v}_N$, onde \mathbf{v}_N é a última coluna de \mathbf{V} e ordenar os elementos do vetor \mathbf{f} formando a matriz $\hat{\mathbf{F}}$.
3. Encontrar a matriz singular $\hat{\mathbf{F}}'$ que minimiza $\|\hat{\mathbf{F}} - \hat{\mathbf{F}}'\|$.
 - (a) Realizar a decomposição em valores singulares $\hat{\mathbf{F}} = \mathbf{U}\mathbf{D}\mathbf{V}^T$.
 - (b) Fazer $\hat{\mathbf{F}}' = \mathbf{U}\mathbf{D}'\mathbf{V}^T$, onde \mathbf{D}' é igual à matriz \mathbf{D} com o terceiro elemento de sua diagonal substituído por zero.
4. Encontrar a matriz \mathbf{F} para coordenadas desnormalizadas: $\mathbf{F} = \mathbf{H}'^T \hat{\mathbf{F}} \mathbf{H}$.
5. Calcular o epipolo \mathbf{e}' , usando a equação $\mathbf{F}^T \mathbf{e}' = \mathbf{0}$.
 - (a) Realizar a decomposição em valores singulares $\mathbf{F}^T = \mathbf{U}\mathbf{D}\mathbf{V}^T$.

- (b) Fazer $\mathbf{e} = \mathbf{v}_3$, onde \mathbf{v}_3 é a terceira coluna de \mathbf{V} .
6. Utilizando o resultado da Seção 3.4.1, fazer $\mathbf{P} = \begin{bmatrix} \mathbf{I} & | & \mathbf{0} \end{bmatrix}$ e $\mathbf{P}' = \begin{bmatrix} [\mathbf{e}]_{\times} \mathbf{F} & | & \mathbf{e}' \end{bmatrix}$.
7. Obter a reconstrução com distorção projetiva. A partir de \mathbf{P} e \mathbf{P}' , calcular as coordenadas 3D dos pontos a serem reconstruídos resolvendo a Equação (2.8) para duas câmeras.
8. Obter a reconstrução sem distorção projetiva.
- (a) Calcular a matriz essencial através da Equação (3.10): $\mathbf{E} = \mathbf{K}'^T \mathbf{F} \mathbf{K}$.
- (b) Realizar a decomposição em valores singulares $\mathbf{E} = \mathbf{U} \mathbf{D} \mathbf{V}^T$.
- (c) Conforme o desenvolvimento da Seção 3.4.2, fazer $\mathbf{P} = \mathbf{K} \begin{bmatrix} \mathbf{I} & | & \mathbf{0} \end{bmatrix}$ e $\mathbf{P}' = \mathbf{K}' \begin{bmatrix} \mathbf{U} \mathbf{W} \mathbf{V}^T & | & \mathbf{u}_3 \end{bmatrix}$, ou $\mathbf{P}' = \mathbf{K}' \begin{bmatrix} \mathbf{U} \mathbf{W} \mathbf{V}^T & | & -\mathbf{u}_3 \end{bmatrix}$, ou $\mathbf{P}' = \mathbf{K}' \begin{bmatrix} \mathbf{U} \mathbf{W}^T \mathbf{V}^T & | & \mathbf{u}_3 \end{bmatrix}$, ou $\mathbf{P}' = \mathbf{K}' \begin{bmatrix} \mathbf{U} \mathbf{W}^T \mathbf{V}^T & | & -\mathbf{u}_3 \end{bmatrix}$. Todas as quatro alternativas são exploradas.
- (d) A partir de \mathbf{P} e de cada alternativa para \mathbf{P}' , calcular as coordenadas 3D dos pontos a serem reconstruídos resolvendo a Equação (2.8) para duas câmeras. Verificar, analisando os resultados, qual opção de \mathbf{P}' é a correta.

Capítulo 4

Usando Três Câmeras

De forma semelhante ao caso de duas câmeras, analisando a projeção de uma cena em três câmeras é possível estabelecer relações entre as três imagens. A entidade matemática que engloba essas relações é o tensor trifocal, detalhado na Seção 4.1. O tensor trifocal, assim como a matriz fundamental, pode ser calculado a partir das coordenadas nas três imagens das projeções de um conjunto de pontos de referências, como pode ser visto na Seção 4.2.

Após a obtenção do tensor trifocal, pode-se extrair dele as matrizes das câmeras. Para isso, primeiro são calculadas as duas matrizes fundamentais que relacionam a primeira com a segunda câmera e a primeira com a terceira câmera, como mostra a Seção 4.3. Na Seção 4.4 é visto como, das matrizes fundamentais, extrair as três matrizes das câmeras. Esse procedimento é semelhante ao explicado no Capítulo 3, mas deve-se tomar cuidado para que a matriz da segunda e a da terceira câmera sejam compatíveis, já que as matrizes fundamentais utilizadas para o cálculo das matrizes das câmeras relacionam a segunda e a terceira câmeras apenas com a primeira, mas não uma com a outra.

Na Seção 4.5, é apresentado o algoritmo utilizado para a reconstrução com três câmeras. Nesse algoritmo, assim como no caso de duas câmeras, são utilizadas as matrizes de calibração das câmeras para obter as matrizes essenciais e remover a ambiguidade projetiva da reconstrução, visando tornar a análise qualitativa da reconstrução e a comparação com a reconstrução utilizando duas câmeras possíveis.

4.1 Tensor Trifocal

Uma maneira de se obter a relação entre as imagens de três câmeras é através da análise da projeção de uma reta no espaço. Na Figura 4.1, a reta \mathbf{L} no espaço é projetada nas três câmeras e tem como imagens as retas \mathbf{l} , \mathbf{l}' e \mathbf{l}'' nas câmeras, cujas matrizes de câmera são \mathbf{P} , \mathbf{P}' e \mathbf{P}'' , respectivamente.

Da análise da figura pode-se afirmar que a retroprojeção da reta em cada imagem

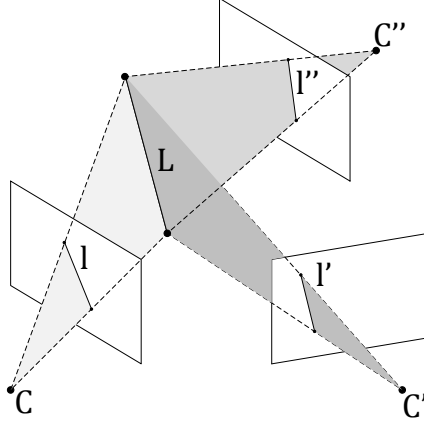


Figura 4.1: Relação entre as imagens projetadas de uma reta no espaço.

define um plano ao qual a reta \mathbf{L} obrigatoriamente pertence e que, portanto, a mesma representa a interseção dos três planos gerados pelas retroprojeções de suas três imagens. De forma inversa, todos os pontos do plano que contém a reta \mathbf{L} e o centro de uma câmera são projetados na reta que corresponde à imagem da reta \mathbf{L} .

Então, o ponto \mathbf{X} , que pertence a um plano π definido pela reta \mathbf{L} e o centro de uma câmera, tem como imagem $\mathbf{x} = \mathbf{P}\mathbf{X}$, onde \mathbf{P} é a matriz da câmera, e essa imagem pertence à reta \mathbf{l} , a imagem projetada de \mathbf{L} , e deve obedecer à equação $\mathbf{x}^T \mathbf{l} = 0$ (Equação (2.3)), ou $(\mathbf{P}\mathbf{X})^T \mathbf{l} = \mathbf{X}^T \mathbf{P}^T \mathbf{l} = 0$, de onde se conclui que $\mathbf{P}^T \mathbf{l}$ representa matematicamente o plano π .

Portanto, os pontos que definem os três planos na Figura 4.1 obedecem às relações $\mathbf{X}^T \mathbf{P}^T \mathbf{l} = 0$, $\mathbf{X}^T \mathbf{P}'^T \mathbf{l}' = 0$ e $\mathbf{X}^T \mathbf{P}''^T \mathbf{l}'' = 0$. E os pontos que pertencem à reta \mathbf{L} de interseção dos três planos obedecem ao sistema de equações:

$$\begin{bmatrix} \mathbf{l}^T \mathbf{P} \\ \mathbf{l}'^T \mathbf{P}' \\ \mathbf{l}''^T \mathbf{P}'' \end{bmatrix} \mathbf{X} = \begin{bmatrix} \pi \\ \pi' \\ \pi'' \end{bmatrix}^T \quad \mathbf{X} = \mathbf{M}^T \mathbf{X} = \mathbf{0}. \quad (4.1)$$

A matriz \mathbf{M} dessa equação deve ter posto igual a 2 para que a interseção entre os três planos seja uma reta, o que significa que uma coluna de \mathbf{M} pode ser escrita como uma combinação linear das outras duas.

Para três matrizes de câmera na forma $\mathbf{P} = [\mathbf{I} | \mathbf{0}]$, $\mathbf{P}' = [\mathbf{A} | \mathbf{a}_4]$ e $\mathbf{P}'' = [\mathbf{B} | \mathbf{b}_4]$, onde \mathbf{a}_4 e \mathbf{b}_4 são os epipolos relativos à primeira câmera na segunda e terceira câmera respectivamente, tem-se que:

$$\mathbf{M} = [\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3] = [\mathbf{P}^T \mathbf{l}, \mathbf{P}'^T \mathbf{l}', \mathbf{P}''^T \mathbf{l}''] = \begin{bmatrix} 1 & \mathbf{A}^T \mathbf{l}' & \mathbf{B}^T \mathbf{l}'' \\ 0 & \mathbf{a}_4^T \mathbf{l}' & \mathbf{b}_4^T \mathbf{l}'' \end{bmatrix}. \quad (4.2)$$

Analisando a quarta linha de \mathbf{M} e sabendo-se que, devido a seu posto igual a 2, a

primeira coluna pode ser escrita como uma combinação linear da segunda e da terceira, nota-se que a menos de um fator de escala, a combinação linear que relaciona as colunas de \mathbf{M} é

$$\mathbf{m}_1 = (\mathbf{b}_4^T \mathbf{l}'') \mathbf{m}_2 - (\mathbf{a}_4^T \mathbf{l}') \mathbf{m}_3. \quad (4.3)$$

Levando em consideração apenas as três primeiras linhas de \mathbf{M} , tem-se a equação

$$\mathbf{l} = (\mathbf{b}_4^T \mathbf{l}'') \mathbf{A}^T \mathbf{l}' - (\mathbf{a}_4^T \mathbf{l}') \mathbf{B}^T \mathbf{l}'' = (\mathbf{l}''^T \mathbf{b}_4) \mathbf{A}^T \mathbf{l}' - (\mathbf{l}'^T \mathbf{a}_4) \mathbf{B}^T \mathbf{l}'', \quad (4.4)$$

que, para o i -ésimo elemento de \mathbf{l} , pode ser reescrita como

$$\begin{aligned} l_i &= (\mathbf{l}''^T \mathbf{b}_4) \mathbf{a}_i^T \mathbf{l}' - (\mathbf{l}'^T \mathbf{a}_4) \mathbf{b}_i^T \mathbf{l}'' = \mathbf{l}''^T (\mathbf{b}_4 \mathbf{a}_i^T) \mathbf{l}' - \mathbf{l}'^T (\mathbf{a}_4 \mathbf{b}_i^T) \mathbf{l}'' \\ &= \mathbf{l}'^T (\mathbf{a}_i \mathbf{b}_4^T) \mathbf{l}'' - \mathbf{l}'^T (\mathbf{a}_4 \mathbf{b}_i^T) \mathbf{l}''. \end{aligned} \quad (4.5)$$

E definindo a matriz \mathbf{T}_i como

$$\mathbf{T}_i = (\mathbf{a}_i \mathbf{b}_4^T) - (\mathbf{a}_4 \mathbf{b}_i^T), \quad (4.6)$$

tem-se que a componente l_i da reta \mathbf{l} se relaciona com \mathbf{l}' e \mathbf{l}'' de acordo com a equação

$$l_i = \mathbf{l}'^T \mathbf{T}_i \mathbf{l}''. \quad (4.7)$$

Portanto, a reta \mathbf{l} pode ser escrita como

$$\mathbf{l} = \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} = \begin{bmatrix} \mathbf{l}'^T \mathbf{T}_1 \mathbf{l}'' \\ \mathbf{l}'^T \mathbf{T}_2 \mathbf{l}'' \\ \mathbf{l}'^T \mathbf{T}_3 \mathbf{l}'' \end{bmatrix}. \quad (4.8)$$

Ao conjunto $\mathbf{T} = \{\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3\}$, que relaciona as imagens das três câmeras, é dado o nome de tensor trifocal.

A Equação (4.8) estabelece a relação entre imagens de uma reta em três câmeras. É possível obter uma relação entre dois pontos aplicando-se a Equação (2.3). Inicialmente, reescreve-se a Equação (4.8) como $\mathbf{l} = \mathbf{H} \mathbf{l}''$, onde $\mathbf{H} = \left[(\mathbf{l}'^T \mathbf{T}_1)^T \quad (\mathbf{l}'^T \mathbf{T}_2)^T \quad (\mathbf{l}'^T \mathbf{T}_3)^T \right]^T$. De acordo com a Equação (2.3), $\mathbf{x}^T \mathbf{l} = \mathbf{x}^T \mathbf{H} \mathbf{l}'' = 0$ e, para que a relação $\mathbf{x}^T \mathbf{H} \mathbf{l}'' = 0$ seja verdadeira, $\mathbf{x}^T = \mathbf{x}''^T \mathbf{H}^{-1}$. Esta equação pode ser interpretada como uma transformação das coordenadas da imagem na primeira câmera de um ponto \mathbf{X} no espaço para as coordenadas da imagem do mesmo ponto na terceira, através de um plano definido pela imagem na segunda câmera de uma reta \mathbf{L} que passa pelo ponto \mathbf{X} , como visto na Figura 4.2. Transpondo ambos os lados dessa equação e multiplicando-os à esquerda por \mathbf{H}^T , obtém-se a relação entre

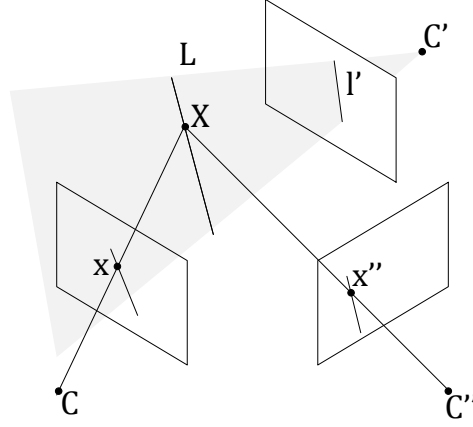


Figura 4.2: Relação entre duas imagens \mathbf{x} e \mathbf{x}'' de um ponto e a imagem \mathbf{l}' de uma reta que passa por ele.

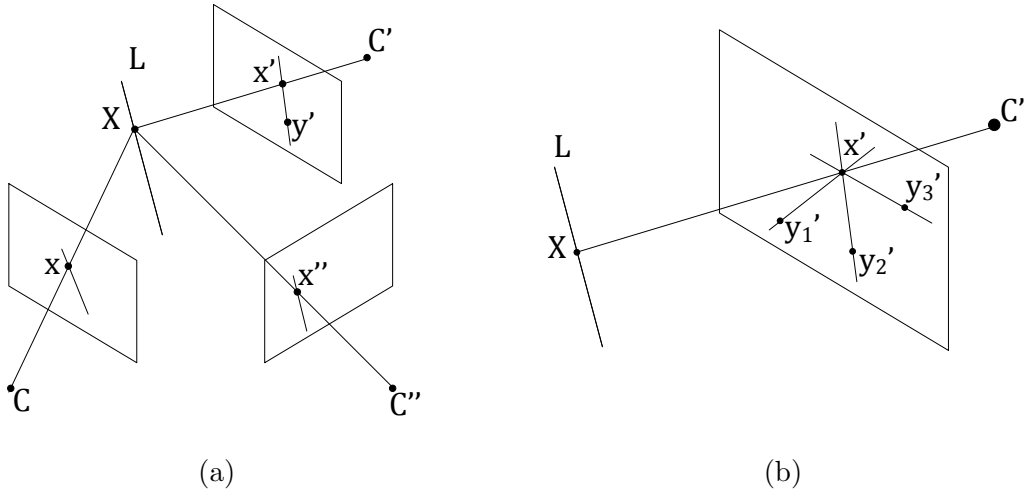


Figura 4.3: Relação entre três imagens de um ponto.

os pontos \mathbf{x} e \mathbf{x}'' :

$$\mathbf{x}'' = \mathbf{H}^T \mathbf{x} = \left[\mathbf{T}_1^T \mathbf{l}', \mathbf{T}_2^T \mathbf{l}', \mathbf{T}_3^T \mathbf{l}' \right] \mathbf{x} = \left(\sum_{i=1}^3 (x_i \mathbf{T}_i^T) \right) \mathbf{l}'. \quad (4.9)$$

Para eliminar o fator de escala [7], realiza-se o produto vetorial de ambos os lados da equação com \mathbf{x}'' . Ou, de forma equivalente, multiplica-se a transposta de ambos os lados à esquerda pela matriz $[\mathbf{x}'']_{\times}$, definida como a matriz antissimétrica que tem \mathbf{x}'' como vetor nulo, resultando em

$$\mathbf{x}''^T [\mathbf{x}'']_{\times} = \mathbf{l}'^T \left(\sum_{i=1}^3 (x_i \mathbf{T}_i) \right) [\mathbf{x}'']_{\times} = \mathbf{0}^T. \quad (4.10)$$

Por fim, para se obter uma relação entre três pontos, escreve-se a reta \mathbf{l}' como a junção do ponto \mathbf{x}' (a imagem na segunda câmera do ponto \mathbf{X} , como ilustrado na

Figura 4.3(a)) com um outro ponto \mathbf{y}' pertencente à reta \mathbf{l}' : $\mathbf{l}' = \mathbf{x}' \times \mathbf{y}'$, ou, usando a notação definida para a Equação (4.10), $\mathbf{l}' = [\mathbf{x}']_{\times} \mathbf{y}'$. Escrevendo \mathbf{l} nessa forma na Equação (4.10) tem-se, então:

$$\mathbf{l}^T \left(\sum_{i=1}^3 (x_i \mathbf{T}_i) \right) [\mathbf{x}'']_{\times} = \mathbf{y}'^T [\mathbf{x}']_{\times} \left(\sum_{i=1}^3 (x_i \mathbf{T}_i) \right) [\mathbf{x}'']_{\times} = \mathbf{0}^T. \quad (4.11)$$

Como essa equação é válida para qualquer reta \mathbf{l}' que passe por \mathbf{x} (Figura 4.3(b)) e conseqüentemente qualquer ponto \mathbf{y}' , tem-se que

$$[\mathbf{x}']_{\times} \left(\sum_{i=1}^3 (x_i \mathbf{T}_i) \right) [\mathbf{x}'']_{\times} = \mathbf{0}_{3 \times 3}, \quad (4.12)$$

o que define uma relação entre três pontos.

4.2 Calculando o Tensor Trifocal

Assim como a matriz fundamental pode ser calculada a partir da correspondência das imagens de pontos de referência em duas câmeras, o tensor trifocal também pode ser calculado usando as imagens de pontos de referência em três câmeras.

O tensor trifocal é composto por três matrizes de 9 elementos cada, tendo então 27 elementos. Portanto, com 27 ou mais equações é possível obter uma solução pelo método dos mínimos quadrados para a equação $\mathbf{A}\mathbf{t}$, onde \mathbf{t} é o vetor formado pelos elementos do tensor trifocal. A Equação (4.12), que relaciona pontos em três imagens, fornece um conjunto de 9 equações com os elementos do tensor trifocal como variáveis, mas segundo [7] apenas 4 são linearmente independentes. Então, para obter 27 ou mais equações a partir de correspondência de pontos são necessários ao menos 7 pontos de referência.

Montada a equação, a resolução pode ser feita a partir da decomposição em valores singulares $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$. O vetor \mathbf{t} que minimiza a norma $\|\mathbf{A}\mathbf{t}\|$ é a coluna de \mathbf{V} correspondente ao menor valor singular. Se os elementos de \mathbf{D} forem ordenados de forma decrescente, o vetor solução escolhido será então $\mathbf{t} = \mathbf{v}_{27}$, a última coluna de \mathbf{V} .

Contudo, o tensor trifocal obtido dessa maneira não será geometricamente válido [7]. Assim como para a matriz fundamental era necessário forçar a singularidade alterando a matriz fundamental obtida inicialmente, para o tensor trifocal ser geometricamente válido é necessário que existam três matrizes de câmera $\mathbf{P} = [\mathbf{I}|\mathbf{0}]$, $\mathbf{P}' = [\mathbf{A}|\mathbf{a}_4]$, $\mathbf{P}'' = [\mathbf{B}|\mathbf{b}_4]$ que estejam de acordo com a Equação (4.6): $\mathbf{T}_i = (\mathbf{a}_i \mathbf{b}_4^T) - (\mathbf{a}_4 \mathbf{b}_i^T)$.

4.2.1 Corrigindo as Características Geométricas do Tensor Trifocal

Para calcular o tensor trifocal geometricamente válido, inicialmente calcula-se uma estimativa inicial do tensor $\mathbf{T} = [\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3]$ pelo método da decomposição em valores singulares. A partir dessa estimativa, recupera-se os epipolos \mathbf{e}' e \mathbf{e}'' .

Lembrando que $\mathbf{e}' = \mathbf{a}_4$ e $\mathbf{e}'' = \mathbf{b}_4$, tem-se que $\mathbf{T}_i = (\mathbf{a}_i \mathbf{e}''^T) - (\mathbf{e}' \mathbf{b}_i^T)$. O vetor nulo à esquerda de \mathbf{T}_i é $\mathbf{v}_i^e = \mathbf{a}_i \times \mathbf{e}'$, pois $\mathbf{v}_i^{eT} \mathbf{T}_i = (\mathbf{v}_i^{eT} \mathbf{a}_i) \mathbf{e}''^T - (\mathbf{v}_i^{eT} \mathbf{e}') \mathbf{b}_i^T = \mathbf{0}^T$ e o vetor nulo à direita de \mathbf{T}_i é $\mathbf{v}_i^d = \mathbf{b}_i \times \mathbf{e}''$, pois $\mathbf{T}_i \mathbf{v}_i^d = \mathbf{a}_i (\mathbf{e}''^T \mathbf{v}_i^d) - \mathbf{e}' (\mathbf{b}_i^T \mathbf{v}_i^d) = \mathbf{0}$. Percebe-se também que \mathbf{e}' é perpendicular a \mathbf{v}_i^e e que \mathbf{e}'' é perpendicular a \mathbf{v}_i^d .

Então, para recuperar os epipolos, deve-se minimizar as normas $\|\mathbf{T}_i^T \mathbf{v}_i^e\|$ e $\|\mathbf{T}_i \mathbf{v}_i^d\|$ para $i = 1, 2$ e 3 , para encontrar os vetores nulos à esquerda e à direita e depois minimizar as normas $\|[\mathbf{v}_1^e, \mathbf{v}_2^e, \mathbf{v}_3^e]^T \mathbf{e}'\|$ e $\|[\mathbf{v}_1^d, \mathbf{v}_2^d, \mathbf{v}_3^d]^T \mathbf{e}''\|$ para recuperar os epipolos.

Tendo os epipolos calculados, minimiza-se $\|\mathbf{A}\mathbf{t}\| = \|\mathbf{A}\mathbf{O}\mathbf{a}\|$ com $\|\mathbf{t}\| = \|\mathbf{O}\mathbf{a}\| = 1$, onde \mathbf{a} é a matriz formada pelos elementos dos vetores $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, \mathbf{b}_1, \mathbf{b}_2$ e \mathbf{b}_3 e \mathbf{O} é a matriz composta a partir dos epipolos \mathbf{e}' e \mathbf{e}'' que relaciona \mathbf{t} e \mathbf{a} de forma equivalente às equações $\mathbf{T}_i = (\mathbf{a}_i \mathbf{e}''^T) - (\mathbf{e}' \mathbf{b}_i^T)$ para $i = 1, 2, 3$.

O algoritmo para minimizar $\|\mathbf{t}\| = \|\mathbf{O}\mathbf{a}\|$ sujeito a $\|\mathbf{O}\mathbf{a}\| = 1$ tem seu desenvolvimento explicado em [7] (p. 594, Seção A5.4.1). Inicia-se o algoritmo calculando a decomposição em valor singulares $\mathbf{O} = \mathbf{U}\mathbf{D}\mathbf{V}^T$, com os elementos de \mathbf{D} ordenados de forma que os elementos diferentes de zero estejam nas linhas superiores. Então, define-se a matriz \mathbf{U}' formada pelas primeiras r colunas de \mathbf{U} , onde r é o posto da matriz \mathbf{O} . Depois, deve-se encontrar o vetor \mathbf{t}' que minimiza $\|\mathbf{A}\mathbf{U}'\mathbf{t}'\|$, utilizando o mesmo método de minimização utilizado para a primeira estimativa do tensor trifocal. E o vetor \mathbf{t} que soluciona o problema inicial é então dado por $\mathbf{t} = \mathbf{U}'\mathbf{t}'$.

Reordenando os elementos de $\mathbf{t} = \mathbf{O}\mathbf{a}$ obtido por esse algoritmo, obtém-se então o tensor trifocal \mathbf{T} geometricamente válido.

4.2.2 Normalização

Para o cálculo do tensor trifocal, também se deve normalizar as coordenadas dos pontos antes do início dos algoritmos de minimização para evitar problemas de condicionamento da matriz \mathbf{A} . Sendo \mathbf{H}, \mathbf{H}' e \mathbf{H}'' as matrizes de normalização das coordenadas dos pontos nas imagens da primeira, segunda e terceira câmeras, respectivamente, obtidas da mesma maneira que as matrizes de normalização na Seção 3.2.3, após a obtenção do tensor trifocal para as coordenadas normalizadas, deve-se utilizar as matrizes de normalização para calcular o tensor trifocal para as coordenadas originais. Não é possível escrever de forma simples com apenas uma equação (utilizando as notações tradicionais de multiplicação de matrizes) a operação de desnormalização do tensor trifocal como um todo ou para uma de suas

componentes \mathbf{T}_i , mas é possível [7] se obter a seguinte equação para cada elemento $T_i(j, k)$ da j -ésima linha e k -ésima coluna da matriz \mathbf{T}_i :

$$T_i(j, k) = \mathbf{h}_i^T \begin{bmatrix} \mathbf{g}_k'' \hat{\mathbf{T}}_1^T \\ \mathbf{g}_k'' \hat{\mathbf{T}}_2^T \\ \mathbf{g}_k'' \hat{\mathbf{T}}_3^T \end{bmatrix} \mathbf{g}_j'^T, \quad (4.13)$$

onde $\hat{\mathbf{T}}_i$ é uma das matrizes que formam o tensor trifocal para coordenadas normalizadas, \mathbf{h}_i é a i -ésima coluna da matriz \mathbf{H} , \mathbf{g}_k'' é a k -ésima linha da matriz $\mathbf{G}'' = \mathbf{H}''^{-1}$ e \mathbf{g}_j' é a j -ésima linha da matriz $\mathbf{G}' = \mathbf{H}'^{-1}$.

4.3 Extrair as Matrizes Fundamentais do Tensor Trifocal

Conhecido o tensor trifocal, é possível, a partir dele, obter matrizes fundamentais. Como visto no Capítulo 3, todas as retas epipolares relativas a um epipolo \mathbf{e}'' têm como interseção o próprio epipolo e podem ser, portanto, escritas como a junção do epipolo com outro ponto qualquer pertencente a elas: $\mathbf{l}'' = \mathbf{e}'' \times \mathbf{x}'' = [\mathbf{e}'']_{\times} \mathbf{x}''$. Escrevendo \mathbf{x}'' de acordo com a Equação (4.9) como $\mathbf{x}'' = [\mathbf{T}_1^T \mathbf{l}', \mathbf{T}_2^T \mathbf{l}', \mathbf{T}_3^T \mathbf{l}'] \mathbf{x}$, a reta epipolar pode ser reescrita como

$$\mathbf{l}'' = \mathbf{e}'' \times \mathbf{x}'' = [\mathbf{e}'']_{\times} [\mathbf{T}_1^T \mathbf{l}', \mathbf{T}_2^T \mathbf{l}', \mathbf{T}_3^T \mathbf{l}'] \mathbf{x}. \quad (4.14)$$

Escrevendo a Equação (3.2) para a reta \mathbf{l}'' , $\mathbf{l}'' = \mathbf{F}_{31} \mathbf{x}$, e comparando-a com a Equação (4.14), percebe-se que \mathbf{F}_{31} , a matriz fundamental que relaciona as imagens da primeira e da terceira câmera, pode ser calculada através da equação

$$\mathbf{F}_{31} = [\mathbf{e}'']_{\times} [\mathbf{T}_1^T \mathbf{l}', \mathbf{T}_2^T \mathbf{l}', \mathbf{T}_3^T \mathbf{l}']. \quad (4.15)$$

Essa equação é válida para qualquer vetor \mathbf{l}' , desde que este não faça parte do espaço nulo à direita das matrizes \mathbf{T}_i^T [7]. Da Equação 4.6, $\mathbf{T}_i = (\mathbf{a}_i \mathbf{b}_4^T) - (\mathbf{a}_4 \mathbf{b}_i^T)$, fica claro que o vetor $\mathbf{a}_4 \times \mathbf{a}_i = \mathbf{e}' \times \mathbf{a}_i$ é o vetor nulo à esquerda da matriz \mathbf{T}_i , e à direita da matriz \mathbf{T}_i^T , pois é perpendicular a \mathbf{a}_i e \mathbf{e}' . Então, como \mathbf{e}' é perpendicular ao vetor nulo à direita de \mathbf{T}_i^T , ele é uma boa escolha para \mathbf{l}' . Para essa escolha de \mathbf{l}' , a Equação (4.15) pode ser reescrita, então, como

$$\mathbf{F}_{31} = [\mathbf{e}'']_{\times} [\mathbf{T}_1^T \mathbf{e}', \mathbf{T}_2^T \mathbf{e}', \mathbf{T}_3^T \mathbf{e}']. \quad (4.16)$$

Um desenvolvimento semelhante pode ser feito para a matriz fundamental \mathbf{F}_{21} , que relaciona as imagens da primeira câmera e da segunda câmera, resultando na

equação

$$\mathbf{F}_{21} = [\mathbf{e}']_{\times} \begin{bmatrix} \mathbf{T}_1 \mathbf{e}'' & \mathbf{T}_2 \mathbf{e}'' & \mathbf{T}_3 \mathbf{e}'' \end{bmatrix}. \quad (4.17)$$

O procedimento para obtenção dessa matriz deve começar com a definição da reta epipolar \mathbf{l}' que passa pelo epipolo \mathbf{e}' como $\mathbf{l}' = [\mathbf{e}']_{\times} \mathbf{x}'$. Transpondo os elementos do lado direito da Equação (4.8), pode-se obter a equação do ponto \mathbf{x}' na forma $\mathbf{x}' = \mathbf{H}^T \mathbf{x} = \begin{bmatrix} \mathbf{T}_1 \mathbf{l}'' & \mathbf{T}_2 \mathbf{l}'' & \mathbf{T}_3 \mathbf{l}'' \end{bmatrix} \mathbf{x}$. Substitui-se, então, essa equação na da reta \mathbf{l}' e a continuação do desenvolvimento é idêntica ao desenvolvimento anterior.

4.4 Obtendo as Matrizes das Câmeras

Como visto na Seção 3.2.1, dadas duas matrizes de câmera, \mathbf{P} e \mathbf{P}' , a matriz fundamental \mathbf{F} que relaciona as duas câmeras é definida como $\mathbf{F} = [\mathbf{e}']_{\times} \mathbf{P}' \mathbf{P}^+$.

Se a matriz \mathbf{P} é escolhida como $\mathbf{P} = [\mathbf{I} | \mathbf{0}]$, sua pseudo-inversa \mathbf{P}^+ será igual a $\begin{bmatrix} \mathbf{I}_{3 \times 3} \\ \mathbf{0} \end{bmatrix}$ e se a matriz \mathbf{P}' é da forma $\mathbf{P}' = [\mathbf{M} | \mathbf{m}]$, o epipolo \mathbf{e}' é dado por $\mathbf{e}' = \mathbf{m}$. Nesse caso, a matriz fundamental pode ser escrita como

$$\mathbf{F} = [\mathbf{m}]_{\times} \begin{bmatrix} \mathbf{M} & \mathbf{m} \end{bmatrix} = [\mathbf{m}]_{\times} \mathbf{M}. \quad (4.18)$$

Tendo então a matriz $\mathbf{F}_{21} = [\mathbf{e}']_{\times} \begin{bmatrix} \mathbf{T}_1 \mathbf{e}'' & \mathbf{T}_2 \mathbf{e}'' & \mathbf{T}_3 \mathbf{e}'' \end{bmatrix}$ e escolhendo $\mathbf{P} = [\mathbf{I} | \mathbf{0}]$, de acordo com a Equação (4.18) a matriz \mathbf{P}' pode ser escolhida como

$$\mathbf{P}' = \begin{bmatrix} \mathbf{T}_1 \mathbf{e}'' & \mathbf{T}_2 \mathbf{e}'' & \mathbf{T}_3 \mathbf{e}'' & | & \mathbf{e}' \end{bmatrix}. \quad (4.19)$$

Para se obter a matriz \mathbf{P}'' , não é correto utilizar o mesmo procedimento em relação à matriz \mathbf{F}_{31} . Fazendo isso, a matriz da primeira câmera formaria dois pares válidos com a da terceira e a da segunda câmera, respeitando suas respectivas matrizes fundamentais, mas o conjunto das três matrizes de câmera não seria consistente com o tensor trifocal.

Então, para obter uma matriz válida para a terceira câmera, utiliza-se a Equação (4.6), fazendo $\mathbf{a}_i = \mathbf{T}_i \mathbf{e}''$, em concordância com a Equação (4.19):

$$\mathbf{T}_i = (\mathbf{T}_i \mathbf{e}'' \mathbf{e}''^T) - (\mathbf{e}' \mathbf{b}_i^T) \quad (4.20)$$

Então manipulando a Equação (4.20) de forma que em seu lado esquerdo fique

apenas o termo dependente de \mathbf{b}_i , pode-se escrever

$$\mathbf{e}'\mathbf{b}_i^T = (\mathbf{T}_i\mathbf{e}''\mathbf{e}''^T) - \mathbf{T}_i = \mathbf{T}_i(\mathbf{e}''\mathbf{e}''^T - \mathbf{I}). \quad (4.21)$$

Como \mathbf{e}' é definido a menos de um fator de escala, pode-se escolher seu módulo como sendo igual a 1 e, portanto $\mathbf{e}'^T\mathbf{e}' = 1$. Multiplicando-se ambos os lados da equação à esquerda por \mathbf{e}'^T e transpondo o resultado, chega-se à equação

$$\mathbf{b}_i^T = (\mathbf{e}''\mathbf{e}''^T - \mathbf{I})\mathbf{T}_i^T\mathbf{e}'. \quad (4.22)$$

Então, como $\mathbf{P}'' = \left[\begin{array}{c|c} [\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3] & \mathbf{e}'' \end{array} \right]$, tem-se

$$\mathbf{P}'' = \left[(\mathbf{e}''\mathbf{e}''^T - \mathbf{I}) \left[\mathbf{T}_1^T\mathbf{e}', \mathbf{T}_2^T\mathbf{e}', \mathbf{T}_3^T\mathbf{e}' \right] \mid \mathbf{e}'' \right]. \quad (4.23)$$

A matriz \mathbf{P}'' obtida dessa maneira, juntamente com \mathbf{P} e \mathbf{P}' previamente definidos, é consistente com a matriz fundamental \mathbf{F}_{31} e com o tensor trifocal.

4.5 Algoritmo

Esta seção resume o algoritmo utilizado para a obtenção das matrizes \mathbf{P} , \mathbf{P}' e \mathbf{P}'' de três câmeras a partir do tensor trifocal. O código desenvolvido a partir desse algoritmo para o aplicativo deste trabalho é apresentado no Apêndice A.3.

Entradas: N conjuntos de pontos $(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}''_i)$ correspondentes às imagens nas três câmeras de N pontos \mathbf{X}_i no espaço.

1. Normalizar os vetores \mathbf{x}_i , \mathbf{x}'_i e \mathbf{x}''_i de forma que o centroide para cada conjunto de vetores seja a origem do sistema de coordenadas e o comprimento médio dos vetores seja igual a $\sqrt{2}$, obtendo as matrizes de normalização \mathbf{H} , \mathbf{H}' e \mathbf{H}'' conforme visto na Seção 3.2.3.
2. Calcular uma estimativa inicial $\hat{\mathbf{T}}$ do tensor trifocal minimizando $\|\mathbf{A}\mathbf{t}\|$.
 - (a) Formar a matriz \mathbf{A} a partir da Equação (4.12), utilizando os N conjuntos de pontos.
 - (b) Realizar a decomposição em valores singulares $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$.
 - (c) Fazer $\mathbf{t} = \mathbf{v}_N$, onde \mathbf{v}_N é a última coluna de \mathbf{V} , e ordenar os elementos do vetor \mathbf{t} formando as matrizes $\hat{\mathbf{T}}_1$, $\hat{\mathbf{T}}_2$ e $\hat{\mathbf{T}}_3$ que compõem o tensor trifocal $\hat{\mathbf{T}}$.
3. Calcular o tensor trifocal $\hat{\mathbf{T}}'$ geometricamente válido.

- (a) Minimizar as normas $\|\hat{\mathbf{T}}_i^T \mathbf{v}_i^e\|$ e $\|\hat{\mathbf{T}}_i^T \mathbf{v}_i^d\|$ para $i = 1, 2$ e 3 , para encontrar os vetores nulos à esquerda, \mathbf{v}_i^e , e à direita, \mathbf{v}_i^d , das matrizes $\hat{\mathbf{T}}_i$.
- (b) Recuperar os epipolos minimizando as normas $\|[\mathbf{v}_1^e, \mathbf{v}_2^e, \mathbf{v}_3^e]^T \mathbf{e}'\|$ e $\|[\mathbf{v}_1^d, \mathbf{v}_2^d, \mathbf{v}_3^d]^T \mathbf{e}''\|$.
- (c) Montar a matriz \mathbf{O} , composta a partir dos epipolos \mathbf{e}' e \mathbf{e}'' , que relaciona \mathbf{t} e o vetor $\mathbf{a} = [\mathbf{a}_1^T \ \mathbf{a}_2^T \ \mathbf{a}_3^T \ \mathbf{b}_1^T \ \mathbf{b}_2^T \ \mathbf{b}_3^T]^T$ de forma equivalente à equação $\hat{\mathbf{T}}_i^T = (\mathbf{a}_i \mathbf{e}'^T) - (\mathbf{e}'' \mathbf{b}_i^T)$ para $i = 1, 2, 3$.
- (d) Minimizar $\|\mathbf{A}\mathbf{t}\| = \|\mathbf{A}\mathbf{O}\mathbf{a}\|$ com $\|\mathbf{t}\| = \|\mathbf{O}\mathbf{a}\| = 1$.
- i. Calcular a decomposição em valor singulares $\mathbf{O} = \mathbf{U}\mathbf{D}\mathbf{V}^T$, com os elementos de \mathbf{D} ordenados de forma que os elementos diferentes de zero estejam nas linhas superiores.
 - ii. Definir a matriz \mathbf{U}' formada pelas primeiras r colunas de \mathbf{U} , onde r é o posto da matriz \mathbf{O} .
 - iii. Encontrar o vetor \mathbf{t}' que minimiza $\|\mathbf{A}\mathbf{U}'\mathbf{t}'\|$, utilizando o mesmo método de minimização utilizado para a primeira estimativa do tensor trifocal.
 - iv. Obter \mathbf{t} fazendo $\mathbf{t} = \mathbf{U}'\mathbf{t}'$ e reordenar seus elementos para formar tensor trifocal $\hat{\mathbf{T}}$ geometricamente válido.
4. Encontrar \mathbf{T} para as coordenadas originais, utilizando a Equação (4.13).
- (a) Calcular as inversas das matrizes de normalização $\mathbf{G}' = \mathbf{H}'^{-1}$ e $\mathbf{G}'' = \mathbf{H}''^{-1}$.
- (b) Calcular as componentes $T_i(j, k) = \mathbf{h}_i^T \begin{bmatrix} \mathbf{g}_k'' \hat{\mathbf{T}}_1^T \\ \mathbf{g}_k'' \hat{\mathbf{T}}_2^T \\ \mathbf{g}_k'' \hat{\mathbf{T}}_3^T \end{bmatrix} \mathbf{g}_j'^T$ do tensor.
5. Obter as matrizes fundamentais $\mathbf{F}_{21} = [\mathbf{e}']_{\times} [\mathbf{T}_1 \mathbf{e}'', \mathbf{T}_2 \mathbf{e}'', \mathbf{T}_3 \mathbf{e}'']$ e $\mathbf{F}_{31} = [\mathbf{e}'']_{\times} [\mathbf{T}_1^T \mathbf{e}', \mathbf{T}_2^T \mathbf{e}', \mathbf{T}_3^T \mathbf{e}']$.
6. Fazer $\mathbf{P} = [\mathbf{I} | \mathbf{0}]$ e calcular as duas outras matrizes de câmera através das Equações (4.19) e (4.23): $\mathbf{P}' = [[\mathbf{T}_1 \mathbf{e}'', \mathbf{T}_2 \mathbf{e}'', \mathbf{T}_3 \mathbf{e}''] \ | \ \mathbf{e}']$ e $\mathbf{P}'' = [(\mathbf{e}'' \mathbf{e}''^T - \mathbf{I}) [\mathbf{T}_1^T \mathbf{e}', \mathbf{T}_2^T \mathbf{e}', \mathbf{T}_3^T \mathbf{e}'] \ | \ \mathbf{e}'']$.
7. Obter a reconstrução com distorção projetiva. A partir de \mathbf{P} , \mathbf{P}' e \mathbf{P}'' , calcular as coordenadas 3D dos pontos a serem reconstruídos resolvendo a Equação (2.8) para três câmeras.

8. Para a obtenção das matrizes das câmeras sem ambiguidade projetiva, realizar o mesmo procedimento adotado no algoritmo da Seção 3.5 para o cálculo das matrizes essenciais e das matrizes das câmeras.

Capítulo 5

Sistema Completo

Este capítulo apresenta o aplicativo desenvolvido neste trabalho e detalha suas funcionalidades. A Seção 5.1 exibe uma descrição do sistema, detalhando o *hardware* controlado pelo aplicativo e a interface gráfica do programa. As rotinas do aplicativo, que incluem todas as etapas desde a captura das imagens até a reconstrução das cenas, são explicadas na Seção 5.2.

5.1 Visão Geral do Sistema

O *hardware* utilizado neste trabalho é composto por três *webcams* com resolução *full HD* (1080 x 1920 pixels) conectadas por interface USB a um computador, como mostra o diagrama da Figura 5.1.

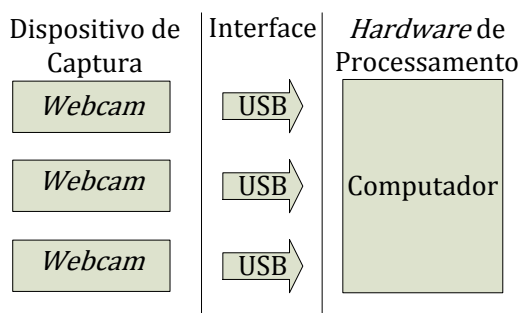


Figura 5.1: Diagrama do *hardware* do trabalho.

O programa executado no computador, que tem sua tela principal exibida na Figura 5.2, foi programado para o aplicativo MATLAB. A interface do programa é formada por cinco botões que iniciam a execução das principais rotinas e uma área para a exibição das imagens.

Durante a execução das rotinas que demandam interação com o usuário, essa área exibe as imagens de cada uma das três câmeras, de acordo com os algoritmos executados. Os procedimentos de correspondência de pontos para os cálculos da

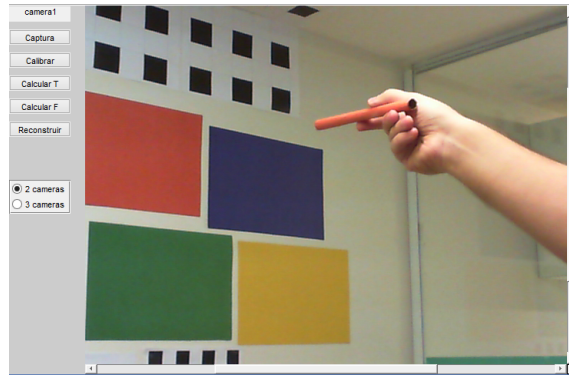


Figura 5.2: Interface do aplicativo.

matriz fundamental e do tensor trifocal são realizados manualmente pelo usuário, com uma sequência de cliques em pontos de referência nos *grids* presentes na cena, como pode ser visto na Seção 5.2. A interface deve então apresentar as imagens das duas câmeras utilizadas para o cálculo da matriz fundamental ou das três imagens no caso do tensor trifocal, alternadamente, a cada interação do usuário. O processo de reconstrução, dadas as imagens e já calculadas as matrizes das câmeras por meio dos métodos de duas e três câmeras, é automático e a Figura 5.3 mostra um exemplo do resultado que é exibido em uma janela separada da interface do aplicativo.

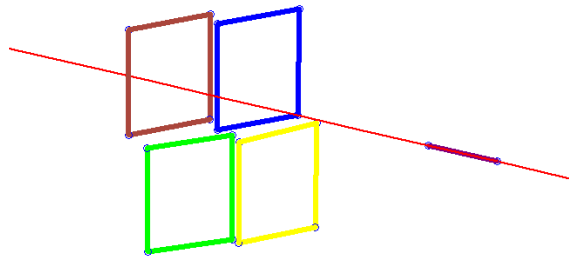


Figura 5.3: Resultado de uma reconstrução 3D.

5.2 Rotinas do Programa

Nesta seção as principais rotinas do *software* são melhor detalhadas. De forma geral, o programa tem como entradas as imagens capturadas das cenas e a saída, após todo o processamento necessário sobre as imagens de entrada para a obtenção das informações sobre o conjunto de câmeras, deve ser a reconstrução 3D das cenas, como ilustrado no diagrama da Figura 5.4.

Então o aplicativo apresenta uma rotina para a aquisição das imagens de entrada explicada na Seção 5.2.1, três rotinas para o cálculo das matrizes de câmera: uma

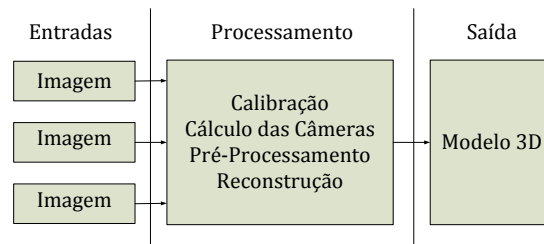


Figura 5.4: Diagrama de entradas e saídas do aplicativo.

para calibração (Seção 5.2.2) e duas para obtenção das matrizes de câmera (uma através do cálculo da matriz fundamental e outra do cálculo do tensor trifocal), baseadas em um procedimento de correspondência de pontos (Seção 5.2.3), e uma última rotina para a reconstrução, detalhada na Seção 5.2.4. A Figura 5.5 relaciona os botões da interface do aplicativo com as rotinas que eles iniciam.

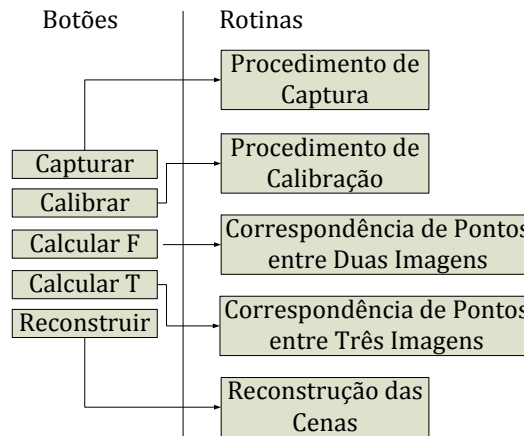


Figura 5.5: Rotinas do programa.

5.2.1 Captura de Imagens

A captura de imagem é procedimento responsável pela aquisição dos dados que são utilizados como entrada nas demais rotinas do trabalho. Todas as outras etapas da reconstrução dependem da existência de pelo menos um conjunto de imagens para ser processado. Os conjuntos de imagens gerados são todos automaticamente armazenados e nomeados sequencialmente, para serem utilizados na rotina de reconstrução. O primeiro conjunto de imagens é também utilizado nos processos de calibração e obtenção das matrizes das câmeras.

A Figura 5.6 ilustra o procedimento de captura adotado neste trabalho. Para facilitar a obtenção de um conjunto grande de imagens sem a necessidade de se utilizar o botão na interface gráfica diversas vezes, uma variável no código do programa

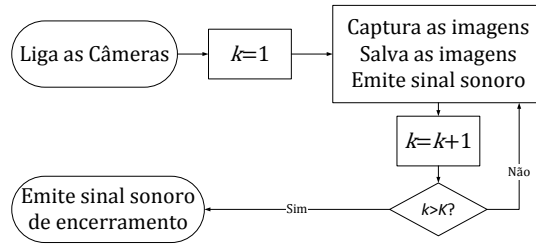


Figura 5.6: Procedimento de captura de imagens.

(representada pela constante K na figura) permite escolher quantas fotografias consecutivas serão capturadas. Quando o usuário inicia a captura, o programa aguarda alguns segundos para que ele se posicione na cena apontando para o objeto a ser selecionado. Obtido o conjunto de fotografias, é emitido um sinal sonoro para comunicar que o usuário deve apontar para o próximo objeto escolhido e esse processo se repete até que o número de conjuntos determinado por aquela variável seja atingido, quando é emitido outro sinal sonoro que indica o fim da captura.

5.2.2 Calibração

Neste trabalho a calibração das câmeras foi realizada com auxílio de um *grid* presente na cena. A rotina tem como entrada um conjunto de duas ou três imagens e retorna como saída as matrizes de calibração das câmeras referentes às imagens utilizadas.

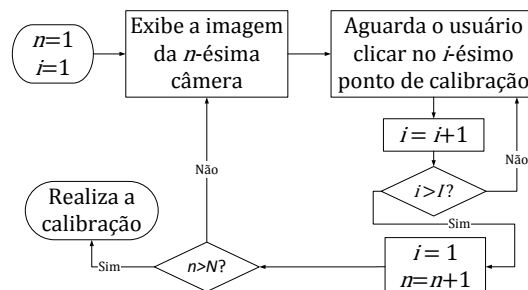


Figura 5.7: Rotina de marcação de pontos de referência para calibração.

Esse processo depende de um procedimento manual de seleção de pontos, descrito no diagrama da Figura 5.7 para um conjunto de N câmeras, onde em um *grid* tridimensional são selecionados, para cada câmera n , um número I de pontos (18 pontos neste trabalho) suficientes para realizar o cálculo de suas matrizes de calibração. Após o término da rotina, essas matrizes são salvas para que não seja necessária a repetição deste etapa do trabalho em uma nova execução do programa.

5.2.3 Correspondência de Pontos

Como visto nos Capítulos 3 e 4, as matrizes fundamentais e o sensor trifocal podem ser obtidos a partir de uma correspondência de pontos entre imagens capturadas pelas câmeras. Neste trabalho, essa tarefa, assim como a calibração, é realizada manualmente.

Quando o usuário inicia a rotina de obtenção da matriz fundamental, ou do tensor trifocal, o programa inicialmente exibe a imagem da primeira câmera. O usuário, então, escolhe um ponto em um dos *grids* que seja visível na imagem de todas as câmeras e clica nesse ponto. A seguir, a imagem de outra câmera é exibida e o usuário clica na imagem do ponto correspondente ao selecionado na imagem anterior. Quando o primeiro ponto de referência é indicado na imagem de todas as câmeras (duas ou três neste trabalho), o programa volta a exibir a imagem da primeira câmera e o usuário repete o mesmo procedimento para outro ponto de referência. A tarefa é repetida até que o número de pontos de referência determinado no código seja alcançado. Todo o processo é resumido no diagrama da Figura 5.8 que ilustra a rotina para N câmeras e I pontos de referência. Neste trabalho foram utilizados 32 pontos de referência tanto para a correspondência entre duas quanto para a entre três câmeras.

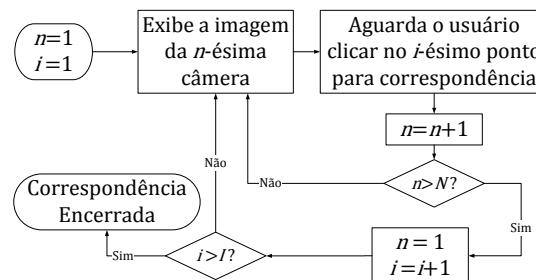


Figura 5.8: Processo de correspondência de pontos entre imagens.

5.2.4 Reconstrução

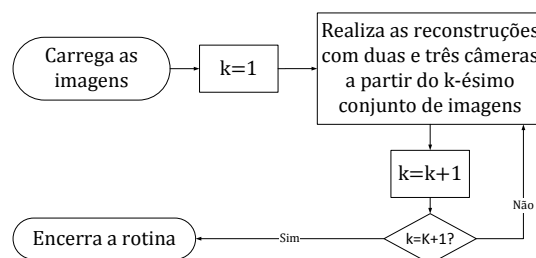


Figura 5.9: Rotina de reconstrução 3D das cenas.

Quando há pelo menos um conjunto de imagens disponível e já foram realizadas a calibração e a obtenção da matriz fundamental e do tensor trifocal, a reconstrução 3D pode ser realizada. A função principal utiliza como entradas as matrizes das câmeras sem ambiguidade projetiva obtidas através dos algoritmos apresentados nas Seções 3.5 e 4.5 e as imagens das cenas a serem reconstruídas.

Como explicado na Figura 5.9, que apresenta o diagrama para a reconstrução de um número K de cenas, a rotina carrega um conjunto de imagens por vez e, usando as matrizes das câmeras, realiza e exibe duas reconstruções da cena, uma utilizando duas imagens e outra três. Após a utilização de todos os conjuntos, a rotina é encerrada e os resultados podem então ser comparados entre si e com o resultado esperado para cada conjunto.

Capítulo 6

Resultados

Para a obtenção dos resultados foram utilizados para compor a cena quatro objetos representados por retângulos de quatro diferentes cores fixados em uma parede e um apontador como descrito na Seção 2.1. Três câmeras do tipo *webcam* com resolução *full* HD fizeram a aquisição das imagens.

As câmeras foram acopladas a tripés e posicionadas a aproximadamente dois metros dos objetos. A configuração dos tripés foi planejada para que o usuário da sala não ficasse posicionado entre as câmeras e os objetos, de forma que a reconstrução não fosse prejudicada por possíveis oclusões de objetos em nenhum dos conjuntos de imagens, como ilustrado na Figura 6.1, onde as câmeras são representadas por três triângulos, o segmento de reta representa a área da parede ocupada pelos objetos e o círculo, o usuário da sala. A figura mostra que, com essa configuração da posição das câmeras, todos os objetos e o usuário da sala são visíveis nas imagens da primeira câmera (Figura 6.1(a)), da segunda câmera (Figura 6.1(b)) e da terceira câmera (Figura 6.1(c)).

As imagens e as matrizes de câmera utilizadas foram obtidas de acordo com as rotinas descritas no Capítulo 5. Não foi realizado nenhum processamento nas imagens além dos descritos neste trabalho. Por exemplo, correções de distorção radial das lentes baseadas em ajuste de polinômios de grau par [4] não foram utilizadas.

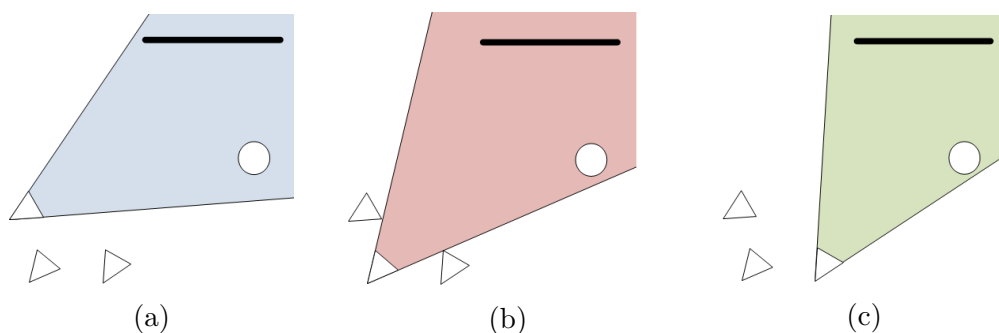


Figura 6.1: Configuração das câmeras.

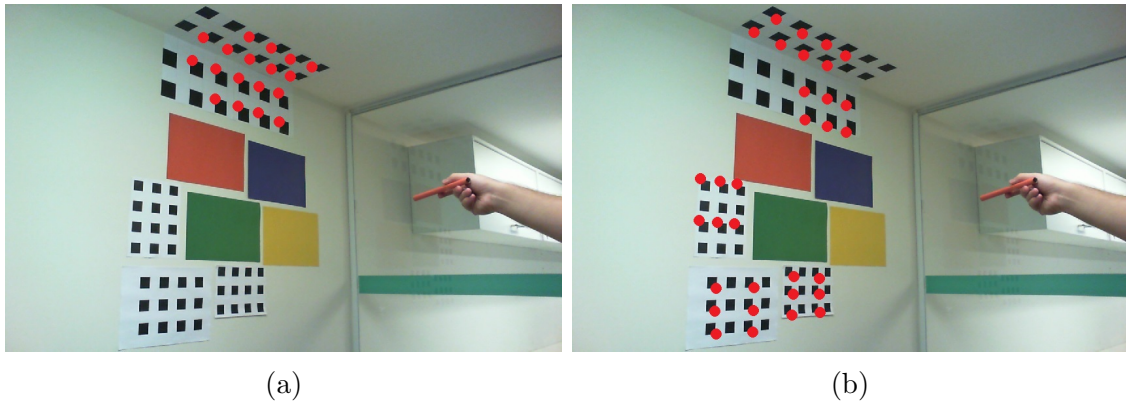


Figura 6.2: Pontos utilizados para calibração (a) e correspondência de pontos (b).

Os pontos de referência utilizados para os processos de calibração e correspondência de pontos estão destacados nas Figuras 6.2(a) e 6.2(b), respectivamente.

O procedimento adotado para a obtenção dos resultados consiste em carregar cada conjunto de imagens, nos quais o usuário da sala está apontando para um dos objetos, realizar reconstruções da cena utilizando as imagens de duas e três câmeras, verificar, em cada reconstrução, se o objeto pelo qual o prolongamento da imagem reconstruída do apontador passa corresponde ao objeto verdadeiramente apontado e analisar qualitativamente os resultados. Para que distorções projetivas não interferissem na aferição dos resultados, foram utilizadas as matrizes das câmeras sem ambiguidade projetiva na análise da qualidade das reconstruções.

As Tabelas 6.1 e 6.2 exibem os conjuntos de imagens utilizados e as Tabelas 6.3 e 6.4 apresentam as reconstruções obtidas para os oito conjuntos de imagens. Nas tabelas que apresentam os resultados da reconstrução, a primeira coluna mostra o número que identifica o conjunto de imagens utilizado, de acordo com as Tabelas 6.1 e 6.2, a segunda coluna exibe as reconstruções feitas a partir das imagens de duas câmeras e a terceira coluna, as reconstruções feitas utilizando as imagens das três câmeras.

Tabela 6.1: Imagens dos conjuntos 1, 2, 3 e 4, utilizados nas reconstruções das cenas.

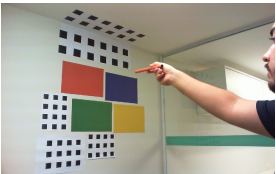

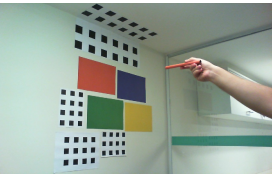






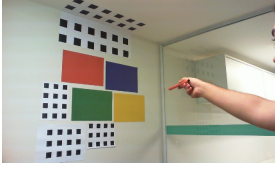


Conjunto n°	Imagem da Primeira Câmera	Imagem da Segunda Câmera	Imagem da Terceira Câmera
1			
2			
3			
4			

Tabela 6.2: Imagens dos conjuntos 5, 6, 7 e 8, utilizados nas reconstruções das cenas.

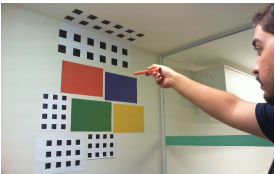

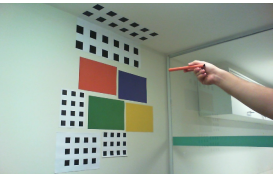



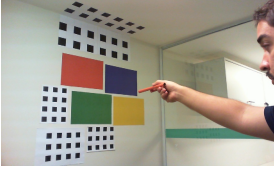


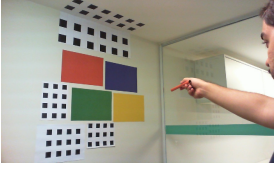


Conjunto n°	Imagem da Primeira Câmera	Imagem da Segunda Câmera	Imagem da Terceira Câmera
5			
6			
7			
8			

Tabela 6.3: Resultados das reconstruções das cenas a partir dos conjuntos de imagens 1, 2, 3 e 4.

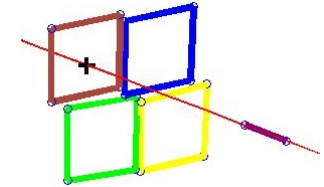
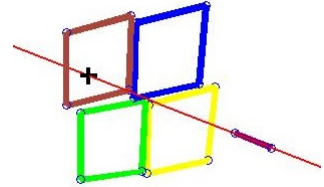
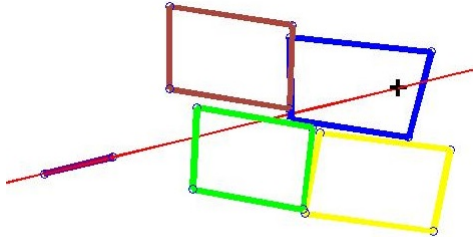
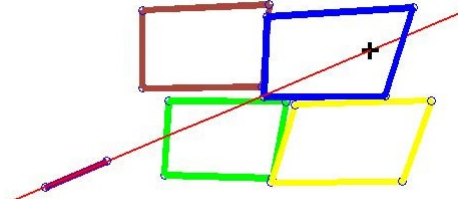
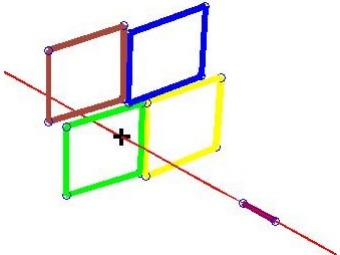
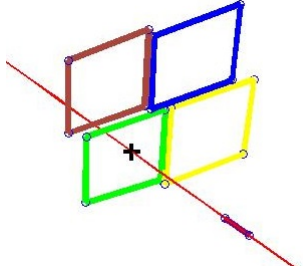
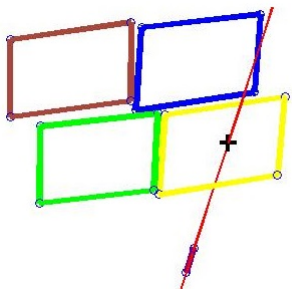
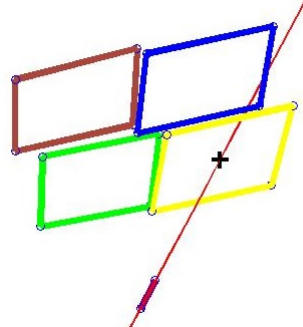
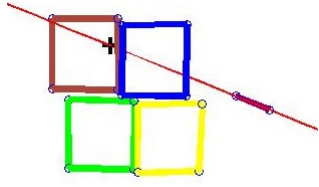
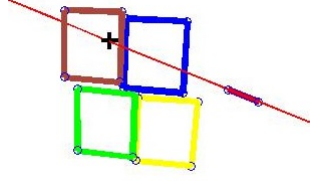
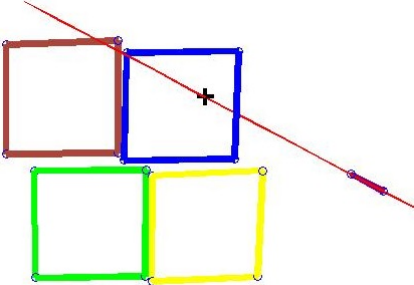
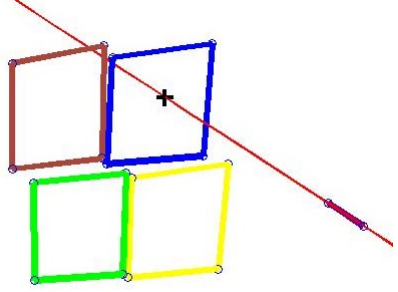
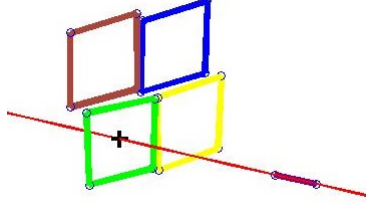
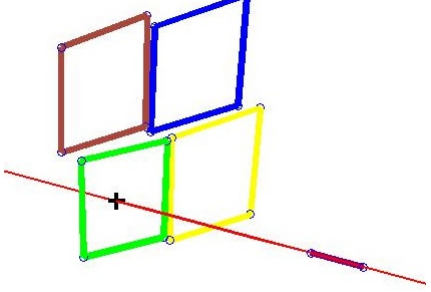
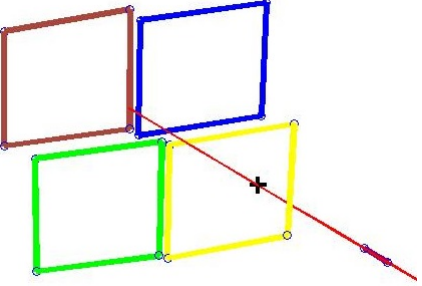
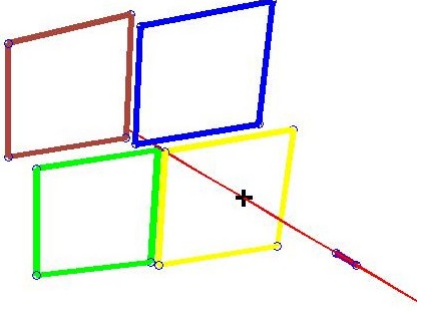
Conjunto n°	Resultado da Reconstrução com Duas Câmeras	Resultado da Reconstrução com Três Câmeras
1		
2		
3		
4		

Tabela 6.4: Resultados das reconstruções das cenas a partir dos conjuntos de imagens 5, 6, 7 e 8.

Conjunto n°	Resultado da Reconstrução com Duas Câmeras	Resultado da Reconstrução com Três Câmeras
5		
6		
7		
8		

Capítulo 7

Conclusões

Os resultados deste trabalho mostram que os algoritmos utilizados foram capazes de reconstruir corretamente as cenas e atender ao objetivo de determinar os objetos apontados em cada uma. Tanto para o caso de duas imagens quanto para o de três, não houve nenhum erro de detecção.

Do resultado da análise qualitativa da reconstrução feita sobre os dados das Tabelas 6.3 e 6.4, utilizando as cenas reconstruídas a partir das matrizes de câmera obtidas das matrizes essenciais (da qual é possível extrair matrizes de câmera sem distorção projetiva) e de informações a respeito da calibração das câmeras, conclui-se que entre as reconstruções realizadas com duas e três câmeras não houve diferenças relevantes para a detecção dos objetos apontados que foi proposta neste trabalho. As variações observadas em algumas das reconstruções ocorreram devido à imprecisão na obtenção dos vértices dos retângulos na fase de pré-processamento. É possível, também, que a terceira câmera tenha introduzido erros numéricos adicionais.

Como trabalho futuro, pode-se integrar este sistema com outros sistemas de visão 3D, utilizando uma possível vantagem da terceira câmera, que é a menor ocorrência de oclusão. Outra possibilidade é a modificação do sistema implementado neste trabalho. É possível acrescentar outras funcionalidades como, por exemplo, reconhecimento de gestos para eliminar a necessidade de utilização de um apontador para indicar os objetos selecionados.

Outro ponto a ser explorado é a realização de um estudo quantitativo do desempenho do sistema com diferentes técnicas de reconstrução. Neste trabalho optou-se por não utilizar referenciais globais para a reconstrução. Utilizando um *grid* de calibração como referência para um sistema de coordenadas globais seria possível reconstruir e dimensionar objetos em uma cena e comparar as dimensões dos objetos reconstruídos com as dos objetos reais, possibilitando uma análise mais precisa dos resultados.

Ainda com relação ao estudo quantitativo, para se verificar a acurácia da marcação dos pontos de interseção entre o prolongamento do apontador e os objetos,

pode-se utilizar uma ponteira laser no lugar do apontador utilizado neste trabalho. Dessa forma, a imagem do ponto luminoso criado pela ponteira poderia ser utilizada como referência para a localização correta da interseção entre o prolongamento da ponteira e o objeto apontado.

Os efeitos de diferentes configurações das câmeras nas reconstruções da cena também podem ser testados. Pode-se, por exemplo, avaliar como a distância entre as câmeras e os objetos afetaria os resultados. Também pode ser realizado um estudo relacionando a resolução das câmeras e a qualidade das reconstruções.

Referências Bibliográficas

- [1] COEN, M. “The future of human-computer interaction, or how I learned to stop worrying and love my intelligent room”, *IEEE Intelligent Systems*, v. 14, n. 2, pp. 8–19, 1999.
- [2] IRIE, K., UMEDA, K. “3D Measurement by Distributed Camera System for Constructing an Intelligent Room”, *Networked Sensing Systems*, pp. 118–121, 2007.
- [3] PINTO, D. C. C. *Dimensionamento 3D Utilizando Câmeras Digitais de Alta Resolução*. Projeto de graduação, Universidade Federal do Rio de Janeiro, 2010.
- [4] SILVA, L. C. *Método Robusto para a Calibração de Câmeras em Estereofotogrametria*. Tese de doutorado, COPPE - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, Universidade Federal do Rio de Janeiro, 2003.
- [5] AIRES, G. S. *Estimação de Movimento de uma Câmera Via Geometria Epipolar*. Dissertação de mestrado, COPPE - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia, Universidade Federal do Rio de Janeiro, 2010.
- [6] FAUGERAS, O. *Three-Dimensional Computer Vision: A Geometric View Point*. MIT Press, 1993.
- [7] ZISSERMAN, A., HARTLEY, R. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003.

Apêndice A

Códigos em MATLAB

A.1 Obtendo a Matriz de Calibração

```
%Cálculo da matriz da câmera (P) e da matriz de calibração (K)
%Entradas:
%N: número de pontos de referência
%u e v: coordenadas dos pontos de referência na imagem
%grid: coordenadas 3D dos pontos de referência

function [P K] = calculate_P(N, u, v, grid)

%Obtendo a matriz P pelo método dos multiplicadores de lagrange

for index_t = 0:N-1,
    A(index_t*2+1,:) = [grid(index_t+1,:) 1 zeros(1,4) -grid(index_t+1,:)*u(index_t
        +1) -u(index_t+1)];
    A(index_t*2+2,:) = [zeros(1,4) grid(index_t+1,:) 1 -grid(index_t+1,:)*v(index_t
        +1) -v(index_t+1)];
end

C = [A(:,1:8) A(:,12)];
D = A(:,9:11);

E = D*(eye(2*N)-C*(C'*C)^-1*C')*D;
[eval eval] = eig(E);
min_eval = min(diag(eval));
[lin col] = find(eval==min_eval);

z = evec(:, col);
y = -(C'*C)^-1*C'*D*z;

p = [y(1:8); z; y(9)];
P = reshape(p,4,3)';

%Obtendo K através da decomposição RQ

K = P(:,1:3);

theta = atan(-K(3,2)/K(3,3));
c = cos(theta);
s = sin(theta);
```

```

Qx = [1 0 0;
      0 c -s;
      0 s c];

K = K*Qx;

theta = atan(K(3,1)/K(3,3));
c = cos(theta);
s = sin(theta);

Qy = [c 0 s;
      0 1 0;
      -s 0 c];

K = K*Qy;

theta = atan(-K(2,1)/K(2,2));
c = cos(theta);
s = sin(theta);

Qz = [c -s 0;
      s c 0;
      0 0 1];

K = K*Qz;
R = Qz'*Qy'*Qx';

I = diag([sign(K(1,1)) sign(K(2,2)) sign(K(3,3))]);
K = K*I;
R = I*R;

end

```

A.2 Obtendo as Matrizes das Câmeras Usando a Matriz Fundamental

```

%Cálculo das matrizes das câmeras com ambiguidade projetiva (P1 e P2)
%Sem ambiguidade projetiva (P11 e P21 ou P22 ou P23 ou P24)
%Entradas:
%N: número de pontos de referência
%u e v: coordenadas dos pontos de referência nas imagens (uma coluna para cada
        imagem)
%K e K1: matrizes de calibração (para remover ambiguidade)

function [ P1, P2, P11, P21, P22, P23, P24 ] = calculate_F( u, v, K, K1, R, R1, N )

%normalizando as coordenadas dos pontos
%translação -> centroide em (0,0)
%escalamento -> RMS = sqrt(2)

translation_u = - mean(u);
translation_v = - mean(v);

u2 = (u+repmat(translation_u,N,1));
v2 = (v+repmat(translation_v,N,1));

scale = sqrt(2./(sum(u2.*u2+v2.*v2)/N));

```

```

u3 = u2.*repmat(scale,N,1);
v3 = v2.*repmat(scale,N,1);

%utilizando SVD para minimizar Af, com ||f|| = 1;

A = [u3(:,2).*u3(:,1) u3(:,2).*v3(:,1) u3(:,2) v3(:,2).*u3(:,1) v3(:,2).*v3(:,1) v3
(:,2) u3(:,1) v3(:,1) ones(N,1)];
[U D V] = svd(A);
f = V(:,9);
F = reshape(f,3,3)';
[U D V] = svd(F);
D(3,3) = 0;
F2 = U*D*V';

%desnormalizando

T1 = scale(1)*[[1 0;0 1; 0 0] [translation_u(1); translation_v(1); 1/scale(1)]];
T2 = scale(2)*[[1 0;0 1; 0 0] [translation_u(2); translation_v(2); 1/scale(2)]];

F3 = T2'*F2*T1;

%calculando as matrizes projetivas

P1 = [eye(3) zeros(3,1)];
[U D V] = svd(F3');
e = V(:,3);
ex = [ 0 -e(3) e(2);
      e(3) 0 -e(1);
      -e(2) e(1) 0];

F=F3;
P2 = [ex*F e];

%removendo ambiguidade

E = K1'*F*K;

[U D V] = svd(E);
%V = V*mean([D(1,1) D(2,2)]);
%D = D/mean([D(1,1) D(2,2)]);

W = [0 -1 0;
      1 0 0;
      0 0 1];

U3 = U*[0 0 1]';
P21 = [U*W*V' U3];
P22 = [U*W*V' -U3];
P23 = [U*W*V' U3];
P24 = [U*W*V' -U3];

P11 = K*P1;
P21 = K1*P21;
P22 = K1*P22;
P23 = K1*P23;
P24 = K1*P24;

end

```

A.3 Obtendo as Matrizes das Câmeras Usando o Tensor Trifocal

```

%Cálculo das matrizes de câmera com distorção projetiva (P, P1, P2)
%Sem distorção (Pa, P1a, P1b, P1c, P1d, P2a, P2b, P2c, P2d)
%Entradas:
%N: número de pontos de referência
%u e v: coordenadas dos pontos de referência nas imagens (uma coluna para cada
    imagem)
%K, K1 e K2: matrizes de calibração para remoção da ambiguidade projetiva

function [ P,P1,P2,Pa,P1a,P1b,P1c,P1d,P2a,P2b,P2c,P2d ] = calculate_T( u, v, K, K1,
    K2, N )

x=[u(:,1) v(:,1) ones(N,1)];
x1=[u(:,2) v(:,2) ones(N,1)];
x2=[u(:,3) v(:,3) ones(N,1)];

%normalizando as coordenadas dos pontos
%translação -> centroide em (0,0)
%escalamento -> RMS = sqrt(2)

translation_x = - mean(x(:,1:2));
translation_x1 = - mean(x1(:,1:2));
translation_x2 = - mean(x2(:,1:2));

xb(:,1:2) = (x(:,1:2)+repmat(translation_x,N,1));
x1b(:,1:2) = (x1(:,1:2)+repmat(translation_x1,N,1));
x2b(:,1:2) = (x2(:,1:2)+repmat(translation_x2,N,1));

scale_x = 1/sqrt(sum(sum(xb(:,1:2).*xb(:,1:2)))/(2*N));
scale_x1 = 1/sqrt(sum(sum(x1b(:,1:2).*x1b(:,1:2)))/(2*N));
scale_x2 = 1/sqrt(sum(sum(x2b(:,1:2).*x2b(:,1:2)))/(2*N));

xc = [xb.*repmat(scale_x,N,2) ones(N,1)];
x1c = [x1b.*repmat(scale_x1,N,2) ones(N,1)];
x2c = [x2b.*repmat(scale_x2,N,2) ones(N,1)];

H = scale_x*[[1 0;0 1; 0 0] [translation_x'; 1/scale_x]];
H1 = scale_x1*[[1 0;0 1; 0 0] [translation_x1'; 1/scale_x1]];
H2 = scale_x2*[[1 0;0 1; 0 0] [translation_x2'; 1/scale_x2]];

%Montando as equações
count_eq = 1;
Ab = zeros(4*N,27);
for n=1:N,
    for i=1:2,
        for l=1:2,
            for k=1:3,
                Ab(count_eq,3+(3-1)*3+(k-1)*9) = xc(n,k)*x1c(n,i)*x2c(n,l);
                Ab(count_eq,3+(i-1)*3+(k-1)*9) = -xc(n,k)*x2c(n,l);
                Ab(count_eq,1+(3-1)*3+(k-1)*9) = -xc(n,k)*x1c(n,i);
                Ab(count_eq,1+(i-1)*3+(k-1)*9) = xc(n,k);
            end
            count_eq = count_eq+1;
        end
    end
end

```

```

end

[U D V] = svd(Ab);

tb = V(:,27);
Tb(:, :, 1) = reshape(tb(1:9),3,3)';
Tb(:, :, 2) = reshape(tb(10:18),3,3)';
Tb(:, :, 3) = reshape(tb(19:27),3,3)';

%Recuperando os epipolos
[U D V] = svd(Tb(:, :, 1));
v1 = V(:,3);
[U D V] = svd(Tb(:, :, 2));
v2 = V(:,3);
[U D V] = svd(Tb(:, :, 3));
v3 = V(:,3);

Vv = [v1 v2 v3]';
[U D V] = svd(Vv);

e2 = V(:,3);

[U D V] = svd(Tb(:, :, 1)');
v1 = V(:,3);
[U D V] = svd(Tb(:, :, 2)');
v2 = V(:,3);
[U D V] = svd(Tb(:, :, 3)');
v3 = V(:,3);

Vv = [v1 v2 v3]';
[U D V] = svd(Vv);

e1 = V(:,3);

% Montando a equação t = Oa
a4 = e1;
b4 = e2;
O = 0;

for j=1:3
    for k=1:3
        for i=1:3
            O(k+3*(j-1)+9*(i-1), j+3*(i-1)) = b4(k);
            O(k+3*(j-1)+9*(i-1), k+3*(i-1)+9) = -a4(j);
        end
    end
end

% Calculando 'a', o vetor das componentes restantes de P1 e P2 e obtendo o
% tensor trifocal final.
[U D V] = svd(O);
U1 = U(:, 1:rank(O));
[Ub Db Vb] = svd(Ab*U1);
temp = Vb(:, size(Vb,2));
tc = U1*temp;
V1 = V(:, 1:rank(O));
D1 = D(1:rank(O), 1:rank(O));
a = V1*D1^-1*temp;

```

```

Tc(:, :, 1) = reshape(tc(1:9), 3, 3)';
Tc(:, :, 2) = reshape(tc(10:18), 3, 3)';
Tc(:, :, 3) = reshape(tc(19:27), 3, 3)';

%desnormalizando

H2i = H2^-1;
H1i = H1^-1;
Hi = H^-1;

for i=1:3,
    for j=1:3,
        for k=1:3,
            temp1(:, 1) = H2i(k, :) * Tc(:, :, 1)';
            temp1(:, 2) = H2i(k, :) * Tc(:, :, 2)';
            temp1(:, 3) = H2i(k, :) * Tc(:, :, 3)';
            temp2 = H1i(j, :) * temp1;
            T(j, k, i) = H(:, i)' * temp2';
        end
    end
end

e1temp = e1;
e2temp = e2;
e1 = H1i * e1;
e2 = H2i * e2;

% Recuperando as matrizes das câmeras
P = Hi * [eye(3) zeros(3, 1)];
P1 = H1i * [reshape(a(1:9), 3, 3) e1temp];
P2 = H2i * [reshape(a(10:18), 3, 3) e2temp];

% Obtendo as matrizes fundamentais
e1x = [0 -e1(3, 1) e1(2, 1); e1(3, 1) 0 -e1(1, 1); -e1(2, 1) e1(1, 1) 0];
e2x = [0 -e2(3, 1) e2(2, 1); e2(3, 1) 0 -e2(1, 1); -e2(2, 1) e2(1, 1) 0];

F21 = e1x * [T(:, :, 1) * e2, T(:, :, 2) * e2, T(:, :, 3) * e2];
F31 = e2x * [T(:, :, 1) * e1, T(:, :, 2) * e1, T(:, :, 3) * e1];

% Obtendo P1 (com parâmetros internos)
Pa = K * [eye(3) zeros(3, 1)];

% Removendo ambiguidade da matriz P1

E21 = K' * F21 * K1;

[U D V] = svd(E21);

W = [0 -1 0;
      1 0 0;
      0 0 1];

U3 = U * [0 0 1]';
P1a = [U * W * V' U3];
P1b = [U * W * V' -U3];
P1c = [U * W * V' U3];
P1d = [U * W * V' -U3];

P1a = K1 * P1a;

```

```

P1b = K1*P1b;
P1c = K1*P1c;
P1d = K1*P1d;

% Removendo ambiguidade da matriz P2

E31 = K2'*F31*K;

[U D V] = svd(E31);

W = [0 -1 0;
      1 0 0;
      0 0 1];

U3 = U*[0 0 1]';
P2a = [U*W*V' U3];
P2b = [U*W*V' -U3];
P2c = [U*W'*V' U3];
P2d = [U*W'*V' -U3];

P2a = K2*P2a;
P2b = K2*P2b;
P2c = K2*P2c;
P2d = K2*P2d;

end

```