



Centro de Tecnologia e Urbanismo
Departamento de Engenharia Elétrica

Rodrigo Henrique Cunha Palácios

Um Conjunto de Ferramentas para Estimar a Matriz Fundamental em Aplicações de Visão Computacional

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Estadual de Londrina para obtenção do Título de Mestre em Engenharia Elétrica.



Rodrigo Henrique Cunha Palácios

Um Conjunto de Ferramentas para Estimar a Matriz Fundamental em Aplicações de Visão Computacional

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Estadual de Londrina para obtenção do Título de Mestre em Engenharia Elétrica.

Área de concentração: Visão Computacional
Especialidade: Calibração de Câmeras

Orientador:
Prof. Dr. José Alexandre de França

Londrina, PR
2010

**Catálogo elaborado pela Divisão de Processos Técnicos da Biblioteca Central da
Universidade Estadual de Londrina.**

Dados Internacionais de Catalogação-na-Publicação (CIP)

P153c Palácios, Rodrigo Henrique Cunha.
Um conjunto de ferramentas para estimar a matriz fundamental em
aplicações de visão computacional / Rodrigo Henrique Cunha Palácios. –
Londrina, 2010.
81 f. : il.

Orientador: José Alexandre de França.
Dissertação (Mestrado em Engenharia Elétrica) – Universidade
Estadual de Londrina, Centro de Tecnologia e Urbanismo, Programa de
Pós-Graduação em Engenharia Elétrica, 2010.
Inclui bibliografia.

1. Engenharia elétrica – Matemática – Teses. 2. Matlab (Programa de
computador) – Teses. 3. Calibração de câmeras – Teses. 4. Computação –
Matemática – Teses. 5. Engenharia elétrica – Matriz fundamental – Teses.
I. França, José Alexandre de. II. Universidade Estadual de Londrina.
Centro de Tecnologia e Urbanismo. Programa de Pós-graduação em
Engenharia Elétrica. III. Título.

CDU 621.3:519.68

Rodrigo Henrique Cunha Palácios

Um Conjunto de Ferramentas para Estimar a Matriz Fundamental em Aplicações de Visão Computacional

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Estadual de Londrina para obtenção do Título de Mestre em Engenharia Elétrica.

Área de concentração: Visão Computacional
Especialidade: Calibração de Câmeras

Comissão Examinadora

Prof. Dr. José Alexandre de França
Departamento de Engenharia Elétrica
Universidade Estadual de Londrina
Orientador

Prof. Dr. Leonimer Flávio de Melo
Departamento de Engenharia Elétrica
Universidade Estadual de Londrina

Prof. Dr.-Ing. Marcelo Ricardo Stemmer
Departamento de Automação e Sistemas
Universidade Federal de Santa Catarina

13 de dezembro de 2010

Aos meus filhos, Mariana e João.

À minha esposa Daniela.

Aos meus pais.

Todos, razão da minha vida.

Agradecimentos

Primeiramente a Deus.

À minha esposa e meus filhos, Mariana e João, por todo carinho, apoio e compreensão.

Aos professores do Programa de Mestrado em Engenharia Elétrica da Universidade Estadual de Londrina.

Aos colegas da Universidade Tecnológica Federal do Paraná, pelo apoio nos momentos difíceis.

Ao meu orientador, José Alexandre de França, pelos vários ensinamentos e apoio incondicional durante o período do Mestrado.

Resumo

Na visão binocular, quando o sistema não está calibrado, as informações geométricas das câmeras estão concentradas em uma matriz quadrada de ordem três, conhecida como matriz fundamental. Esta torna todas as operações da visão binocular menos complexas e pode ser estimada a partir de um conjunto de correspondências de pontos. O presente trabalho apresenta um estudo dos principais pacotes do Matlab aplicados ao ensino, mais especificamente os voltados à visão computacional, além da proposta e implementação de um novo ambiente computacional interativo, para Matlab, que permite a comparação entre os principais métodos para estimar a matriz fundamental. Esses métodos são classificados em três tipos: lineares, não-lineares e robustos. Por sua vez, a matriz fundamental pode ser estimada por meio de dados sintéticos, configurados pelo próprio usuário e, também, a partir de um conjunto de correspondência de pontos extraídos de imagens fornecidas pelo próprio usuário. As etapas envolvidas no processo são apresentadas de forma bastante intuitiva, nas quais é possível escolher os métodos a serem utilizados na estimação, bem como os parâmetros de configuração de cada um dos algoritmos. Isto, além de permitir o entendimento dos mesmos, facilita a confrontação dos métodos em termos de exatidão e número de iterações.

Abstract

In binocular vision, when the system is not calibrated, the geometric information of the cameras are focused on a square matrix of order three, known as the fundamental matrix. This makes all operations of binocular vision less complex and can be estimated from a set of correspondence points. This dissertation presents a study of the main packages of Matlab applied to education, more specifically focused on computer vision beyond the proposal and implementation of a new interactive computing environment for Matlab, which allows comparison of the main methods for estimating the matrix fundamental. These methods are classified into three types: linear, nonlinear and robust. In turn, the fundamental matrix can be estimated by using synthetic data, configured by the user and also from a matching set of points extracted from images provided by the user. The steps involved in the process are presented in an intuitive way in which you can choose the methods to be used in the estimation and the configuration parameters of each algorithm. This, besides allowing the understanding of them, facilitates the comparison of methods in terms of accuracy and number of iterations.

Sumário

Lista de Figuras

Lista de Símbolos e Abreviações

1	Introdução	1
1.1	Objetivos	3
1.1.1	Objetivos Específicos	4
1.2	Justificativas	4
1.3	Estrutura da Dissertação	5
2	Fundamentação Teórica	6
2.1	Matlab	6
2.1.1	Pacotes Educacionais para Matlab	8
2.1.2	Pacotes de Funções para Visão Computacional	9
2.2	O Modelo da Câmera	11
2.3	Visão Binocular	15
2.3.1	Geometria Epipolar	15
2.3.2	Matriz Fundamental	17
2.3.3	Estimação da Matriz Fundamental	19
2.3.4	Correspondência de Pontos	33
3	A Construção do Ambiente Interativo Computacional para o Cálculo da Matriz Fundamental	36
3.1	Menu Inicial	37
3.2	Geração de Dados Sintéticos	37

3.3	Selecionando Imagens Reais	39
3.4	Detecção de Cantos	41
3.5	Correspondência de Pontos	41
3.6	Comparação de Métodos para Estimar Matriz Fundamental	43
4	Resultados e Discussões	48
4.1	Resultados com Imagens Reais	48
4.2	Resultados com Dados Sintéticos	51
5	Conclusão	53
 Anexo A – Artigo para Publicação - Um Ambiente Interativo para o Ensino da Geometria Epipolar		55
A.1	Introdução	55
A.2	Notação	58
A.3	Preliminares	58
A.4	Estimação da Matriz Fundamental	61
A.4.1	Métodos Lineares	61
A.4.2	Métodos Não-Lineares	63
A.4.3	Métodos Robustos	65
A.5	A Construção do Ambiente de Simulação para Estimar a Matriz Fundamental	67
A.6	Um Estudo de Caso	70
A.6.1	Geração de Dados Sintéticos	70
A.6.2	Selecionar Imagens	72
A.6.3	Detectar Cantos	72
A.6.4	Corresponder Pontos	73
A.6.5	Comparar Métodos para Estimar Matriz Fundamental	73
A.7	Conclusão	76
	 Referências	78

Lista de Figuras

1.1	Câmera 3D da Panasonic.	2
2.1	Esboço de uma câmera <i>pinhole</i>	12
2.2	Coordenadas do modelo de câmera <i>pinhole</i>	12
2.3	Determinação da ordenada do ponto \mathbf{m} em função das coordenadas do ponto \mathbf{M}	13
2.4	Geometria Epipolar	16
2.5	Diagrama de Blocos que Demonstra os Passos para Estimar a Matriz Fundamental	19
2.6	Figuras com os cantos identificados por meio do método de Harris(HARRIS; STEPHENS, 1988): a) câmera 1 (esquerda); b) câmera 2 (direita)	35
3.1	Atividades do Ambiente Computacional Proposto	37
3.2	Atividades do Menu Inicial	38
3.3	Menu Inicial do Simulador para Estimar a Matriz Fundamental	38
3.4	Atividades da Geração de Dados Sintéticos	39
3.5	Geração de Dados Sintéticos a Partir de um Cubo Projetado	39
3.6	Atividades da Leitura das Imagens pelo Ambiente Computacional	40
3.7	Carregando as Imagens PLANTA no Ambiente Interativo	40
3.8	Atividades da Detecção de Cantos pelo Ambiente Computacional	41
3.9	Execução da Detecção de Cantos das Imagens PLANTA	42
3.10	Atividades da Correspondência de Pontos pelo Ambiente Computacional	42
3.11	Execução da Correspondência de Pontos	43
3.12	Atividades da Simulação de Comparações entre Métodos para Estimar a Matriz Fundamental	44

3.13	O Novo Ambiente Estimando a Matriz Fundamental dos Métodos Lineares, Não-Lineares e Robustos	45
3.14	Resultados Textuais Gerados pelo Novo Ambiente da Simulação da Estimação da Matriz Fundamental no Ambiente Matlab	45
3.15	Exibição das Retas Epipolares Calculadas a partir da Estimação da Matriz Fundamental pelo Método Liu e Manner	46
4.1	Imagens Utilizadas nos Testes Realizados: a) câmera 1 (esquerda); b) câmera 2 (direita)	48
4.2	Resultados da Simulação da Matriz Fundamental dos Métodos Lineares com Imagens Reais	49
4.3	Resultados da Simulação da Matriz Fundamental dos Métodos Não-Lineares com Imagens Reais	50
4.4	Resultados da Simulação da Matriz Fundamental dos Métodos Robustos com Imagens Reais	50
4.5	Interface de Simulação de Dados Sintéticos para Simulação do Cálculo da Matriz Fundamental	51
4.6	Resultados da Simulação da Matriz Fundamental com um Método de cada Classificação com Dados Sintéticos	52
A.1	Esboço da geometria epipolar.	59
A.2	Atividades do Ambiente Computacional Proposto	68
A.3	Geração de Dados Sintéticos a Partir de um Cubo Projetado	71
A.4	Carregando as Imagens PLANTA no Ambiente Interativo	72
A.5	Execução da Detecção de Cantos das Imagens PLANTA	73
A.6	Execução da Correspondência de Pontos	74
A.7	O Novo Ambiente Estimando a Matriz Fundamental dos Métodos Lineares, Não-Lineares e Robustos	75
A.8	Resultados Textuais Gerados pelo Novo Ambiente da Simulação da Estimação da Matriz Fundamental no Ambiente Matlab	75
A.9	Exibição das Retas Epipolares Calculadas a partir da Estimação da Matriz Fundamental pelo Método Golden	76

Lista de Símbolos e Abreviações

$\text{diag}(\dots)$	$\text{diag}(a_1, a_2, \dots, a_n)$ representa uma matriz $n \times n$, cuja diagonal principal é dada por a_1, a_2, \dots, a_n e todos os outros elementos são iguais a zero.
$\mathbf{x} \leftrightarrow \mathbf{y}$	Indica correspondência entre os pontos \mathbf{x} e \mathbf{y} ou entre as retas \mathbf{x} e \mathbf{y} .
$\mathbb{R}^{n \times m}$	Representa o conjunto de todas as matrizes de dimensão $n \times m$.
\mathbb{R}^n	Representa o conjunto de todos os vetores $n \times 1$.
$[\mathbf{x}]_{\times}$	É a matriz antissimétrica (3×3) do vetor \mathbf{x} , ou seja, $[\mathbf{x}]_{\times} \mathbf{v} = \mathbf{x} \times \mathbf{v}$, onde $\mathbf{v} \in \mathbb{R}^3$,
θ_{sk}	Ângulo que corrige a ortogonalidade da matriz de sensores da câmera. $\theta_{sk} \approx \frac{\pi}{2}$.
$\mathbf{0}_n$	Vetor nulo de dimensão $n \times 1$.
$d(\mathbf{x}, \mathbf{y})$	Indica distância Euclidiana entre os pontos \mathbf{x} e \mathbf{y} ou entre o ponto \mathbf{x} e a reta \mathbf{y} .
f	Distância focal da câmera.
\mathbf{A}	Matriz 3×3 que contém os parâmetros intrínsecos da câmera 1.
\mathbf{C}_x	Representa o centro da câmera x .
\mathbf{F}	Matriz fundamental (3×3) que relaciona duas imagens de um sistema binocular não-calibrado.
\mathbf{E}	Matriz essencial (3×3) que relaciona duas imagens de um sistema binocular calibrado.
$\hat{\mathbf{F}}$	Matriz fundamental estimada.
\mathbf{T}_x	Transformações aplicadas à um conjunto de correspondências, como exemplo, tem-se $(\mathbf{T}_2 \tilde{\mathbf{m}}_{1i} \leftrightarrow \mathbf{T}_2 \tilde{\mathbf{m}}_{2i})$.
\mathbf{H}_{∞}	Matriz 3×3 que representa a homografia do infinito.
\mathbf{I}_n	Matriz identidade de dimensão $n \times n$.
I_x	O plano de imagem da “câmera x ”. A “câmera 1” refere-se a câmera que está mais a esquerda em um conjunto estéreo. A “câmera 2” é a que está a direita.

continua...

\mathbf{l}_∞	Vetor que representa a equação da reta no infinito, no plano da imagem.
\mathbf{l}_{xi}	Vetor que representa a i -ésima reta epipolar que está no plano I_x .
$\widetilde{\mathbf{M}}_i$	Representa, em coordenadas homogêneas, um ponto 3D. Geralmente, $\widetilde{\mathbf{M}}_i = [X_i, Y_i, Z_i, T_i]^T$ e $\mathbf{M}_i = [X_i, Y_i, Z_i]^T$.
$\widetilde{\mathbf{m}}_{xi}$	Representa, em coordenadas homogêneas, a projeção de ponto 3D no plano de imagem I_x . Geralmente, $\widetilde{\mathbf{m}}_{xi} = [u_{xi}, v_{xi}, t_{xi}]^T$ e $\mathbf{m}_{xi} = [u_{xi}, v_{xi}]^T$.
\mathbf{m}_0	Ponto central da câmera. Geralmente, $\mathbf{m}_0 = [u_0, v_0]^T$.
\mathbf{P}	Matriz de projeção (3×4) da câmera.
SVD	Decomposição do autovalor (Do inglês: Singular Value Decomposition).
\mathbf{T}_{PE}	Matriz 4×4 que representa a homografia que atualiza uma matriz de projeção Euclidiana em uma projetiva.
1D	Uma dimensão.
2D	Duas dimensões.
3D	Três dimensões.

1 Introdução

O sentido da visão humana permite obter uma quantidade grande de informações sem nenhum contato físico, fato esse que possibilita interagir com os mais diferentes objetos. Apesar da complexidade biológica, existe na comunidade científica um esforço enorme na busca de teorias, tecnologias, algoritmos e sistemas que possibilitem representar, computacionalmente, a visão humana. Os sistemas, atualmente, são construídos para aplicações específicas, necessitando de condições especiais para o funcionamento. Em virtude desses fatos, entende-se que as técnicas de visão computacional ainda está distante de uma representação fidedigna da visão biológica humana, mas, pode-se descrever como um complemento. Na visão computacional, estuda-se e descreve-se sistemas de visão artificial implementados por *hardware* ou *software*.

De acordo com Godoy (2005), a simulação de duas imagens da cena que são projetadas nos olhos em pontos de observação ligeiramente diferentes é denominado estereoscopia. Nessa simulação, o cérebro funde as duas imagens e, nesse processo, obtém-se informações quanto à profundidade, distância, posição e tamanho dos objetos, gerando uma sensação de visão 3D. Já o fenômeno ótico denominado paralaxe, consiste na mudança aparente de posição de um objeto causada pela mudança de posição do observador, pode-se medir distâncias com base na paralaxe, isto é, criar métodos de medição baseada na paralaxe. Para aplicações em visão computacional, Boufama e Mohr (1998) propuseram uma técnica para calcular a matriz fundamental baseada na paralaxe entre duas câmeras, denominada paralaxe virtual, na qual, computa-se a homografia entre duas imagens de um plano virtual e na detecção do paralaxe de pontos não pertencentes a este plano.

Um par de câmeras simula a visão do olho humano, onde cada lente deve ser colocada em pontos de vista distintos. Nesse processo ainda devem ser controlados zoom, foco, abertura, enquadramento e o ângulo relativo entre elas. Já há no mercado câmeras com tecnologia 3D, que possuem duas matrizes de sensores em um único equipamento, conforme mostrado na figura 1.1.



Figura 1.1: Câmera 3D da Panasonic.

É notável que o dispositivo mais importante de um sistema de visão computacional é a câmera, pois é ela que fornece os dados que serão processados, assim como o olho humano na visão biológica. Contudo, ao se utilizar esse dispositivo na formação das imagens, todas as informações caracterizadas em três dimensões são perdidas. Uma das alternativas para minimizar essa perda é a utilização de sistemas com visão binocular. Para isso, duas câmeras fixas capturam imagens de uma mesma cena. Porém, tais sistemas não estão livres de restrições. Hartley e Zisserman (2003) deixam claro em seus pressupostos que a principal restrição é a epipolar. Essa relaciona pontos na imagem capturada por uma das câmeras a pontos na imagem da outra câmera. A inexistência de calibração no sistema em questão é completamente mapeada pela matriz fundamental. Tal matriz, concentra uma série de informações sobre o sistema binocular e torna todas as operações realizadas menos complexas. Porém, para se estimar a referida matriz é necessário obter um conjunto inicial de correspondência de pontos entre duas imagens distintas de uma mesma cena.

A estimação da matriz fundamental facilita uma série de tarefas na visão computacional como, por exemplo, correspondência binocular, reconstrução 3D, calibração de câmeras, entre outras. Calibrar câmeras consiste em determinar as características geométricas e ópticas internas da câmera assim como sua orientação e posição em relação ao sistema de coordenadas da cena. A reconstrução 3D consiste na criação de modelos espaciais que visam dar uma noção mais realística de uma determinada cena obtida a partir de imagens. Pode-se destacar a recuperação das informações de documentos arquitetônicos e arqueológicos. Há, também, aplicações na área médica, principalmente relacionadas à reconstrução de órgãos para obtenção de diagnósticos médicos. Se os parâmetros da câmera são conhecidos, o sistema é dito calibrado. No entanto, é possível recuperar as informações 3D de um conjunto binocular, sem mesmo haver calibração. Isso pode ser

perfeitamente realizado por meio da geometria epipolar, através de um conjunto de correspondências de pontos. A matriz fundamental encapsula as informações dessa geometria.

Há muitas décadas, estimar a matriz fundamental é objeto de estudo. Até o início da década de 90, os esforços concentravam-se na exatidão da estimação. A um custo computacional elevado, a matriz fundamental era obtida por meio de parametrizações complexas e métodos não-lineares que minimizam algum critério específico. A partir de meados dos anos 90, foram propostos diversos métodos robustos que identificam e eliminam falsas correspondências que degradam a qualidade da matriz fundamental obtida (ZHANG, 1998; TORR; ZISSERMAN, 1998; TORR; MURRAY, 1997). Esses métodos tornaram possível estimar a matriz fundamental de forma automática. Mais recentemente, surge a tendência de tentar reduzir o custo computacional dos algoritmos não-lineares utilizando parametrizações mais simples (BARTOLI; STURM, 2004) ou espaços de busca reduzidos (ZHANG; LOOP, 2001; BARTOLI; STURM; HORAUD, 2001).

O presente trabalho propõe um conjunto de ferramentas e um ambiente computacional interativo, desenvolvidos para o Matlab, que permite estimar a matriz fundamental. O processo para estimar a matriz é iniciado com um conjunto de correspondência de pontos, extraído de um par de imagens de uma mesma cena, fornecidas por um usuário qualquer. O referido processo pode ser executado manualmente, pelo Matlab ou por meio do ambiente computacional interativo proposto. O conjunto de ferramentas permite estimar a matriz fundamental com diversos métodos advindos da literatura, possibilitando comparações entre os mesmos. Facilitar o entendimento e a escolha do método mais adequado para uma aplicação específica é uma das prerrogativas desse ambiente computacional.

1.1 **Objetivos**

O objetivo deste trabalho é o desenvolvimento de um conjunto de ferramentas, voltadas ao Matlab, para simular o processo do cálculo da matriz fundamental de um conjunto binocular, por meio dos principais métodos proposto na literatura. Além disso, é objetivado a sua utilização de maneira facilitada, para isso, é proposto um novo ambiente computacional interativo, que provê funcionalidades intuitivas, que podem ser aplicadas ao ensino da geometria epipolar. Para isso, inicialmente, é feito um estudo dos temas relacionados à pesquisa. Posteriormente, a implementação de funções, em forma de um *toolbox* para Matlab. Esse provê várias implementações de algoritmos de métodos já bem conhecidos

na literatura. Então, é projetado um ambiente gráfico interativo que suporte a utilização dos métodos do *toolbox*.

1.1.1 Objetivos Específicos

- Pesquisar as principais ferramentas para Matlab aplicadas ao ensino e, particularmente, voltadas à visão computacional;
- Estabelecer uma relação de comparação entre as ferramentas pesquisadas ao ambiente computacional proposto neste trabalho;
- Fazer um estudo sobre geometria epipolar e suas implicações inerentes;
- Implementar os métodos para o processo de estimação da matriz fundamental, baseados em um *toolbox* para Matlab;
- Elaborar um ambiente interativo, utilizando-se da GUI (*Graphical User Interface*) do Matlab, que apoie o ensino da geometria epipolar, possibilitando a execução de todas as etapas envolvidas no processo, de forma bastante intuitiva, permitindo ao usuário escolher quais métodos (lineares, não-lineares e robustos) serão utilizados na estimação, processando gráficos comparativos para uma análise detalhada dos resultados.

1.2 Justificativas

Devido à complexidade para o entendimento da teoria associada à geometria epipolar, há grande dificuldade, principalmente por iniciantes no estudo da visão computacional, para compreenderem as técnicas utilizadas no cálculo e extração de resultados da matriz fundamental. No entanto, algumas ferramentas que facilitam essa abstração são disponibilizadas por vários autores conhecidos na literatura (CORKE, 2005, 1996; ASTROM et al., 1997; KANWAL; ARIF; MAJEED, 2003; SAMPER et al., 2010; MARIOTTINI; PRATTICHIZZO, 2005; SCARAMUZZA; SIEGWART, 2007). No entanto, a maioria delas deixam algumas lacunas que são exploradas neste trabalho, tais como, a possibilidade de manipulações de dados reais e sintéticos, para uma simulação do cálculo da matriz fundamental, por meio dos principais métodos propostos na literatura (HARTLEY, 1997; TRAJKOVIC; HEEDLEY, 1997; LIU; MANNER, 2003; ZHANG; LOOP, 2001; BARTOLI; STURM, 2004; HARTLEY; ZISSERMAN, 2003; FISCHLER; BOLLES, 1981; TORR; ZISSERMAN, 1998; ROUSSEEUW, 1987), o que possibilita a confrontação desses métodos. Além disso,

o ambiente interativo permite uma análise detalhada dos dados processados, por meio de resultados gerados em gráficos e de forma textual.

1.3 Estrutura da Dissertação

Esta dissertação de mestrado consiste em cinco capítulos e um anexo, assim organizados: no Capítulo 1, é feita a introdução do trabalho, uma visão geral dos objetivos e principais características do contexto do problema. No Capítulo 2, é demonstrada a fundamentação teórica, no qual são discutidos os principais conceitos necessários ao desenvolvimento deste trabalho. No Capítulo 3, a construção do ambiente interativo computacional, proposto neste trabalho, é detalhada. No Capítulo 4, são discutidos os resultados de simulações, no ambiente computacional proposto, com dados sintéticos e reais. No Capítulo 5, é feita a conclusão e discussão sobre trabalhos futuros. Por fim, no Anexo A, é apresentado o artigo para publicação, no qual os resultados do trabalho realizado são apresentados.

2 Fundamentação Teórica

Este capítulo limita-se à apresentação dos principais conceitos teóricos necessários ao desenvolvimento deste trabalho. Inicia-se com a definição da principal ferramenta utilizada para o desenvolvimento do projeto, o Matlab. Também é apresentada a revisão bibliográfica relacionada aos principais pacotes e ferramentas educacionais desenvolvidas no Matlab e as principais aplicadas à visão computacional. Na sequência do texto, são apresentados os principais conceitos relacionados ao modelo da câmera. Enfim, os assuntos relacionados à visão binocular aplicados à visão computacional são evidenciados.

2.1 Matlab

De acordo com Mathworks (2010), o Matlab (*Matrix Laboratory*) é um programa computacional interativo, com alta capacidade para resolução de problemas voltados a criação de modelos matemáticos. Entre outras características, integra análise numérica, cálculo com matrizes, processamento de sinais e construção de poderosos gráficos. É muito popular no meio acadêmico, e devido a isso, inúmeros trabalhos científicos são realizados com o apoio desse software, nos mais diversos campos de aplicação, inclusive nas áreas educacionais. O Matlab possui uma linguagem de programação de alto nível, similar à linguagem C, na qual é possível criar rotinas de programas de forma modularizada, possibilitando uma programação de qualidade. É possível construir projetos com programação híbrida, utilizando-se de outras linguagens de programação, normalmente, quando se exige um poder de processamento mais eficiente. Um exemplo é a utilização de módulos de programação na linguagem C ou FORTRAN, por meio das funções MEX (Matlab EXternal file). Alguns processamentos, como laços de repetição, podem tomar muito tempo de execução no Matlab e se tornar “gargalos”. Nesses casos, parte do código em Matlab pode ser reescrito na linguagem C ou FORTRAN e ser chamado dentro do algoritmo.

O Matlab possui uma ferramenta que permite construir interfaces gráficas.

Nela, são disponibilizados vários componentes. Isso contribui para o desenvolvimento de soluções com alta interatividade, facilitando a manipulação da aplicação pelo usuário final. Entretanto, como a construção de interfaces gráficas não é o foco do projeto Matlab, observa-se que há uma deficiência na variedade de componentes para a criação da interface gráfica, principalmente, quando comparada a outras linguagens de programação (C++, Java, VB.NET). Contudo, supre a necessidade básica para se ter uma interatividade razoável.

Os engenheiros de software da MathWorks fizeram um trabalho muito louvável em adicionar capacidade orientada a objetos para Matlab. Seu modelo de objeto é perfeitamente coerente com todos os requisitos básicos de programação orientada a objetos. O Matlab possibilita realizar polimorfismo, encapsulamento e tem capacidade para realizar heranças entre objetos. A maior diferença entre o Matlab e as mais tradicionais linguagens de programação orientadas a objeto, resulta de uma das propriedades fundamentais do Matlab, as variáveis sem tipo. A falta de tipificação forte de variáveis representa uma desvantagem, pois não é possível detectar todo de tipo de erro, relacionado ao tipo de variável, em tempo de interpretação do código fonte da aplicação.

A GUI (*Graphical User Interface*) do Matlab é baseada em orientação a objetos. Esses objetos são disponibilizado em forma de componentes, tornando os programas mais fáceis de usar, pois fornecem uma aparência consistente e com controles intuitivos. Dentre os controles mais importantes, destacam-se: botões, régua, caixas de listagem e menus. A sua utilização depende de um projeto de interface gráfica de qualidade, que deve ser previsível e compreensível, para que o usuário tenha facilidade em operar o sistema.

Embora o Matlab seja uma ferramenta bem conhecida, é necessário pagar por licença para sua utilização. Uma alternativa gratuita é utilizar o Octave ou Scilab. No entanto, não possui compatibilidade com a GUI do Matlab, para isso, deve-se fazer grandes adaptações no código para a devida migração.

Com base nas informações anteriores, para a construção do Ambiente Interativo para estimar a matriz fundamental proposto neste trabalho, utilizou-se do recurso para geração de interface gráfica GUI do Matlab. Evidentemente, sua utilização permitiu a construção de uma solução mais agradável, de fácil entendimento e alta interatividade, o que facilitou a demonstração das etapas e apresentação dos resultados ao usuário.

2.1.1 Pacotes Educacionais para Matlab

A estrutura do Matlab pode ser facilmente estendida por meio da inclusão de funções correlacionadas e agrupadas em pacotes (*toolboxes*). De fato, existem inúmeros pacotes disponíveis comercialmente e outros com licenças livres. Por conta disso, ao longo de vários anos, o Matlab tem sido utilizado com sucesso no ensino de diversas áreas da engenharia, devido a grande disponibilidade de ferramentas e um excelente ambiente de simulação, em conjunto com uma poderosa linguagem de programação, que permite a resolução de problemas, principalmente matemáticos, de forma intuitiva. Como exemplo, em controle de sistemas, Uran e Jezernik (2008) propuseram um interessante laboratório virtual, baseado no *Matlab Web Server* (MWS), para experimentos de projetos de controle de sistemas, que consistem de dois laboratórios virtuais, que possibilita simulações de forma intuitiva pelos estudantes.

Nos trabalhos de Knudsen (2006), Teixeira, Assuncao e Covacic (2007) e Aliane (2010), os autores não se preocuparam tanto com a interatividade, entretanto, os pacotes ferramentais satisfazem as necessidades por meio da precisão dos resultados. Em processamento digital de sinais, destaca-se o trabalho proposto por Cavicchi (2005), no qual o conjunto de ferramentas interage com dados de entrada e produz os resultados em um ambiente gráfico interativo. No campo do eletromagnetismo, Magistris (2005) desenvolveu um laboratório virtual, baseado em Matlab, como um ambiente de simulação por meio de uma interface Web, no qual o aluno acessa a aplicação pelo navegador de internet. Essa ferramenta foca a experimentação de assuntos introdutórios para cursos de graduação. Também, Sagnard (2004) propôs o desenvolvimento de um software educacional baseado em três interfaces gráficas do Matlab. Permite a visualização dos fenômenos de propagação eletromagnética e análise de experimentais em nível de graduação. Em cada interface, permite-se ao aluno definir os parâmetros associado a um determinado estudo e executar simulações em tempo real.

Para o ensino apoiado por pacotes do Matlab na área de máquinas elétricas, destaca-se o trabalho de Ayasun e Nwankpa (2005) que descreve a execução de três testes de motor de indução, utilizando o Matlab/Simulink. Para tanto, foram criados modelos de simulação desenvolvidos para suportar e reforçar o ensino de máquinas elétricas em nível de graduação. Em robótica, o trabalho apresentado por Pota (1992) utiliza-se do Matlab/Simulink para efetuar as simulações do método de ensino proposto. Na eletrônica de potência, destaca-se uma ferramenta de software para a aprendizagem do comportamento dinâmico dos circuitos, desen-

volvida por Chau (1996), focando os estudantes de graduação. Essa ferramenta incorpora o mérito de dois conhecidos pacotes de software, ou seja, o PSpice e o processo de identificação paramétrica de Matlab. Portanto, sem passar por matemática complicada, os alunos podem facilmente simular um projeto.

2.1.2 Pacotes de Funções para Visão Computacional

Como já discutido, motivados por trabalhos como os mencionados anteriormente, o enfoque do presente trabalho é usar o Matlab para ensino de temas relacionados à visão computacional. Evidentemente, diversos autores já escreveram trabalhos com esse perfil. Os principais encontrados na literatura são descritos a seguir.

- Corke (2005, 1996): são coleções de funções que têm sido desenvolvidas por mais de uma década e abrangem algoritmos desde processamento de imagens até manipulação de braços robóticos. Além disso, as funções são versáteis o suficiente para serem utilizadas em aplicações de tempo real. No entanto, a sua utilização no controle de manipuladores é muito dependente do hardware usado pelo autor. Além disso, para utilizar o pacote, é exigido do usuário um elevado nível de conhecimento prévio. Por isso, eles não são indicados para uso na educação.
- Astrom et al. (1997): apresentam diferentes instrumentos, aplicados em Matlab, que realiza uma série de tarefas de visão computacional. As principais características do conjunto de ferramenta proposto é a extração de pontos característicos (cantos) em uma imagem, extração de curvas de borda em uma imagem, cálculo de coordenadas de pontos 3D, cálculo da geometria multi-câmeras, autocalibração de câmeras, entre outros. Contudo, a sua utilização é dada em modo texto, ou seja, por meio da chamada de funções na linha de comando do Matlab.
- Kanwal, Arif e Majeed (2003): disponibilizaram um conjunto de funções relacionadas à visão binocular, no qual imagens são capturadas, pontos são segmentados, a matriz fundamental é estimada e até mesmo uma reconstrução euclideana de um objeto é realizada após a calibração do conjunto binocular. Contudo, trata-se apenas de uma coleção de algoritmos, mas possui algumas limitações: a) não há nenhuma interface gráfica com o usuário que torne o uso das funções, bem como a sua sequência correta de execução; b) existe apenas um único algoritmo para cada problema proposto (o que torna difícil a percepção da importância de determinados aspectos do có-

digo devido a dificuldade de visualizar a solução do problema sem eles), e; c) o resultado da reconstrução é apresentado em um aplicativo externo ao Matlab, baseado no padrão VRML (*Virtual Reality Modeling Language*).

No que se refere, especificamente, a pacotes de funções para ambiente educacional voltados à visão computacional, destacam-se:

- Samper et al. (2010): propuseram um aplicativo computacional, denominado Metrovisionlab, implementado como um conjunto de ferramentas para o Matlab. Tem como finalidade o ensino de calibração de câmeras. O seu desenvolvimento foi focado ao ensino de graduação. Especificamente, a aplicação simula uma câmera virtual, possibilitando a configuração e fornecendo os dados que são influenciados no ambiente. É possível gerar dados sintéticos para calibração. Assim, o objetivo principal é caracterizar a precisão, repetibilidade, mecanismos de erro, influências de diferentes condições de medição e algoritmos de calibração da câmera. Nos testes realizados, o *software* tem demonstrado ser muito eficaz, além de possuir uma interface, em modo gráfico, muito amigável e intuitiva. Contudo, essa ferramenta não permite a comparação entre os principais métodos para calibração de câmeras e não aborda de forma completa e educativa o cálculo da matriz fundamental.
- Mariottini e Prattichizzo (2005): desenvolveram um conjunto de ferramentas para o Matlab, denominada EGT (*Epipolar Geometry Toolbox*), com a finalidade de permitir simulação de múltiplas câmeras, bem como para a manipulação da informação visual e da geometria entre elas. Permite simular aplicações para câmeras modelo *pinhole* (será abordado a seguir) e para sensores de visão panorâmica. Dentre as principais características, destacam-se as que tratam o posicionamento das câmeras e visualização, os cálculos para estimativas dos parâmetros das câmeras e o cálculo para estimar a geometria epipolar. Uma outra característica interessante da EGT é a compatibilidade com a *Robotics Toolbox* (CORKE, 1996), possibilitando simulações de servo-visão. Embora, o foco principal seja o cálculo da matriz fundamental, não é possível uma análise entre vários métodos para tal, uma vez que somente é utilizado o método dos 8 Pontos (HARTLEY, 1997) e um iterativo baseado na distância geométrica (SAMPSON, 1982). Ressalta-se também que o conjunto de ferramentas EGT não possui um ambiente de simulação interativo, o usuário necessita chamar as funções por linha de comando, ou seja, somente pelo modo texto do Matlab.

- Scaramuzza e Siegwart (2007): propuseram um *toolbox* para calibração de câmeras omnidirecionais ¹, denominada OcamCalib, que possui um ambiente gráfico interativo simples. A principal finalidade é permitir que qualquer usuário, com facilidade, possa rapidamente calibrar sua própria câmera omnidirecional. As características mais notáveis desse conjunto de ferramentas são os seguintes: a) capacidade de calibrar os diferentes tipos de câmeras Omnidirecional, sem qualquer conhecimento sobre os parâmetros da câmera; b) detecção automática do centro; c) *feedback* visual sobre a qualidade dos resultados de calibração, reprojando os pontos 3D; d) seleção dos cantos da imagem de entrada assistida por um algoritmo de detecção de cantos. Contudo, não permite a utilização de vários métodos para calibração, nem mesmo calcular a matriz fundamental de um conjunto de câmeras.

Além dos pacotes supracitados, também há uma grande demanda, de estudantes e pesquisadores em visão computacional, por pacotes disponibilizados pela Mathworks, tais como, *Image Processing Toolbox*, *Image Acquisition Toolbox* e *Video and Image Processing Blockset* (MATHWORKS, 2010).

É notável que nenhum dos trabalhos pesquisados tem como característica o ensino da geometria epipolar, principalmente, no que se diz respeito a simulação dos principais métodos para estimar a matriz fundamental e que possibilite uma análise crítica dos resultados. Essa, por sinal, foi a principal motivação do presente trabalho. Além disso, para auxiliar no entendimento do trabalho, nas próximas seções, são apresentados alguns conceitos importantes, relacionados ao tema abordado.

2.2 O Modelo da Câmera

O modelo de câmera utilizado neste trabalho é o *pinhole*. Nesse, um raio de luz, proveniente de um ponto \mathbf{M} no espaço, passa através de um orifício (localizado no ponto \mathbf{C}) e incide em um plano no ponto \mathbf{m} , de acordo com a figura 2.1. O plano de incidência dos raios é chamado de “plano de imagem” e o ponto \mathbf{C} é chamado de “centro da câmera”. A distância do plano de imagem ao ponto \mathbf{C} é um parâmetro intrínseco da câmera e é conhecido como “distância focal” e, geralmente, é representado por f . Na figura 2.2, a câmera é inserida em um sistema de coordenadas. Esse é conhecido como “sistema de coordenadas

¹Câmeras omnidirecionais produzem imagens de 360° do ambiente, podendo ser utilizados, entre outras aplicações, em navegação, vigilância remota e em robôs.

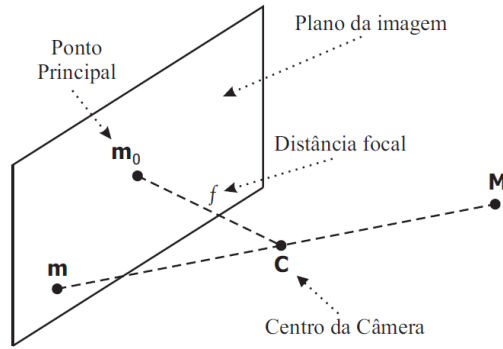


Figura 2.1: Esboço de uma câmera *pinhole*.

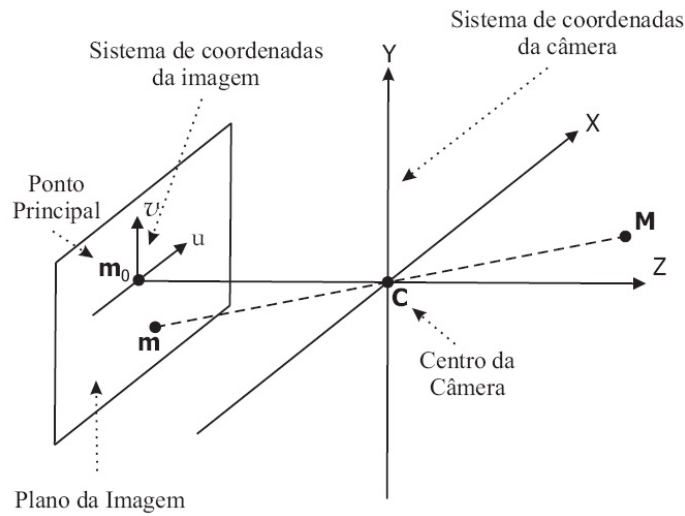


Figura 2.2: Coordenadas do modelo de câmera *pinhole*

da câmera” e tem sua origem no ponto C . Além disso, um outro sistema de coordenadas, com origem em \mathbf{m}_0 , é representado no plano da imagem. Esse é chamado “sistema de coordenadas da imagem”. Por sua vez, o ponto \mathbf{m}_0 , chamado “centro da imagem”, é a projeção ortogonal do ponto C , ou seja, o ponto no qual o eixo Z corta o plano de imagem.

É importante conhecer a relação entre um ponto \mathbf{M} no sistema de coordenadas da câmera e sua projeção, \mathbf{m} , no sistema de coordenadas da imagem. Isto pode ser conseguido facilmente por meio de semelhanças de triângulos. Por exemplo, por meio da figura 2.3, vê-se que a ordenada de \mathbf{m} é $u = fY/Z$. De maneira semelhante, pode-se estender o raciocínio para a abscissa de \mathbf{m} , logo $\mathbf{m} = (fX/Z, fY/Z)$. A equação anterior relaciona as coordenadas dos pontos \mathbf{M} e \mathbf{m} de forma não-linear. Para contornar esse problema, podemos utilizar uma

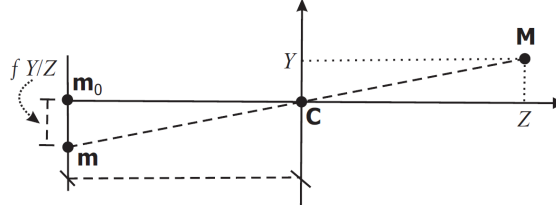


Figura 2.3: Determinação da ordenada do ponto \mathbf{m} em função das coordenadas do ponto \mathbf{M}

equação homogênea, ou seja,

$$\begin{bmatrix} fX \\ fY \\ Z \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \quad (2.1)$$

A equação (2.1) é reescrita frequentemente como

$$s\tilde{\mathbf{m}} \simeq \mathbf{A} [\mathbf{I} \mathbf{0}_3] \tilde{\mathbf{M}}, \quad (2.2)$$

onde s é uma constante não-nula, $\mathbf{0}_3$ é um vetor nulo de 3 elementos, \mathbf{I} é a matriz identidade. O símbolo “ \simeq ” representa que os dois lados da equação podem diferir por uma constante (trata-se de uma equação homogênea) e,

$$\mathbf{A} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

concentra os “parâmetros intrínsecos da câmera”.

A matriz \mathbf{A} , dada em (2.3), é um modelo muito simplificado de uma câmera. Na prática, utiliza-se um modelo mais geral, definido por

$$\mathbf{A} = \begin{bmatrix} \alpha & s_k & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (2.4)$$

onde α e β são fatores de escala para, respectivamente, os eixos x e y , $\mathbf{m}_0 = [u_0, v_0]^T$ é o ponto central da câmera e s_k está relacionado ao ângulo que corrige a ortogonalidade da matriz de sensores. Contudo, atualmente é possível fabricar câmeras com s_k muito próximo de zero. Por isto, neste trabalho, considera-se $s_k = 0$. Dessa forma, há apenas quatro parâmetros intrínsecos a serem estimados.

As coordenadas do ponto \mathbf{M} podem ser fornecidas em relação a um referencial diferente do sistema de coordenadas da câmera. Quando isto acontece, o referencial em questão é dito ser o “sistema de coordenadas do ambiente” e a equação (2.2) deve ser reescrita como sendo

$$\tilde{\mathbf{m}} \simeq \mathbf{A}[\mathbf{R} \ \mathbf{t}]\widetilde{\mathbf{M}}, \quad (2.5)$$

onde \mathbf{R} é uma matriz 3×3 que representa a rotação da câmera em relação ao sistema de coordenadas do ambiente, \mathbf{t} é um vetor que equivale a translação do centro da câmera com relação ao sistema de coordenadas do ambiente. A matriz antissimétrica $[\mathbf{t}]_{\times}\mathbf{R}$ possui toda informação a respeito dos “parâmetros extrínsecos da câmera”.

Frequentemente, define-se a “matriz de projeção” de uma câmera como sendo

$$\mathbf{P} \simeq \mathbf{A}[\mathbf{R} \ \mathbf{t}]. \quad (2.6)$$

Nesse caso, a equação (2.5) torna-se

$$\tilde{\mathbf{m}} \simeq \mathbf{P}\widetilde{\mathbf{M}}. \quad (2.7)$$

Em um sistema de visão binocular, frequentemente, assume-se que a câmera da esquerda está na origem do sistema de coordenadas do ambiente. Então, considerando a equação (2.2), tem-se

$$\tilde{\mathbf{m}}_1 \simeq \mathbf{A}_1[\mathbf{I} \ \mathbf{0}]\widetilde{\mathbf{M}} \text{ e } \tilde{\mathbf{m}}_2 \simeq \mathbf{A}_2[\mathbf{R} \ \mathbf{t}]\widetilde{\mathbf{M}}, \quad (2.8)$$

onde \mathbf{R} representa a rotação² da câmera 1 em relação a 2, \mathbf{t} equivale a translação³ do centro da câmera 1 com relação a 2.

A equação anterior muitas vezes é expressa em função das matrizes de projeção das câmeras. Nesse caso,

$$\mathbf{P}_1 \simeq \mathbf{A}_1[\mathbf{I} \ \mathbf{0}] \text{ e } \mathbf{P}_2 \simeq \mathbf{A}_2[\mathbf{R} \ \mathbf{t}] \quad (2.9)$$

e o processo de calibração do conjunto binocular consiste em estimar as matrizes \mathbf{P}_1 e \mathbf{P}_2 .

²Rotação é a transformação do sistema de coordenadas da câmera 1 em relação a 2.

³Translação é o deslocamento paralelo em linha reta de um objeto da câmera 1 em relação a 2

2.3 Visão Binocular

O principal objetivo da visão binocular é analisar o problema da reconstrução da informação tridimensional de objetos, a partir de um par de imagens capturadas simultaneamente, mas, com um pequeno deslocamento de uma em relação a outra. Tal deslocamento, produz diferenças entre as imagens. Essas diferenças, permite se ter a percepção tridimensional.

Dadas as coordenadas de um ponto \mathbf{M} , a equação (2.7) permite encontrar a sua projeção \mathbf{m} no plano de imagem. Contudo, tal processo não é reversível, ou seja, dado o ponto \mathbf{m} , é impossível encontrar as coordenadas do ponto \mathbf{M} unicamente. Isto ocorre porque qualquer ponto sobre a reta $\overline{\mathbf{CM}}$, conforme observado na figura 2.4, tem a projeção no mesmo ponto \mathbf{m} . Nesta seção, apresenta-se uma discussão sobre geometria epipolar e matriz fundamental, que são teorias associadas à visão binocular, nas quais permitem contornar o problema apresentado anteriormente.

2.3.1 Geometria Epipolar

Se os parâmetros intrínsecos e extrínsecos das câmeras são conhecidos, para um sistema binocular, o sistema é dito calibrado e, por meio das correspondências de pontos e utilizando a técnica de triangulação, a informação 3D pode ser completamente recuperada. Contudo, em um sistema não-calibrado, a única informação disponível é a geometria epipolar. A geometria epipolar é a geometria de projeção intrínseca entre duas visões. É independente de estrutura de cena e só depende dos parâmetros internos das câmeras e pose relativa (HARTLEY; ZISSERMAN, 2003). A matriz fundamental \mathbf{F} encapsula essa geometria intrínseca.

Se um ponto \mathbf{M} em um espaço 3D é projetado como \mathbf{m}_1 na primeira câmera e \mathbf{m}_2 pela segunda, então os pontos da imagem satisfazem a relação desenvolvida por meio da equação (2.10).

$$\tilde{\mathbf{m}}_2^T \mathbf{F} \tilde{\mathbf{m}}_1 = 0, \quad (2.10)$$

A matriz fundamental pode ser construída do conjunto de correspondências de imagens, sem requerer conhecimento prévio dos parâmetros internos das câmeras ou posição relativa. Se o sistema está calibrado, é fácil obter a matriz fundamental. Contudo, mesmo na ausência de calibração, tal matriz facilita uma série de tarefas como, por exemplo, correspondência binocular, retificação de ima-

gens e até mesmo calibração de câmeras. Nesse caso, a matriz fundamental deve ser estimada a partir de um conjunto de correspondências de pontos que, por sua vez, necessita da identificação de cantos ou bordas das imagens.

A restrição epipolar consiste em considerar duas imagens comum a cena. Basicamente, a geometria epipolar pode ser entendida se for considerado o caso de duas câmeras como apresentado na figura 2.4. Nesta, \mathbf{C}_1 e \mathbf{C}_2 são, respectivamente, os centros ópticos da primeira e segunda câmeras. Então, dado um ponto \mathbf{m}_1 na imagem I_1 , o ponto correspondente \mathbf{m}_2 na imagem I_2 está restrito a uma reta chamada “reta epipolar” do ponto \mathbf{m}_1 , representada na figura 2.4 por \mathbf{l}_2 . A reta \mathbf{l}_2 é a intersecção do plano π , definido por \mathbf{M} , \mathbf{C}_1 e \mathbf{C}_2 (chamado de plano epipolar), com o plano I_2 . Isto acontece porque o ponto \mathbf{m}_1 pode corresponder a qualquer ponto da reta $\overline{\mathbf{C}_1\mathbf{M}}$ e a projeção de $\overline{\mathbf{C}_1\mathbf{M}}$ em I_2 é a reta \mathbf{l}_2 . Além

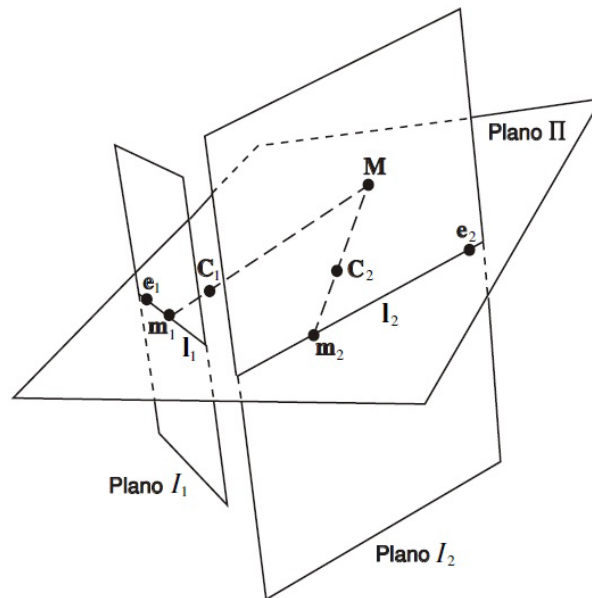


Figura 2.4: Geometria Epipolar

disso, observa-se que todas as retas epipolares dos pontos de I_1 passam através de um ponto comum, \mathbf{e}_2 , em I_2 . Esse ponto é conhecido como “epipolo”. O ponto \mathbf{e}_1 é a intersecção da reta $\overline{\mathbf{C}_1\mathbf{C}_2}$ com o plano I_2 . Finalmente, pode-se facilmente observar a simetria da geometria epipolar. O correspondente em I_1 de cada ponto \mathbf{m}_{2i} , sobre a reta \mathbf{l}_{2i} , precisa pertencer a reta epipolar \mathbf{l}_{1i} , que é a intersecção do mesmo plano π_i com o plano I_1 . Todas as retas epipolares formam um conjunto contendo o epipolo \mathbf{e}_1 , que é a intersecção da reta $\overline{\mathbf{C}_1\mathbf{C}_2}$ com o plano I_1 .

2.3.2 Matriz Fundamental

Na visão computacional, de acordo com Luong e Faugeras (1996), a matriz fundamental é uma matriz de posto 2, que relaciona pontos correspondentes de um par de imagens distintas. A matriz fundamental, \mathbf{F} , tem sete graus de liberdade, então, são necessários, no mínimo, sete correspondências de pontos para estimá-la.

Para um sistema de câmeras não calibradas (os parâmetros intrínsecos de cada câmera estão inseridos em \mathbf{F}), a matriz que representa a geometria entre ambas é denominada fundamental, também chamada de “tensor bifocal”. Já para um conjunto calibrado, a matriz é denominada essencial. A matriz essencial, \mathbf{E} , tem a restrição de seu determinante ser nulo, $\det(\mathbf{E}) = 0$ e contém cinco parâmetros. Já a matriz fundamental também tem determinante nulo, $\det(\mathbf{F}) = 0$, seu posto é 2 e possui sete parâmetros.

A restrição epipolar é útil quando deseja-se realizar correspondência de pontos entre duas imagens, ou seja, dado um ponto $\mathbf{m}_1 \in I_1$, sabe-se que o seu correspondente $\mathbf{m}_2 \in I_2$ está sobre a reta $\mathbf{l}_2 = \mathbf{F}\tilde{\mathbf{m}}_1$. Isto reduz o problema de duas para apenas uma única dimensão (ALVES, 2007).

Observa-se, na figura 2.4, que a reta \mathbf{l}_2 pode ser completamente definida pelo epipolo \mathbf{e}_2 e qualquer projeção em I_2 de um ponto sobre $\overline{\mathbf{C}_1\mathbf{M}}$. Em particular, um ponto sobre $\overline{\mathbf{C}_1\mathbf{M}}$ pode ser expresso por $\mathbf{P}_1^+\tilde{\mathbf{m}}_1$, cuja projeção em I_2 é dada por $\mathbf{P}_2\mathbf{P}_1^+\tilde{\mathbf{m}}_1$, onde \mathbf{P}_1^+ é a pseudoinversa da matriz \mathbf{P}_1 , ou seja, $\mathbf{P}_1^+ = (\mathbf{P}_1^T\mathbf{P}_1)^{-1}\mathbf{P}_1^T$. Dessa forma,

$$\begin{aligned}\mathbf{l}_2 &= \tilde{\mathbf{e}}_2 \times (\mathbf{P}_2\mathbf{P}_1^+\tilde{\mathbf{m}}_1) \\ &= [\tilde{\mathbf{e}}_2]_{\times}(\mathbf{P}_2\mathbf{P}_1^+)\tilde{\mathbf{m}}_1,\end{aligned}\tag{2.11}$$

onde $[\tilde{\mathbf{e}}_2]_{\times}$ é matriz antissimétrica do vetor $\tilde{\mathbf{e}}_2$.

A equação anterior relaciona \mathbf{m}_1 e \mathbf{l}_2 matematicamente. Normalmente, a matriz

$$\mathbf{F} = [\tilde{\mathbf{e}}_2]_{\times}(\mathbf{P}_2\mathbf{P}_1^+)\tag{2.12}$$

é chamada de “matriz fundamental”.

A equação (2.11) mostra que, para cada ponto $\mathbf{m}_1 \in I_1$, existe uma reta epipolar, \mathbf{l}_2 , na imagem I_2 , ou seja, há uma correspondência entre pontos na

primeira imagem e retas epipolares na segunda imagem.

Partindo da equação (2.12), a matriz fundamental também pode ser expressa em termos dos parâmetros intrínsecos e extrínsecos do conjunto binocular. Esse fato é resumido, de acordo com Alves (2007), como segue.

A matriz fundamental de um conjunto binocular, cujas matrizes de projeção são $\mathbf{P}_1 = \mathbf{A}_1[\mathbf{I} \ \mathbf{0}]$ e $\mathbf{P}_2 = \mathbf{A}_2[\mathbf{R} \ \mathbf{t}]$, é dada por

$$\mathbf{F} = \mathbf{A}_2^{-T}[\mathbf{t}]_{\times} \mathbf{R} \mathbf{A}_1^{-1} \quad (2.13)$$

O epipolo \mathbf{e}_2 é a projeção do ponto \mathbf{C}_1 no plano de imagem I_2 , ou seja,

$$\begin{aligned} \tilde{\mathbf{e}}_2 &= \mathbf{P}_2 \tilde{\mathbf{C}}_1 = \mathbf{A}_2[\mathbf{R} \ \mathbf{t}] \begin{bmatrix} 0 \\ 1 \end{bmatrix} = [\mathbf{A}_2 \mathbf{R} \ \mathbf{A}_2 \mathbf{t}] \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ &= \mathbf{A}_2 \mathbf{t}. \end{aligned} \quad (2.14)$$

Além disso, como já discutido, $\mathbf{P}_1^+ = (\mathbf{P}_1^T \mathbf{P}_1)^{-1} \mathbf{P}_1^T$, ou seja,

$$\begin{aligned} \mathbf{P}_1^+ &= \left(\begin{bmatrix} \mathbf{A}_1^T \\ \mathbf{0}^T \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{A}_1^T \\ \mathbf{0}^T \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{A}_1^{-1} \mathbf{A}_1^{-T} & \mathbf{0} \\ \mathbf{0}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{A}_1^T \\ \mathbf{0}^T \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{A}_1^{-1} \\ \mathbf{0}^T \end{bmatrix}. \end{aligned}$$

Desta forma,

$$\begin{aligned} \mathbf{P}_2 \mathbf{P}_1^+ &= \mathbf{A}_2[\mathbf{R} \ \mathbf{t}] \begin{bmatrix} \mathbf{A}_1^{-1} \\ \mathbf{0}^T \end{bmatrix} = [\mathbf{A}_2 \mathbf{R} \ \mathbf{A}_2 \mathbf{t}] \begin{bmatrix} \mathbf{A}_1^{-1} \\ \mathbf{0}^T \end{bmatrix} \\ &= \mathbf{A}_2 \mathbf{R} \mathbf{A}_1^{-1}. \end{aligned} \quad (2.15)$$

Agora, substituindo (2.14) e (2.15) em (2.12), tem-se

$$\begin{aligned} \mathbf{F} &= [\mathbf{A}_2 \mathbf{t}]_{\times} \mathbf{A}_2 \mathbf{R} \mathbf{A}_1^{-1} \\ &= \mathbf{A}_2^{-T}[\mathbf{t}]_{\times} \mathbf{A}_2^{-1} \mathbf{A}_2 \mathbf{R} \mathbf{A}_1^{-1} \\ &= \mathbf{A}_2^{-T}[\mathbf{t}]_{\times} \mathbf{R} \mathbf{A}_1^{-1}. \end{aligned}$$

2.3.3 Estimação da Matriz Fundamental

Quando o sistema está calibrado, a matriz fundamental, \mathbf{F} , já é conhecida. Entretanto, mesmo na ausência de calibração é possível sua estimação. Para tanto, deve ser obtida a partir de um conjunto de correspondências de pontos.

Conforme já mencionado e de acordo com o proposto por Zhang (1998), dada uma correspondência ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$), pode-se associar uma reta epipolar, \mathbf{l}_{2i} , dada por $\mathbf{l}_{2i} = \mathbf{F}\tilde{\mathbf{m}}_{1i}$, ao ponto \mathbf{m}_{2i} . Idealmente, \mathbf{m}_{2i} está sobre \mathbf{l}_{2i} , ou seja, $d(\mathbf{m}_{2i}, \mathbf{l}_{2i}) = 0$. Contudo, na prática, devido a ruídos nas coordenadas dos pontos ou a uma estimação “ruim” da matriz fundamental, $d(\mathbf{m}_{2i}, \mathbf{l}_{2i}) \neq 0$. Além disso, quanto maior for essa distância, pior será o ajuste de \mathbf{F} a correspondência ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$). Assim, dadas n correspondências ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$) e uma matriz fundamental \mathbf{F}_j , tem-se

$$r^2(\mathbf{F}_j) = \frac{1}{2n} \sum_{i=0}^n [d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{F}_j \tilde{\mathbf{m}}_{1i}) + d^2(\tilde{\mathbf{m}}_{1i}, \mathbf{F}_j^T \tilde{\mathbf{m}}_{2i})], \quad (2.16)$$

para medir o ajuste de \mathbf{F}_j ao conjunto de correspondências, ou seja, $r^2(\mathbf{F}_j)$ é uma medida da qualidade de \mathbf{F}_j com relação ao conjunto de correspondências ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$). No presente trabalho, esta medida é utilizada para avaliar os diversos métodos de estimação da matriz fundamental, os quais são analisados a seguir. Esses métodos são classificados em três grupos: lineares, não-lineares e robustos.

Para um sistema binocular não calibrado, utilizando-se de imagens reais, o diagrama de blocos apresentado na figura 2.5 mostra os principais passos para se obter a matriz fundamental.

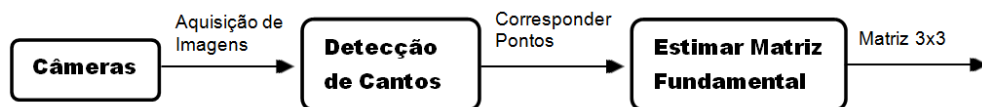


Figura 2.5: Diagrama de Blocos que Demonstra os Passos para Estimar a Matriz Fundamental

2.3.3.1 Métodos Lineares

A principal vantagem dos métodos lineares para cálculo da matriz fundamental é a relativa simplicidade. Esses métodos permitem utilizar redundância para reduzir a influência de ruídos e, se alguns cuidados forem tomados, fornecem uma boa estimação de \mathbf{F} a um custo computacional bastante reduzido.

No conjunto de ferramentas proposto nesse trabalho, foram implementados

os métodos de Hartley (1997) (8 Pontos), Trajkovic e Hedley (1997) e Liu e Maner (2003) (Minimização da Função Sampson). De uma forma geral, todos os métodos lineares propostos para estimar a matriz fundamental são baseados na equação (2.18). Contudo, na tentativa de melhorar ainda mais o desempenho do método dos 8 pontos, muitos outros autores propuseram modificações ao algoritmo. Chojnacki et al. (2003) e Torr P.; Fitzgibbon (2003) analisaram tal função e propuseram métodos semelhantes, baseados numa análise estatística do problema. Contudo, os métodos possuem desempenho muito similar ao algoritmo de 8 pontos normalizado, porém, o método de Hartley (HARTLEY, 1997) é bem mais simples e direto. Outro método popular, é o proposto por Boufama e Mohr (1998), denominado Paralaxe Virtual. O método é baseado na computação da homografia entre duas imagens de um plano virtual e na detecção do paralaxe de pontos não pertencentes a este plano. As aplicações práticas são praticamente nulas para o método paralaxe virtual, pois a matriz fundamental é muito dependente das transformações \mathbf{T}_1 e \mathbf{T}_2 .

Algoritmo dos 8 Pontos

Um mesmo ponto 3D de uma cena projetado no plano de projeção de duas câmeras distintas dá origem a um par de pontos correspondentes entre as duas imagens geradas. Sendo a matriz fundamental que caracteriza a geometria epipolar da cena, sabe-se que esses pontos obedecem à restrição epipolar, de acordo com a equação (2.10). Dessa forma, dado um conjunto de correspondências de pontos ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$), onde $\tilde{\mathbf{m}}_{1i} = [u_{1i}, v_{1i}, t_{1i}]^T$ e $\tilde{\mathbf{m}}_{2i} = [u_{2i}, v_{2i}, t_{2i}]^T$, pode-se reescrever a equação (2.10) na forma vetorial, ou seja,

$$\mathbf{u}_i^T \mathbf{f} = 0, \quad (2.17)$$

onde $\mathbf{u}_i = [u_{1i}u_{2i}, v_{1i}u_{2i}, u_{2i}, u_{1i}v_{2i}, v_{1i}v_{2i}, v_{2i}, u_{1i}, v_{1i}, 1]^T$ e $\mathbf{f} = [f_{11}, f_{12}, f_{13}, f_{21}, f_{22}, f_{23}, f_{31}, f_{32}, f_{33}]^T$, no qual f_{ij} é o elemento da matriz \mathbf{F} , sendo que i e j representam, respectivamente, linha e coluna.

Além disso, dado um conjunto de correspondências, tem-se o seguinte sistema linear a resolver

$$\mathbf{U}_n \mathbf{f} = 0, \quad (2.18)$$

onde $\mathbf{U}_n = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n]^T$.

Contudo, com $n \geq 8$ correspondências de pontos, pode-se encontrar \mathbf{f} resol-

vendo

$$\min_{\mathbf{f}} \mathbf{U}_n \mathbf{f}, \text{ sujeito a } \|\mathbf{f}\| = 1. \quad (2.19)$$

É bem sabido que a solução do problema (2.19) é o autovetor que corresponde ao menor autovalor de $\mathbf{U}_n^T \mathbf{U}_n$.

O método baseado na resolução da equação (2.19) é conhecido como “método dos 8 pontos” (HARTLEY, 1997). Tal método é a maneira mais simples de estimar a matriz fundamental a partir de um conjunto de correspondências de pontos. A desvantagem evidente do método é que esse não garante a restrição de posto dois. Assim, esta tem que ser imposta a posteriori. A forma mais conveniente de fazer isto é substituindo a matriz estimada, $\widehat{\mathbf{F}}$, pela matriz \mathbf{F} que minimiza $\|\widehat{\mathbf{F}} - \mathbf{F}\|$, sujeito a $\det \mathbf{F} = 0$. Isto é feito fazendo $\mathbf{F} = \widehat{\mathbf{F}}$ e substituindo o menor autovalor de \mathbf{F} por zero (FAUGERAS; LUONG, 2001). O resultado do algoritmo dos 8 pontos depende do sistema de coordenadas utilizado. Normalmente, as coordenadas dos pontos utilizados na solução do problema são dadas em relação ao canto superior esquerdo da imagem. Contudo, uma simples mudança nesse referencial torna o problema melhor condicionado e, portanto, conduz a um resultado mais estável. Essa mudança pode ser realizada por meio de transformações, \mathbf{T}_1 e \mathbf{T}_2 , aplicadas ao conjunto de correspondências, ou seja,

$$\widehat{\mathbf{m}}_1 = \mathbf{T}_1 \widetilde{\mathbf{m}}_1 \text{ e } \widehat{\mathbf{m}}_2 = \mathbf{T}_2 \widetilde{\mathbf{m}}_2,$$

isolando $\widetilde{\mathbf{m}}_1$ e $\widetilde{\mathbf{m}}_2$, tem-se

$$\widetilde{\mathbf{m}}_1 = \mathbf{T}_1^{-1} \widehat{\mathbf{m}}_1 \text{ e } \widetilde{\mathbf{m}}_2 = \widehat{\mathbf{m}}_2^T \mathbf{T}_2^{-T},$$

substituindo $\widetilde{\mathbf{m}}_1$ e $\widetilde{\mathbf{m}}_2$ na equação (2.10), tem-se

$$\widehat{\mathbf{m}}_2^T \mathbf{T}_2^{-T} \mathbf{F} \mathbf{T}_1^{-1} \widehat{\mathbf{m}}_1 = 0.$$

Então, essa relação implica que

$$\widehat{\mathbf{F}} = \mathbf{T}_2^{-T} \mathbf{F} \mathbf{T}_1^{-1},$$

onde $\widehat{\mathbf{F}}$ é a matriz fundamental das correspondências $\widehat{\mathbf{m}}_{1i} \leftrightarrow \widehat{\mathbf{m}}_{2i}$.

Hartley (1997), observou que essas transformações devem ser tais que, em cada uma das imagens, independentemente, o centróide do conjunto de pontos esteja na origem e a distância média de um ponto à origem seja $\sqrt{2}$. Esta mudança produz uma grande melhora na exatidão do algoritmo. Além disso, o custo computacional acrescentado ao problema, devido a essa normalização, é desprezível.

Trajkovic e Hedley

Este método também encontra \mathbf{F} resolvendo a equação (2.18). A diferença está na imposição da restrição de posto dois. O algoritmo de Trajkovic e Hedley (1997) impõe esta restrição calculando, inicialmente, o epipolo da matriz fundamental que resolve o problema (2.19). Em seguida, encontra \mathbf{F} que tem o mesmo epipolo e também tem posto dois. Segundo Trajkovic e Hedley (1997), este método tem performance superior ao de Hartley (1997), especialmente, quando há apenas um número pequeno de correspondências de pontos disponíveis para montar a matriz \mathbf{U}_n .

O algoritmo linear para estimar a matriz fundamental é tipicamente realizada em duas etapas: 1) Procurar a matriz fundamental \mathbf{F} que é a melhor solução de (2.18), utilizando a minimização linear dos mínimos quadrados e 2) Substituir \mathbf{F} pela matriz $\hat{\mathbf{F}}$, que minimiza a norma de Frobenius $\|\mathbf{F} - \hat{\mathbf{F}}\|$. O algoritmo linear proposto por Trajkovic e Hedley tem a mesma primeira etapa típica, mas a segunda etapa é diferente, ou seja, primeiro é calculado o epipolo na segunda imagem \mathbf{e}_2 como o autovetor correspondente ao autovalor mínimo da matriz \mathbf{F}^T e, em seguida, encontrar matriz $\hat{\mathbf{F}}$ que tem o mesmo epipolo e minimiza (2.18). É notável que $\hat{\mathbf{F}}$ tem o mesmo epipolo de $\hat{\mathbf{F}}$. Entretanto, $\hat{\mathbf{F}}$ é a melhor solução pois tem uma menor margem de erros.

Minimização da Função Sampson

É um método linear e iterativo proposto por Liu e Manner (2003), que estima a matriz \mathbf{F} que minimiza o “Erro de Sampson” (SAMPSON, 1982) da distância dos pontos às retas epipolares correspondentes. Contudo, faz-se uso da aproximação de primeira ordem para a função de custo geométrica e, portanto, converte o problema em uma minimização linear dos mínimos quadrados.

O método linear (2.19) não minimiza uma quantidade com significado geométrico. Uma idéia natural é, então, minimizar as distâncias entre os pontos e as suas correspondentes retas epipolares por meio de

$$\min_{\mathbf{f}} \sum_i d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{F}\tilde{\mathbf{m}}_{1i}),$$

onde $d(\cdot, \cdot)$ é dada por

$$d(\mathbf{m}_{2i}, \mathbf{l}_{2i}) = \frac{\tilde{\mathbf{m}}_{2i}^T \mathbf{l}_{2i}}{\sqrt{l_{21}^2 + l_{22}^2}} = \frac{1}{c_{2i}} \tilde{\mathbf{m}}_{2i}^T \mathbf{F} \tilde{\mathbf{m}}_{1i}, \quad (2.20)$$

no qual, $c_{2i} = \sqrt{l_{21}^2 + l_{22}^2}$. Sabe-se que o ponto \mathbf{m}_{2i} possui uma reta epipolar correspondente $\mathbf{l}_{2i} = \mathbf{F}\tilde{\mathbf{m}}_{1i}$, onde l_{21} e l_{22} são os dois primeiros elementos do vetor que representam essa reta.

O critério anterior determina apenas a reta epipolar na segunda imagem. Para evitar a inconsistência da geometria epipolar entre as duas imagens, pode-se minimizar o seguinte critério

$$\min_{\mathbf{f}} \sum_i (d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{F}\tilde{\mathbf{m}}_{1i}) + d^2(\tilde{\mathbf{m}}_{1i}, \mathbf{F}^T \tilde{\mathbf{m}}_{2i})), \quad (2.21)$$

que resolve, simultaneamente, o problema das duas imagens.

Como $\mathbf{l}_{2i} = \mathbf{F}\tilde{\mathbf{m}}_{1i} \equiv [l_{21}, l_{22}, l_{23}]^T$ e $\mathbf{l}_{1i} = \mathbf{F}^T \tilde{\mathbf{m}}_{2i} \equiv [l_{11}, l_{12}, l_{13}]^T$, usando (2.20) e o fato de que $\tilde{\mathbf{m}}_{2i}^T \mathbf{F}\tilde{\mathbf{m}}_{1i} = \tilde{\mathbf{m}}_{1i}^T \mathbf{F}^T \tilde{\mathbf{m}}_{2i}$, o critério (2.21) pode ser reescrito por

$$\min_{\mathbf{f}} \sum_i w_i^2 (\tilde{\mathbf{m}}_{2i}^T \mathbf{F}\tilde{\mathbf{m}}_{1i})^2, \quad (2.22)$$

tal que,

$$\begin{aligned} w_i &= \left(\frac{1}{l_{11}^2 + l_{12}^2} + \frac{1}{l_{21}^2 + l_{22}^2} \right)^{\frac{1}{2}} \\ &= \left(\frac{l_{11}^2 + l_{12}^2 + l_{21}^2 + l_{22}^2}{(l_{11}^2 + l_{12}^2)(l_{21}^2 + l_{22}^2)} \right)^{\frac{1}{2}}. \end{aligned} \quad (2.23)$$

A similaridade entre (2.22) e (2.18), leva-se a resolver o referido problema pela técnica linear dos mínimos quadrados. A equação linear correspondente pode ser multiplicada por w_i e o algoritmo de 8 pontos pode ser executado para estimar a matriz fundamental, o que minimiza (2.22). O problema é que os pesos w_i dependem da própria matriz fundamental. Para superar essa dificuldade, é então utilizado um método iterativo linear (LIU; MANNER, 2003). Primeiro, assume que todos os $w_i = 1$ executam o algoritmo dos 8 pontos para obter uma estimativa inicial da matriz fundamental. Os pesos w_i são, então, calculados a partir desta solução inicial. O papel da técnica linear dos mínimos quadrados é ponderar uma melhor solução. Este procedimento pode ser repetido várias vezes. Embora este algoritmo seja simples de implementar e minimiza a quantidade física, não há melhora significativa quando comparado com o método dos 8 pontos. Infelizmente, o método fornece bons resultados apenas quando as câmeras estão dispostas de forma que os epipolos estão próximos ao centro de imagem (FRANÇA, 2005).

2.3.3.2 Métodos Não-Lineares

A restrição de posto dois não pode ser embutida a métodos lineares. Assim, como discutido anteriormente, esta deve ser imposta após uma estimação inicial da matriz fundamental. Essa restrição, implica que as linhas e colunas da matriz devem ser linearmente dependentes. Além disso, geralmente, tais métodos minimizam critérios sem significado físico. Para contornar essas limitações, algoritmos não-lineares foram propostos. Geralmente, a resolução de tais algoritmos exige uma estimação inicial da matriz fundamental. Devido a seu bom desempenho, o método dos 8 pontos normalizado é bastante utilizado para esta tarefa. Sua principal função é fornecer uma solução inicial bem próxima de um mínimo local. Dessa forma, a solução dos problemas não-lineares é atingida após poucas iterações. Dentre os algoritmos não-lineares mais importantes para estimar a matriz fundamental, estão os propostos por Csurka et al. (1997), Wu e Yu (2005), Faugeras e Luong (2001), Zhang e Loop (2001), Bartoli e Sturm (2004) e Hartley e Zisserman (2003) (método da máxima verossimilhança).

Como discutido anteriormente, os métodos lineares não conseguem impor a restrição de posto dois para a matriz fundamental, no entanto, essa pode ser escrita em função dos epipolos e da homografia epipolar. Então, uma abordagem para impor a restrição de posto dois à matriz fundamental é parametrizá-la, uma vez que a matriz fundamental possui sete graus de liberdade. Isso é devido a matriz ser homogênea, então, uma das suas colunas e linhas devem ser linearmente dependentes de forma que a matriz tenha posto 2. Para tanto, normalmente são utilizados métodos não-lineares que parametrizam a matriz. Os trabalhos de Luong e Faugeras (1996), Zhang e Loop (2001) e Bartoli e Sturm (2004) utilizam de parametrização para impor o posto dois da matriz fundamental.

Os elementos da matriz fundamental são referenciados individualmente. Contudo, essa matriz pode ser escrita em função dos epipolos e da homografia epipolar. Então, uma abordagem para impor a restrição de posto dois à matriz fundamental é parametrizá-la, baseado na equação (2.24).

$$\begin{aligned}
f_{11} &= -e_t e'_t h_3 \\
f_{12} &= -e_t e'_t h_4 \\
f_{13} &= e_u e'_t h_3 + e_v e'_t h_4 \\
f_{21} &= e_t e'_t h_1 \\
f_{22} &= e_t e'_t h_2 \\
f_{23} &= -e_u e'_t h_1 - e_v e'_t h_2 \\
f_{31} &= -e_t e'_v h_1 + e_t e'_u h_3 \\
f_{32} &= -e_t e'_v h_2 + e_t e'_u h_4 \\
f_{33} &= e_u e'_v h_1 + e_v e'_v h_2 - e_u e'_u h_3 - e_v e'_u h_4
\end{aligned} \tag{2.24}$$

Supondo que $e_t \neq 0$ e $e'_t \neq 0$, ou seja, os epipolos não estão no infinito, podemos dividir a equação (2.24) por $-f_{11} = e_t e'_t h_3$. Assim, considerando $\mathbf{e}_1 = [\alpha, \beta]^T$ e $\mathbf{e}_2 = [\lambda, \mu]^T$, podemos expressar a matriz fundamental em termos dos epipolos e da homografia epipolar, ou seja,

$$\mathbf{F} = \begin{bmatrix} 1 & a & -\alpha - \alpha\beta \\ b & c & -b\alpha - c\beta \\ -\lambda - b\mu & -a\lambda - c\mu & f_{33} \end{bmatrix}, \tag{2.25}$$

onde $a = -h_4/h_3$, $b = h_1/h_3$, $c = h_2/h_3$ e $f_{33} = \alpha(\lambda + b\mu) + \beta(a\lambda + c\mu)$. A equação anterior expressa a matriz fundamental com apenas sete incógnitas, ou seja, possui sete graus de liberdade.

Além disso, considerando $\mathbf{F} = [c_1, c_2, c_3]$, a equação anterior mostra que $c_3 = -\alpha c_1 - \beta c_2$, isto é, $\alpha c_1 + \beta c_2 + c_3 = 0$. Isso mostra que a parametrização expressa uma matriz de posto dois.

A equação (2.25) é a melhor parametrização apenas se os epipolos não estão no infinito e o primeiro elemento da matriz \mathbf{F} é o maior valor absoluto em relação aos demais. Evidentemente, nem sempre isso ocorre. Devido a isso, Luong et al. (1993) utilizou essa parametrização pela primeira vez e demonstrou que existem 36 parametrizações diferentes e que tudo depende das características da matriz \mathbf{F} . Como não se sabe, a priori, as características de \mathbf{F} , para uma estimação de qualidade, é preciso testar todas as parametrizações possíveis, no entanto, isso tem um custo computacional elevado. Devido a isso, vários autores propuseram métodos que minimizam esses problemas (ZHANG; LOOP, 2001; BARTOLI; STURM, 2004; HARTLEY; ZISSERMAN, 2003).

Máxima Verossimilhança

Conforme já discutido, a estimação da matriz fundamental necessita de um conjunto de correspondência de pontos ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$). Devido ao processo de obtenção dessas correspondências, é natural que exista ruído em suas coordenadas. Este ruído degrada a estimação de \mathbf{F} . Para contornar este problema, o método da máxima verossimilhança estima as coordenadas reais dos pontos de cada correspondência ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$), sujeito a equação (2.10). Para isso, é necessário uma estimação inicial das matrizes de projeção canônicas (HARTLEY; ZISSERMAN, 2003) do conjunto binocular, ou seja,

$$\mathbf{P}_1 = [\mathbf{I} \ \mathbf{0}] \text{ e } \mathbf{P}_2 = [\mathbf{H} \ \mathbf{t}], \quad (2.26)$$

onde \mathbf{H} é uma matriz 3 x 3 que representa a homografia e \mathbf{t} é um vetor que caracteriza a translação de uma câmera em relação a outra. É indispensável também uma estimação inicial dos pontos 3D, \mathbf{M}_i , que satisfazem

$$\tilde{\mathbf{m}}_{1i} = \mathbf{P}_1 \tilde{\mathbf{M}}_i \text{ e } \tilde{\mathbf{m}}_{2i} = \mathbf{P}_2 \tilde{\mathbf{M}}_i. \quad (2.27)$$

Com isso, dado um conjunto com n correspondência de pontos, encontra-se \mathbf{H} , \mathbf{t} e os n pontos \mathbf{M}_i que minimizam

$$\sum_{i=1}^n [d^2(\tilde{\mathbf{m}}_{1i}, \mathbf{P}_1 \tilde{\mathbf{M}}_i) + d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{P}_2 \tilde{\mathbf{M}}_i)], \quad (2.28)$$

onde $d^2(\cdot, \cdot)$ representa o quadrado da distância euclidiana entre dois pontos. Em seguida, obtém-se a matriz fundamental fazendo $\mathbf{F} = [\mathbf{t}]_{\times} \mathbf{H}$.

A minimização da equação anterior envolve a solução de um problema de otimização não-linear. O algoritmo de Levenberg-Marquardt (LEVENBERG, 1944; MARQUARDT, 1963), na visão computacional, é muito utilizado para solucionar tais problemas. Contudo, a complexidade computacional de tal algoritmo é N^3 , onde N é número de incógnitas a serem estimadas (FRANÇA et al., 2008). A equação (2.28) possui um número excessivo de incógnitas. Felizmente, observando as equações (2.26) e (2.27), fica claro que uma mudança nos elementos de \mathbf{H} ou \mathbf{t} altera as coordenadas dos pontos de todas as correspondências, enquanto que uma alteração nas coordenadas de algum ponto \mathbf{M}_i reflete-se apenas nas coordenadas dos pontos da i -ésima correspondência. Dessa forma, o jacobiano da função definida pela equação (2.28) tem uma estrutura esparsa e existe uma derivação do algoritmo Levenberg-Marquardt, popularizada por Hartley e Zisserman (2003) e referenciada por Levenberg-Marquardt Dividido, que toma vantagem da estrutura do problema para reduzir significativamente a complexidade do algoritmo.

Uma implementação do algoritmo Levenberg-Marquardt Dividido já foi implementada como um *script* para o Matlab (FRANÇA et al., 2008) e, no presente trabalho, este *script* é utilizado para minimizar (2.28). Com isso, a complexidade do problema é reduzida de N^3 para apenas N (número de incógnitas) (FRANÇA et al., 2008) e, dessa forma, o método da máxima verossimilhança constitui um método interessante para estimar a matriz fundamental.

Zhang e Loop

Zhang e Loop (2001), propuseram uma técnica para estimar a matriz fundamental, que consiste em transformar os pontos da imagem no espaço projetivo. Ao invés do uso de 36 mapas para parametrizar a matriz fundamental, somente é necessário executar a otimização não-linear com uma parametrização da matriz fundamental.

Em trabalhos anteriores ao proposto pelos autores, requeriam a consideração de 36 parametrizações distintas para explicar o fato de que um epipolo pode ser no infinito e um elemento de transformação epipolar pode ser igual a 0. Esse fato, leva um custo computacional muito elevado no processo de otimização.

Dado um conjunto com n correspondências de pontos, Luong et al. (1993) propuseram um algoritmo no qual busca-se \mathbf{F} que minimiza

$$\sum_{i=1}^n [d^2(\mathbf{m}_{2i}, \mathbf{F}\tilde{\mathbf{m}}_{1i}) + d^2(\mathbf{m}_{1i}, \mathbf{F}^T\tilde{\mathbf{m}}_{2i})], \quad (2.29)$$

com $d^2(\mathbf{x}, \mathbf{l})$ igual ao quadrado da distância euclidiana entre o ponto \mathbf{x} e a reta \mathbf{l} , e a equação (2.25), onde $a, b, c, \beta, \alpha, \lambda$ e μ são constantes que dependem de uma estimativa inicial de \mathbf{F} .

Esse método, propõe a transformação dos pontos de uma imagem no espaço projetivo apenas com uma parametrização, como na equação (2.25), assegurando $a = 1$. A ideia é encontrar uma transformação projetiva em cada imagem, denotada por \mathbf{P}_1 e \mathbf{P}_2 , de modo que no espaço da imagem transformada o primeiro elemento da matriz fundamental tem o maior valor e os epipolos não estão no infinito. Deixando os pontos de imagem em transformação do espaço ser

$$\hat{\mathbf{m}}_{1i} = \mathbf{P}_1\tilde{\mathbf{m}}_{1i} \text{ e } \hat{\mathbf{m}}_{2i} = \mathbf{P}_2\tilde{\mathbf{m}}_{2i}, \quad (2.30)$$

em seguida, a matriz fundamental no espaço transformado é dada por

$$\hat{\mathbf{F}} = \mathbf{P}_2^{-T}\mathbf{F}\mathbf{P}_1^{-1}. \quad (2.31)$$

De acordo com Zhang e Loop (2001), as matrizes de projeção \mathbf{P}_1 e \mathbf{P}_2 são da ordem 3×3 . Para o cálculo, é necessária uma estimativa inicial da matriz fundamental, denominada \mathbf{F}_0 . Normalmente essa matriz é calculada pelo método proposto por Hartley (1997). Então, tem-se os epipolos iniciais, identificados por \mathbf{e}_{10} e \mathbf{e}_{20} . As matrizes de projeção são determinadas como segue:

1. Inicializar \mathbf{P}_1 e \mathbf{P}_2 , como matrizes identidade.
2. Encontrar a posição do maior elemento de \mathbf{F}_0 , denotada por $(i_0; j_0)$.
3. Se $j_0 \neq 0$, permutar linhas 0 e j_0 de \mathbf{P}_1 e permutar os elementos 0 e j_0 do epipolo $\tilde{\mathbf{e}}_{01}$.
4. Se $i_0 \neq 0$, permutar linhas 0 e i_0 de \mathbf{P}_2 e permutar os elementos 0 e i_0 do epipolo $\tilde{\mathbf{e}}_{02}$.
5. Se $|\tilde{\mathbf{e}}_{01}[1]| > |\tilde{\mathbf{e}}_{01}[2]|$, permutar os elementos 1 e 2 do epipolo $\tilde{\mathbf{e}}_{01}$ e permutar as linhas 1 e 2 da matriz \mathbf{P}_1 .
6. Se $|\tilde{\mathbf{e}}_{02}[1]| > |\tilde{\mathbf{e}}_{02}[2]|$, permutar os elementos 1 e 2 do epipolo $\tilde{\mathbf{e}}_{02}$ e permutar as linhas 1 e 2 da matriz \mathbf{P}_2 .

Os passos 3 e 4 garante que o primeiro elemento da matriz fundamental no espaço transformado tem o maior valor, enquanto as etapas 5 e 6 garante que o epipolos não estejam no infinito. Com essas manipulações, pode-se usar a parametrização (2.25) para estimar $\hat{\mathbf{F}}$ a partir dos pontos da imagem transformada $\hat{\mathbf{m}}_{1i}$ e $\hat{\mathbf{m}}_{2i}$. A matriz fundamental no espaço da imagem original é dada por $\mathbf{F} = \mathbf{P}_2^T \hat{\mathbf{F}} \mathbf{P}_1$.

Contudo, a parametrização proposta pela equação (2.25) funciona bem apenas quando os epipolos estão no infinito e a , b e c são todos menores que a unidade. Para contornar esse problema, Zhang e Loop (2001) propuseram transformações projetivas que agem em ambas as imagens e levam as correspondência de pontos para novos sistemas de coordenadas, nos quais essas condições são sempre satisfeitas.

Bartoli e Sturm

Como visto anteriormente, a parametrização é importante para garantir a estimação de uma matriz fundamental \mathbf{F} com posto dois. Entretanto, com a parametrização utilizando 36 mapas de busca, a tarefa torna-se de uma alta complexidade computacional. Zhang e Loop (2001), em seu trabalho, reduzem os

mapas de buscas, analisando a matriz \mathbf{F} estimada, pelo método dos 8 pontos, para determinar uma única parametrização. Se a matriz inicial, \mathbf{F}_0 não for bem condicionada, não há garantias que a parametrização escolhida para o método de Zhang seja a melhor. Assim, Bartoli e Sturm (2004) propuseram um novo método que por meio da decomposição de valor singular (SVD), a matriz fundamental é representada como sendo $\mathbf{F} = \mathbf{U} \text{diag}(1, \sigma, 0) \mathbf{V}^T$, onde σ é a razão entre os dois autovalores de \mathbf{F} . Dessa forma, a matriz fundamental pode ser atualizada por meio das seguintes equações

$$\mathbf{U} \leftarrow \mathbf{U}\mathbf{R}(\theta_u), \mathbf{V} \leftarrow \mathbf{V}\mathbf{R}(\theta_v), \sigma \leftarrow \sigma + \delta,$$

onde $\mathbf{R}(\theta_i)$ representa matrizes de rotação baseadas nos ângulos de Euler. Assim, usando um algoritmo não-linear de otimização, é possível ajustar \mathbf{F} , iterativamente, de forma a minimizar a equação (2.29). A vantagem do método é que \mathbf{F} é representada com o mínimo de parâmetros possível (três para o vetor θ_u , três para θ_v e um para δ_σ).

A estimativa inicial para o método é representado por \mathbf{F}_0 . Normalmente \mathbf{F}_0 é calculada pelo método proposto por Hartley (1997).

Resumindo, o método de Bartoli e Sturm (2004), resolve o problema de estimar a matriz fundamental sobre um conjunto mínimo de sete parâmetros. É proposta uma representação ortonormal que permite uma fácil atualização da matriz fundamental com sete parâmetros. As matrizes de projeção canônicas podem ser extraídas diretamente da representação ortonormal. O método pode ser utilizado com otimizadores não-lineares, como Levenberg-Marquardt.

2.3.3.3 Métodos Robustos

Devido à dificuldade de se estabelecer correspondência de pontos antes da geometria epipolar ser conhecida, espera-se que, em um conjunto de correspondências ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$), existam erros e ruído nas coordenadas dos pontos \mathbf{m}_{1i} e \mathbf{m}_{2i} , que permite classificar as correspondências em boas ou falsas. São consideradas boas correspondências as que o ruído presente nas coordenadas dos pontos \mathbf{m}_{1i} e \mathbf{m}_{2i} for de até dois pixels, dado um conjunto de correspondências ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$) e, uma correspondência é dita uma “falsa correspondência” quando, de fato, o ponto \mathbf{m}_2 não está sobre a reta epipolar de \mathbf{m}_1 .

Normalmente, supõe-se que o ruído presente nas coordenadas dos pontos, \mathbf{m}_{1i} e \mathbf{m}_{2i} , das boas correspondências possui uma distribuição Gaussiana. Isso é uma suposição razoável, pois os erros nas coordenadas são de apenas poucos pixels.

Os algoritmos para cálculo da matriz fundamental são capazes de lidar com o ruído presente nas boas correspondências. Contudo, apenas algumas poucas falsas correspondências presentes no conjunto são capazes de degradar totalmente a estimação de \mathbf{F} . Neste caso, algoritmos robustos que conseguem tratar com tais correspondências devem ser utilizados.

Os algoritmos robustos mais bem sucedidos tentam detectar as falsas correspondências e retirá-las do conjunto. Desse modo, ao final do algoritmo, a matriz fundamental pode ser estimada por um dos algoritmos anteriores, pois haverá apenas boas correspondências. Os métodos robustos mais utilizados para estimação da matriz fundamental são: RANSAC (FISCHLER; BOLLES, 1981), MSAC (TORR; ZISSERMAN, 1998), MLESAC (TORR; ZISSERMAN, 2000) e LMEDS (ROUSSEEUW, 1987). A seguir, alguns métodos são explicados com mais detalhes.

RANSAC

Um dos métodos robustos mais conhecidos na literatura é o RANSAC. Este foi introduzido na visão computacional por Fischler e Bolles (1981). Basicamente, o RANSAC pode ser resumido como segue.

Dado um conjunto de N correspondências $c_i = (\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i})$, escolhe-se, aleatoriamente, N_{sc} subconjuntos de N_{nc} correspondências. Para cada subconjunto (indexado por j), estima-se a matriz fundamental, \mathbf{F}_j , e calcula-se o resíduo, $r_{ji}^2(\mathbf{F}_j, c_i)$, de todas as N correspondências. Cada resíduo é comparado com um limiar r_{th}^2 . Se $r_{ji}^2 < r_{th}^2$, a correspondência c_i é considerada uma boa correspondência. Após N_{sc} iterações, retém-se a matriz fundamental, \mathbf{F}_{win} , que ajustou-se ao maior número de boas correspondências. Por último, estima-se \mathbf{F} com apenas as boas correspondências (identificadas por $r_{(win)i}^2 < r_{th}^2$).

O número de subconjuntos, N_{sc} , utilizado no RANSAC deve ser tal que, supondo, dentre todas as N correspondências, uma porcentagem de falsas correspondências, existe uma probabilidade p de que (ao menos) um subconjunto j seja composto apenas por boas correspondências. Neste caso, N_{sc} é igual a

$$N_{sc} = \frac{\log(1 - p)}{\log(1 - (1 - \epsilon)^{N_{nc}})}. \quad (2.32)$$

Talvez, uma tática inteligente seja iniciar N_{sc} usando a equação anterior e atualizá-lo a cada iteração j , ou seja, uma vez que a porcentagem, ϵ , de falsas correspondências tenha sido determinada, N_{sc} pode ser atualizado por (2.32).

Da equação anterior, observa-se que N_{sc} aumenta exponencialmente com N_{nc} e ϵ . Por exemplo, considerando uma probabilidade $p = 99\%$, se $\epsilon = 25\%$ e $N_{nc} = 7$, então $N_{sc} = 33$. Contudo, se $\epsilon = 40\%$ e $N_{nc} = 8$, tem-se $N_{sc} = 272$. Assim, já que quanto maior N_{sc} , maior o custo computacional, o ideal é ter os valores de N_{nc} e ϵ menores possíveis. O valor de ϵ depende de como o conjunto total de correspondências foi estabelecido, ou seja, ele depende do grau de confiança do algoritmo de correspondência de pontos.

Como \mathbf{F} tem sete graus de liberdade, o valor mínimo para N_{nc} é sete. Contudo, a solução com apenas sete correspondências não é estável. Por isso, pode-se utilizar N_{nc} igual ou um pouco maior que oito. O parâmetro mais crítico a ser escolhido é o limiar r_{th}^2 , pois dele depende o critério que diz se uma correspondência é boa ou ruim. Se r_{th}^2 é muito pequeno, boas correspondências podem ser consideradas ruins. Por outro lado, um r_{th}^2 grande faz com que algumas falsas correspondências não sejam detectadas.

Normalmente, o cálculo do resíduo r_{ji}^2 é dado por

$$r_{ji}^2 = d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{F}_j \tilde{\mathbf{m}}_{1i}) + d^2(\tilde{\mathbf{m}}_{1i}, \mathbf{F}_j^T \tilde{\mathbf{m}}_{2i}). \quad (2.33)$$

Além disso, se for considerado um ruído com uma distribuição Gaussiana de média zero e desvio padrão σ_d , r_{th}^2 é definido como

$$r_{th}^2 = d_{th}^2 \sigma_d^2, \quad (2.34)$$

onde d_{th}^2 deve ser escolhido tal que exista uma probabilidade p_d de uma boa correspondência ser erroneamente considerada uma falsa correspondência. Contudo, muitas vezes, d_{th}^2 é escolhido empiricamente.

O método RANSAC pode obter uma boa estimação de \mathbf{F} mesmo que mais de 50% das correspondências sejam ruins. A desvantagem evidente é que este necessita de uma estimativa do desvio padrão do ruído, σ_d .

MSAC

O algoritmo RANSAC procura a matriz \mathbf{F}_j que minimiza a função

$$\sum_{i=1}^N \mathcal{F}(r_{ji}),$$

onde $r_{ji} = r(\mathbf{F}_j, c_i)$ e

$$\mathcal{F}(r_{ji}) = \begin{cases} 0, & \text{se } r_{ji} \leq r_{th}, \\ 1, & \text{se } r_{ji} > r_{th}. \end{cases}$$

Na equação anterior, é evidente que, se r_{th} tiver um valor muito elevado, todas as correspondências serão consideradas boas. Neste caso, qualquer matriz \mathbf{F}_j teria a mesma pontuação, ou seja, $\sum_{i=1}^N \mathcal{F}(r_{ji})$ seria sempre igual a N . Por isso, Torr e Zisserman (1998) sugeriram uma discreta alteração na função $\mathcal{F}(r_{ji})$, ou seja,

$$\mathcal{F}(r_{ji}) = \begin{cases} r_{ji}, & \text{se } r_{ji} \leq r_{th}, \\ r_{th}, & \text{se } r_{ji} > r_{th}. \end{cases}$$

Assim, cada boa correspondência contribui com um valor diferente e proporcional ao seu grau de ajuste a \mathbf{F}_j . Desse modo, mesmo considerando um número igual de boas correspondências, a função $\mathcal{F}(r_{ji})$ deve ter valores diferentes para matrizes fundamentais diferentes. Os mesmos autores propuseram um novo método, denominado MLESAC (TORR; ZISSERMAN, 2000), que produz uma melhor estimação da matriz fundamental, em comparação ao MSAC e RANSAC, no entanto, possui um custo computacional muito elevado e nunca tornou-se muito popular.

LMEDS

O algoritmo RANSAC e o MSAC só podem ser utilizados quando existe informação sobre a distribuição do ruído inserido no conjunto de correspondências (para que seja determinado o valor de r_{th}). Como discutido anteriormente, essa informação é utilizada no critério que define se uma correspondência é boa ou não. Caso tal informação não esteja disponível, pode-se ainda estimar \mathbf{F} utilizando o método LMEDS. Este é muito semelhante ao RANSAC. A principal diferença é que uma matriz \mathbf{F}_j é considerada boa quando a mediana do resíduo é pequena. Assim, após \mathbf{F}_j ser estimada, calcula-se a mediana dos resíduos, r_{ji} , de todas as N correspondências. Após n iterações, retém-se a matriz fundamental, \mathbf{F}_{win} , que obteve a menor mediana. Por último, estima-se \mathbf{F} com apenas as boas correspondências, identificadas por $r(\mathbf{F}_{win}, c_i) < \sigma_{th}$, onde, do trabalho de Rousseeuw (1987),

$$\sigma_{th} = 1,4826[1 + 5/(N - m)]\sqrt{r_{med}},$$

onde r_{med} é a menor mediana do resíduo, dentre todos os n subconjuntos.

O algoritmo LMEDS tem uma grande desvantagem, pois não consegue estimar a matriz fundamental corretamente se existirem um número maior que 50% de

falsas correspondências.

A seguir, são demonstrados os passos que compõem esse algoritmo.

1. Dado N emparelhamentos, uma técnica Monte Carlo (FRANÇA, 2005) é utilizada para escolher m subconjuntos de n diferentes pares ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$).
2. Para cada subconjunto, indexado por j , usa-se uma técnica que forneça uma boa estimativa de \mathbf{F}_j com n pares ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$).
3. Para cada \mathbf{F}_j , determina-se a média do resíduo ao quadrado, denotado por M_j , com respeito ao conjunto completo de emparelhamentos, isto é,

$$M_j = \frac{1}{N} \sum_{i=1 \dots N} [d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{F}_j \tilde{\mathbf{m}}_{1i}) + d^2(\tilde{\mathbf{m}}_{1i}, \mathbf{F}_j^T \tilde{\mathbf{m}}_{2i})],$$

na qual $d(\tilde{\mathbf{m}}_{2i}, \mathbf{F}_j \tilde{\mathbf{m}}_{1i})$ é a distância do ponto \mathbf{m}_{2i} a reta $\mathbf{F}_j \tilde{\mathbf{m}}_{1i}$ e $d(\tilde{\mathbf{m}}_{1i}, \mathbf{F}_j^T \tilde{\mathbf{m}}_{2i})$ é a distância do ponto \mathbf{m}_{1i} a reta $\mathbf{F}_j^T \tilde{\mathbf{m}}_{2i}$

4. Retêm-se a estimativa \mathbf{F}_j que tem o menor M_j (denotado por M_{min}).
5. Elimina-se os pares ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$) que possuem o resíduo

$$r_{ji}^2 = d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{F}_j \tilde{\mathbf{m}}_{1i}) + d^2(\tilde{\mathbf{m}}_{1i}, \mathbf{F}_j^T \tilde{\mathbf{m}}_{2i}) \geq r_{th}^2$$

6. Refina-se a matriz fundamental estimando-a sem os emparelhamentos ruins.

Os emparelhamentos que compõem cada subconjunto devem estar bem espalhados pela área da imagem. Caso contrário, a matriz \mathbf{F}_j não se ajustará bem aos outros emparelhamentos. Assumindo que o conjunto completo de emparelhamentos possa ter no máximo uma fração ϵ de ruído, a probabilidade, P , de pelo menos um subconjunto dos m subconjuntos seja livre de ruídos é dada pela equação (2.32).

2.3.4 Correspondência de Pontos

Estabelecer correspondência de pontos entre duas ou mais imagens é uma das tarefas mais comuns na visão computacional. De fato, a estimativa da matriz fundamental de um conjunto binocular, assume a existência de um conjunto de correspondências de pontos entre duas imagens. Contudo, apesar dos esforços de pesquisadores em todo o mundo, o problema mostra-se extremamente complexo e ainda não existe uma solução automática que dê bons resultados na maioria dos

casos. Diversos fatores tornam a correspondência de pontos difícil: (i) a ambiguidade inerente ao problema requer a introdução de restrições físicas e geométricas; (ii) pontos em uma imagem sem um correspondente na outra (oclusões); (iii) distorções radiométricas que fazem a projeção de um mesmo ponto 3D ter tons de cinza diferentes, quando ele é capturado por câmeras distintas, e; (iv) distorções projetivas que tornam a forma de um objeto diferente, quando capturadas de pontos-de-vista distintos. Contudo, Zhang et al. (1995) propôs um método interessante para determinar a correspondência de pontos entre duas imagens. O método propõe uma abordagem robusta que explora a restrição epipolar de imagens não calibradas. Assim, as imagens podem ser adquiridas por diferentes câmeras e até em instantes diferentes. São utilizadas técnicas clássicas, tais como, correlação e métodos de relaxamento, para encontrar um conjunto inicial de correspondentes e, então, é utilizada uma técnica robusta, a mediana dos Quadrados Mínimos (LMedS) para eliminar falsas correspondências e calcular a matriz fundamental.

Felizmente, em aplicações como, por exemplo, estimação da matriz fundamental, calibração de câmeras e retificação de imagens (LOOP; ZHANG, 1999), é necessário estabelecer apenas algumas poucas correspondências. Neste contexto, quanto à quantidade e densidade de correspondências, nesse trabalho, trabalhou-se apenas com correspondências esparsas.

2.3.4.1 Detecção de Cantos

A detecção de cantos é uma técnica frequentemente utilizada no processamento de imagem e visão computacional para determinar pontos de uma imagem em que a intensidade luminosa muda repentinamente, mudanças repentinas em imagens geralmente refletem eventos importantes no cenário, como a descontinuação da profundidade, da orientação da superfície, mudança das propriedades do material ou variações na iluminação da cena. Podendo ser aplicada na extração de características. Entretanto, sua utilização reduz significativamente a quantidade de dados a serem processados e, descarta informação que é considerada menos relevante para uma análise da imagem, viabilizando computacionalmente a resolução de problemas. No entanto, a detecção pode identificar falsas bordas, devido a ruídos na imagem, que pode ser reparado por alguma técnica específica.

O método para a detecção de cantos proposto por Harris e Stephens (1988), computa a matriz de momento com média local a partir dos gradientes da imagem e, então, combina os autovalores da matriz de momento para computar o canto

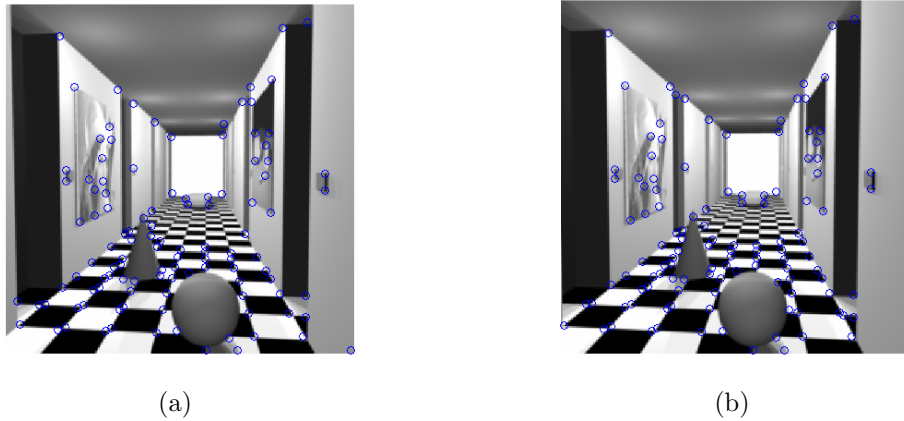


Figura 2.6: Figuras com os cantos identificados por meio do método de Harris(HARRIS; STEPHENS, 1988): a) câmera 1 (esquerda); b) câmera 2 (direita)

candidato, da qual valores máximos indicam as posições dos cantos. A chave desta equação está em examinar seus autovalores. Quando a matriz possui dois autovalores destacados, então corresponde a duas direções principais diferentes no gradiente da imagem subjacente. Na figura 2.6, é apresentado um exemplo de cantos detectados de um conjunto binocular pelo método de Harris e Stephens (1988). As regiões detectadas são assinaladas por um círculo, relativas a máximas locais, que indicam um canto.

No conjunto de ferramentas proposto, para a implementação do módulo de correspondência de pontos, inicialmente é realizada a detecção de cantos pelo método proposto por Harris e Stephens (1988), então, os pontos detectados como cantos evidentes são base de execução do método proposto por Zhang et al. (1995) que realiza a correspondência de pontos. Esse, retorna as informações dos pontos correspondidos entre as imagens do conjunto binocular, que é subsídio para estimar a matriz fundamental.

3 A Construção do Ambiente Interativo Computacional para o Cálculo da Matriz Fundamental

O ambiente computacional proposto neste trabalho tem como característica possibilitar, ao usuário inexperiente, uma rápida compreensão dos métodos para estimar a matriz fundamental de um conjunto binocular. O desenvolvimento deste ambiente computacional foi baseado no Matlab, com alguns módulos de programação na linguagem C, por meio das funções MEX (*Matlab External File*). Na sua implementação, procurou-se adotar boas práticas de programação, tais como a empregabilidade dos conceitos de usabilidade, modularidade e reusabilidade (PRESSMAN, 2006). Para tanto, adotou-se a GUI (*Graphical User Interface*) do Matlab. A cada ação programada, é invocada uma função *m-file* do *toolbox*, que retorna a informação solicitada. O *toolbox*, que dá suporte ao ambiente interativo, está organizado em pacotes e cada pacote possui um conjunto de funções para realizar uma determinada tarefa. No entanto, o usuário executa o processo completo, para estimar a matriz fundamental, por meio do ambiente interativo. Contudo, caso seja necessário, todas as funções utilizadas no cálculo da matriz fundamental, além das outras funções de apoio, podem ser executadas diretamente na linha de comando do Matlab. Todas as funções pertencentes ao conjunto de ferramentas proposto nesse trabalho são de autoria, ou seja, não foi utilizada nenhuma funcionalidade do *toolbox* de processamento de imagens disponibilizado pelo Matlab, uma vez que não atende às necessidades do projeto proposto.

Na construção do ambiente interativo, seguiu-se o processo natural para estimar a matriz fundamental de um sistema binocular não-calibrado. Na figura 3.1, é apresentado o diagrama de atividades do ambiente proposto. Nota-se que há uma sequência lógica para estimar a matriz fundamental e cada passo é necessário para o sucesso do passo subsequente.

Nas próximas seções, é explicada com maiores detalhes cada atividade imple-

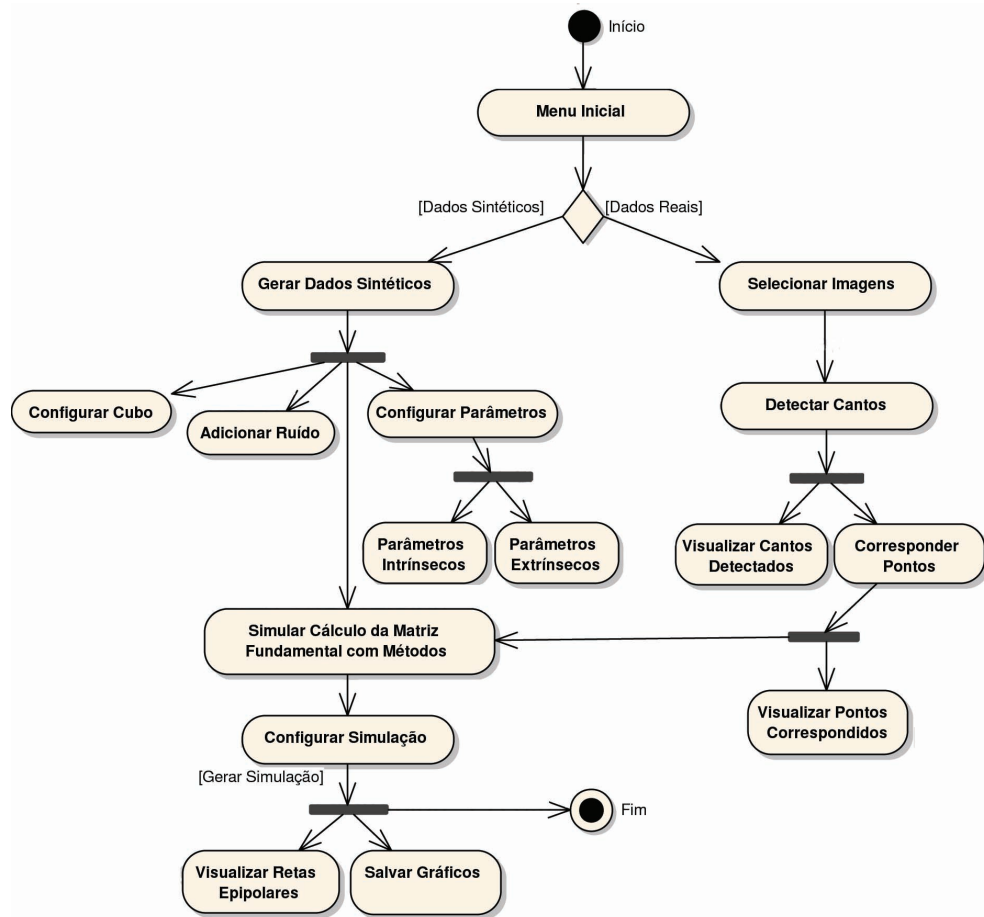


Figura 3.1: Atividades do Ambiente Computacional Proposto

mentada no ambiente computacional proposto neste trabalho.

3.1 Menu Inicial

O menu inicial, mostrado na figura 3.3, permite ao usuário selecionar as opções para a simulação do cálculo da matriz fundamental. Para isso, são disponibilizadas duas opções, uma para simular com dados sintéticos, gerados a partir de uma projeção de um cubo e, com o intuito de algo mais real, é possível por meio da outra opção simular com dados gerados a partir de imagens capturadas por um par de câmeras. Na figura 3.2, é mostrado o fluxo das atividades a partir dessa funcionalidade.

3.2 Geração de Dados Sintéticos

Essa funcionalidade é demonstrada no fluxo de atividades da figura 3.4. Tem como finalidade a geração de dados sintéticos para a simulação do cálculo da

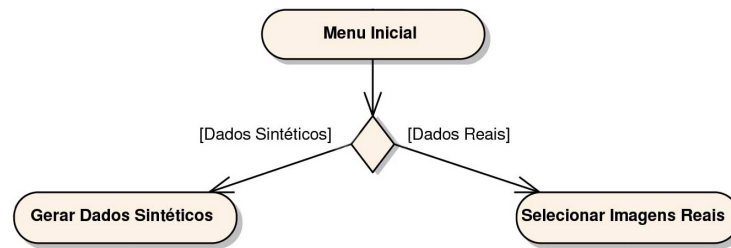


Figura 3.2: Atividades do Menu Inicial

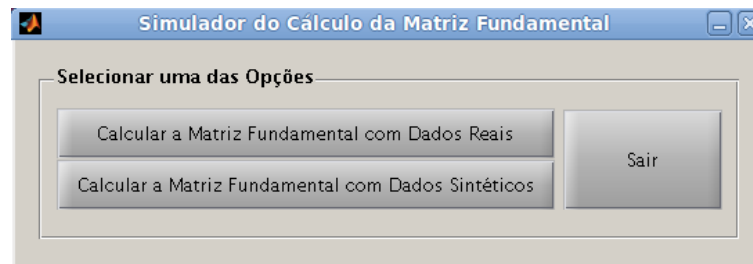


Figura 3.3: Menu Inicial do Simulador para Estimar a Matriz Fundamental

matriz fundamental. Para isso, é gerado um “cubo” sintético, no qual o usuário fornece algumas informações para sua criação, tais como, a largura da face do cubo, a quantidade de pontos de cada face e o ângulo de rotação relacionados aos eixos x , y e z . Entretanto, também, deve-se configurar os parâmetros intrínsecos e extrínsecos das câmeras simuladas. Para os parâmetros intrínsecos, é possível informar a coordenada do centro da projeção, representada por \mathbf{u}_0 e \mathbf{v}_0 e, também, os dados relativos ao fator de escala α (\mathbf{ax}) e β (\mathbf{ay}) para, respectivamente, os eixos x e y . Já os parâmetros extrínsecos, definem a rotação e translação da câmera 2 em relação a câmera 1, informados pelo usuário. A translação é definida pelos ângulos relacionados aos eixos x , y e z . Por outro lado, para a rotação, é gerada uma matriz quadrada de ordem três, que representa as rotações no espaço 3D como um produto de três rotações sucessivas em torno dos eixos x , y e z , baseada nos ângulos de Euler (HARTLEY; ZISSERMAN, 2003) e fornecidos pelo usuário.

Além das informações relacionadas às câmeras, pode-se adicionar ruídos aos dados. Para isso, deve-se informar a variância (σ) de média zero aos pontos sintéticos \mathbf{m}_1 e \mathbf{m}_2 associados, respectivamente, às câmeras 1 e 2. Isso é importante para que o usuário avalie a robustez dos métodos avaliados com relação ao ruído. Enfim, a função estima a matriz fundamental com os dados sintéticos, com e sem ruído, por meio do método dos 8 pontos (HARTLEY, 1997) e, em seguida, é calculado o ajuste de ambos os dados gerados, de acordo com Zhang (1998). Um exemplo da geração de dados sintéticos é mostrado na figura 3.5.

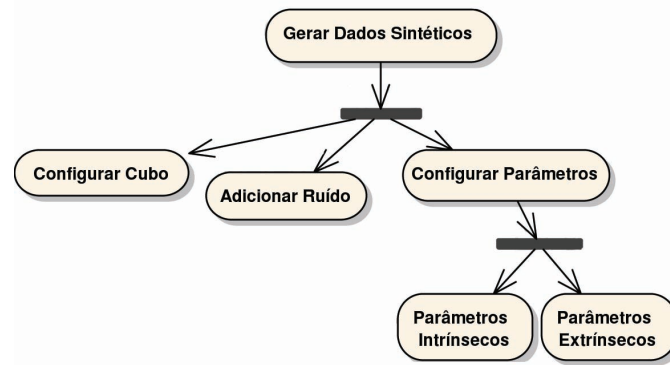


Figura 3.4: Atividades da Geração de Dados Sintéticos

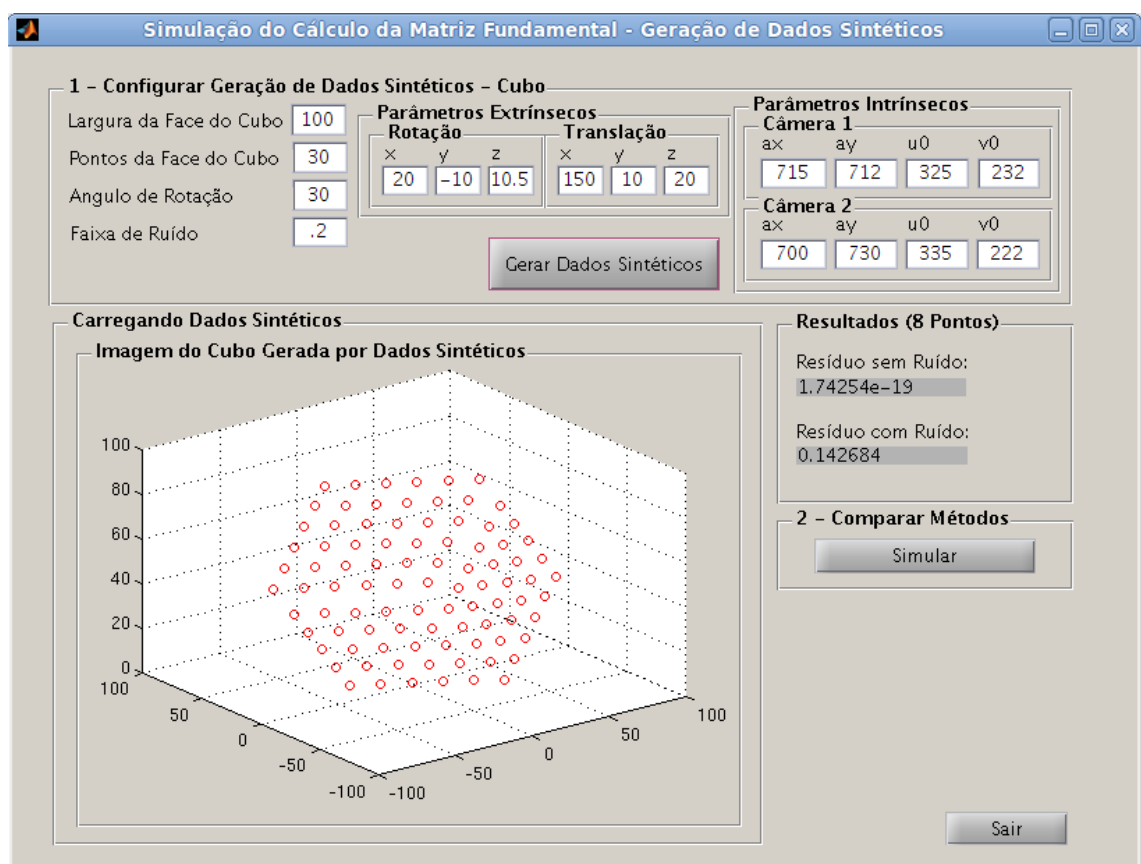


Figura 3.5: Geração de Dados Sintéticos a Partir de um Cubo Projetado

3.3 Selecionando Imagens Reais

Inicialmente, o usuário deve selecionar as imagens do conjunto binocular. Entretanto, foi utilizada uma função do próprio Matlab, denominada *wigetfile*. Com a execução dessa função, é apresentada uma caixa de diálogo ao usuário, na qual é possível selecionar os arquivos das imagens. A tarefa é simples, conforme mostrada na figura 3.6, basta acionar os botões “Câmera 1” e “Câmera 2” na tela principal do ambiente. É importante que as imagens possuam uma boa resolução

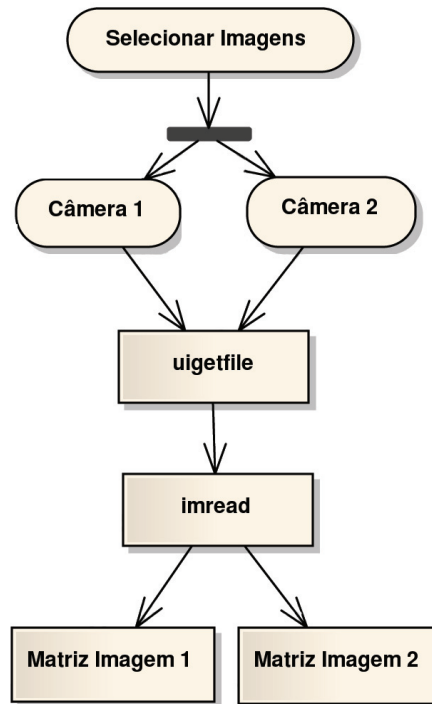


Figura 3.6: Atividades da Leitura das Imagens pelo Ambiente Computacional

e fatores como iluminação devem ser levado em consideração. Esses cuidados,

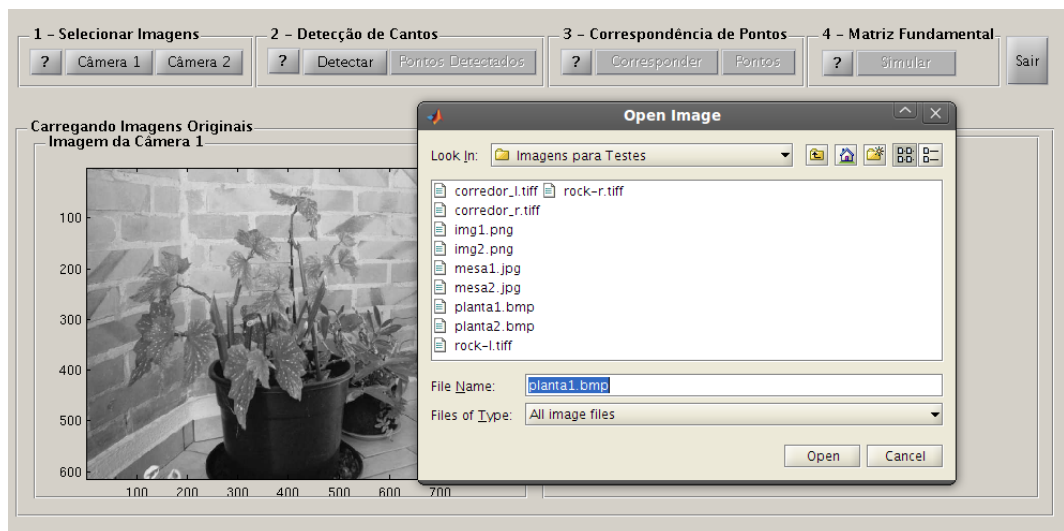


Figura 3.7: Carregando as Imagens PLANTA no Ambiente Interativo

facilitam a obtenção de bons resultados da matriz fundamental. Após isso, as imagens são lidas pela função *imread*, do próprio Matlab. Assim, são obtidas as matrizes das imagens. Enfim, as imagens são mostradas em espaços visíveis à execução da tarefa, especificamente, no componente gráfico da GUI denominado *Axes*. Na figura 3.7, é apresentada o ambiente com as imagens PLANTA carregadas. Em seguida, o botão para detecção de cantos é habilitado.

3.4 Detecção de Cantos

A detecção de cantos é uma tarefa importante para a estimação da matriz fundamental, uma vez que é subsídio para a fase de correspondência de pontos. De acordo com a figura 3.8, o acionamento do botão “Detectar” faz a chamada da função “cn_harris”, que possui a implementação do método proposto por Harris e Stephens (1988). Os parâmetros de entrada dessa função consistem na passagem da matriz obtida pela leitura de cada imagem. Então a função faz a varredura na matriz da imagem e computa a matriz de momento com média local a partir dos gradientes da imagem e, então, combina os autovalores da matriz de momento para computar o canto candidato, da qual valores máximos indicam as posições dos cantos. Portanto, após a execução, é utilizada a função *plot* do Matlab para desenhar círculos na imagem original, que correspondem às coordenadas identificadas como cantos, que substitui a imagem original no ambiente computacional proposto, conforme mostrado na figura 3.9. Também, como retorno dessa função, tem-se uma matriz, para cada imagem, com as coordenadas dos pontos identificados. Essa matriz é importante para efetuar a correspondência de pontos entre as imagens.

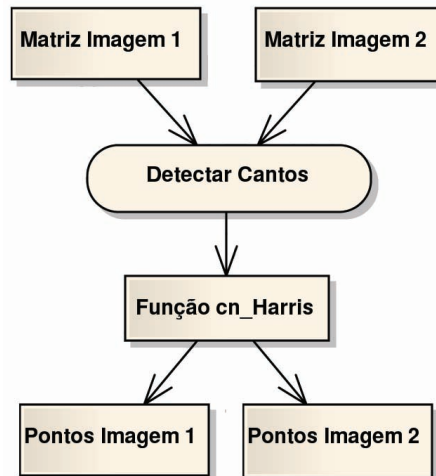


Figura 3.8: Atividades da Detecção de Cantos pelo Ambiente Computacional

3.5 Correspondência de Pontos

O algoritmo de Zhang et al. (1995), para correspondência de pontos, foi implementado para integrar o conjunto de ferramentas deste trabalho. Como resultado dessa função, é retornado um conjunto de pontos correspondidos, a mediana do ajuste e a estimativa inicial da matriz fundamental, calculada pelo método dos

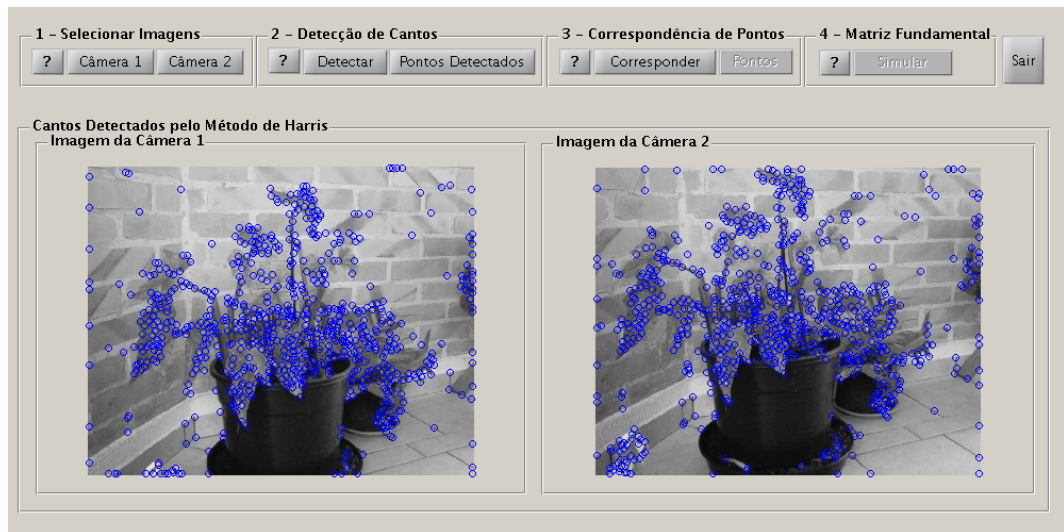


Figura 3.9: Execução da Detecção de Cantos das Imagens PLANTA

8 pontos (HARTLEY, 1997). Na figura 3.11, é mostrado, no ambiente proposto, um exemplo da execução da correspondência de pontos entre as imagens denominadas “PLANTA”. Nota-se que nas imagens, os pontos correspondentes são assinalados e numerados. As atividades dessa funcionalidade podem ser vistas na figura 3.10. Uma vez obtidos os dados dos pontos correspondidos, então,

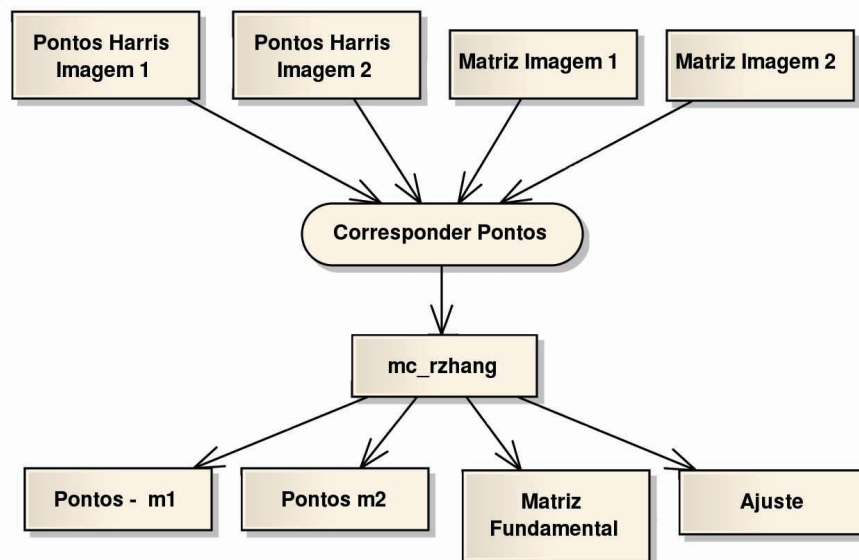


Figura 3.10: Atividades da Correspondência de Pontos pelo Ambiente Computacional

pode-se simular o cálculo da matriz fundamental pelos métodos integrantes no conjunto de ferramentas proposto neste trabalho. Por fim, é habilitado o botão para estimar a matriz fundamental, possibilitando ao usuário a simulação de vários métodos lineares (HARTLEY, 1997; TRAJKOVIC; HEDLEY, 1997; LIU;

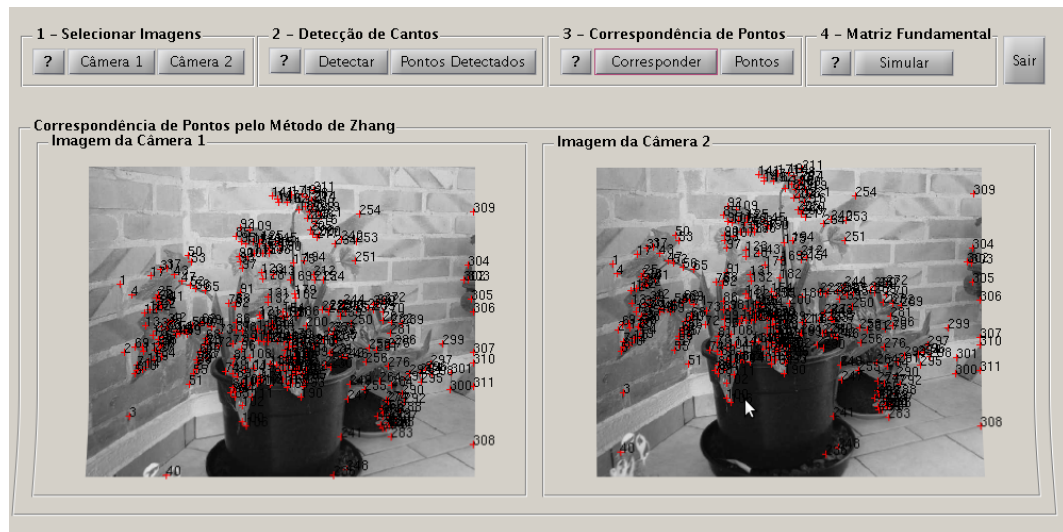


Figura 3.11: Execução da Correspondência de Pontos

MANNER, 2003), não-lineares (HARTLEY; ZISSERMAN, 2003; ZHANG; LOOP, 2001; BARTOLI; STURM, 2004) e robustos (FISCHLER; BOLLES, 1981; TORR; ZISSERMAN, 1998; ROUSSEUW, 1987).

3.6 Comparação de Métodos para Estimar Matriz Fundamental

Esta funcionalidade, do ambiente computacional proposto, tem grande importância para o entendimento dos usuários, pois permite escolher quais métodos são utilizados na estimação da matriz fundamental, bem como seus parâmetros de configuração. Isto, além de permitir o entendimento dos mesmos, facilita a comparação dos métodos em termos de exatidão e número de iterações, por meio de resultados gráficos e textuais. Na figura 3.12, é exemplificado o fluxo das atividades desse simulador. Nota-se que para iniciar a funcionalidade é necessário o conjunto de correspondência de pontos, gerado no passo anterior. Um exemplo da comparação entre métodos, para o cálculo da matriz fundamental, é mostrado na figura 3.13. Observa-se, na figura 3.13, que é possível efetuar várias execuções com configurações variadas, pois, nesse ambiente, são disponibilizadas algumas opções para configurações individuais dos métodos não-lineares e robustos. Também é possível configurar a geração da simulação e testes a serem realizados. Como resultados, são apresentadas as medianas dos ajustes de cada método calculado, os gráficos comparativos e as respectivas matrizes fundamentais calculadas no próprio console do Matlab. Então, é possível também visualizar as retas epipolares e epipolos dos métodos calculados, conforme mostrado na figura 3.15, assim como

salvar as imagens geradas.

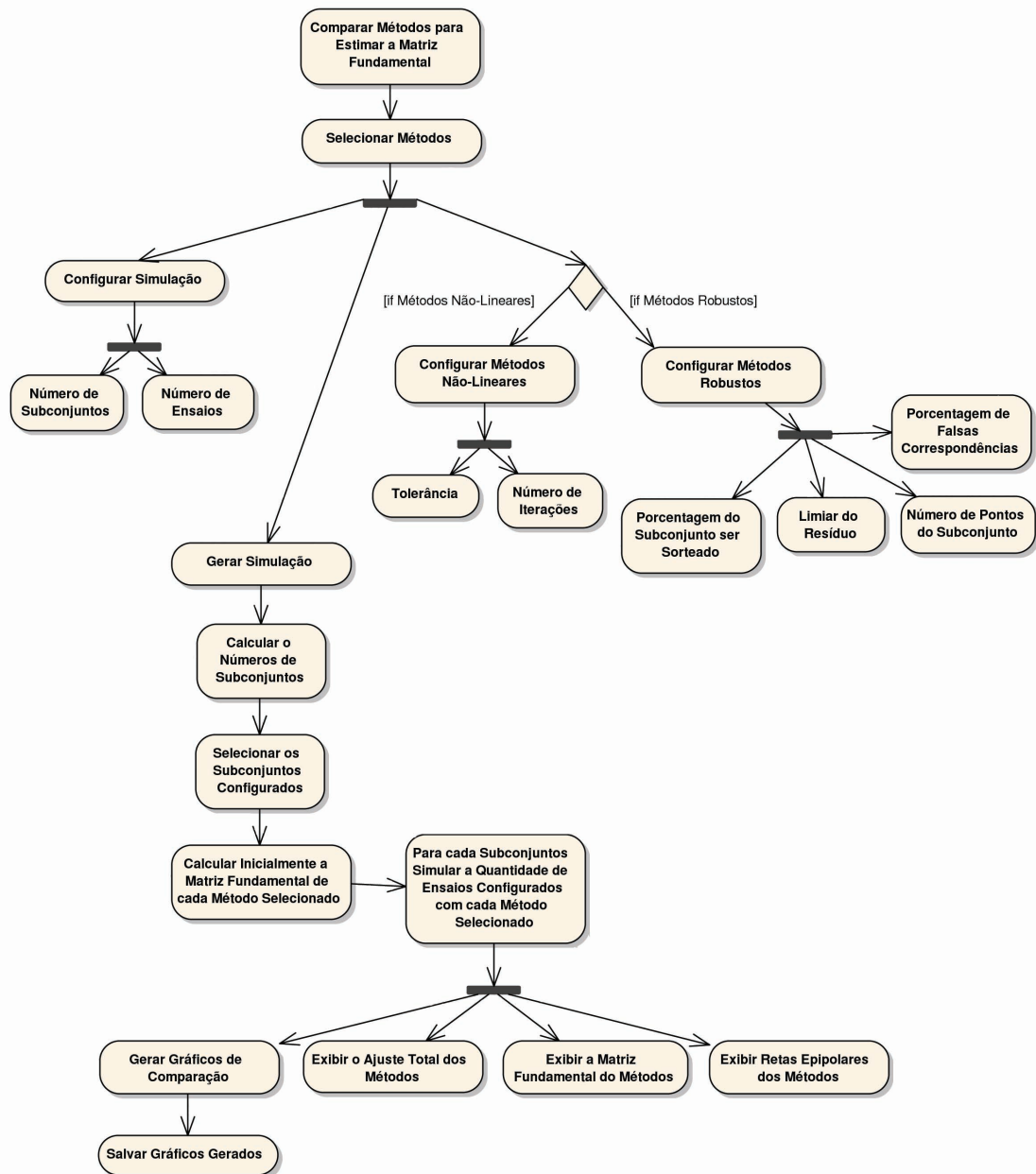


Figura 3.12: Atividades da Simulação de Comparações entre Métodos para Estimar a Matriz Fundamental

O ambiente computacional permite, sem mesmo sair da tela, uma nova simulação com combinações diferentes de métodos para estimar a matriz fundamental. Além das informações geradas no ambiente computacional proposto, também são gerados dados, em modo texto, na janela de comando do Matlab. Na figura 3.14, é mostrada a saída de dados da simulação geradas pelas configurações definidas na figura 3.13.

Comparar o desempenho de dois ou mais métodos para cálculo da matriz fundamental não é uma tarefa óbvia. Isto é devido a dois fatores. Primeiro, a matriz

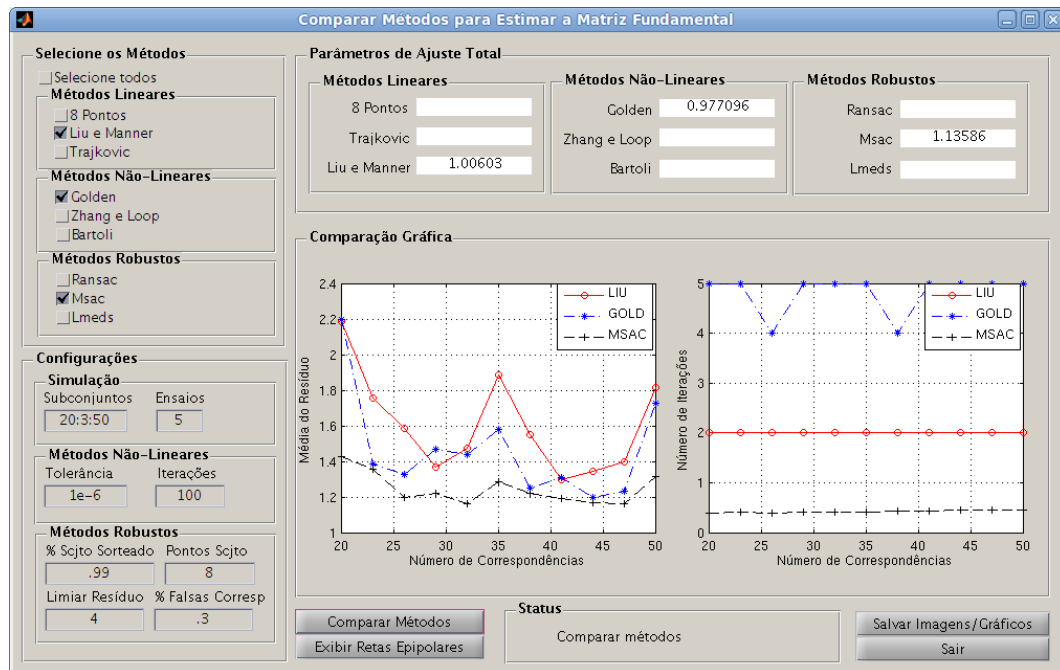


Figura 3.13: O Novo Ambiente Estimando a Matriz Fundamental dos Métodos Lineares, Não-Lineares e Robustos

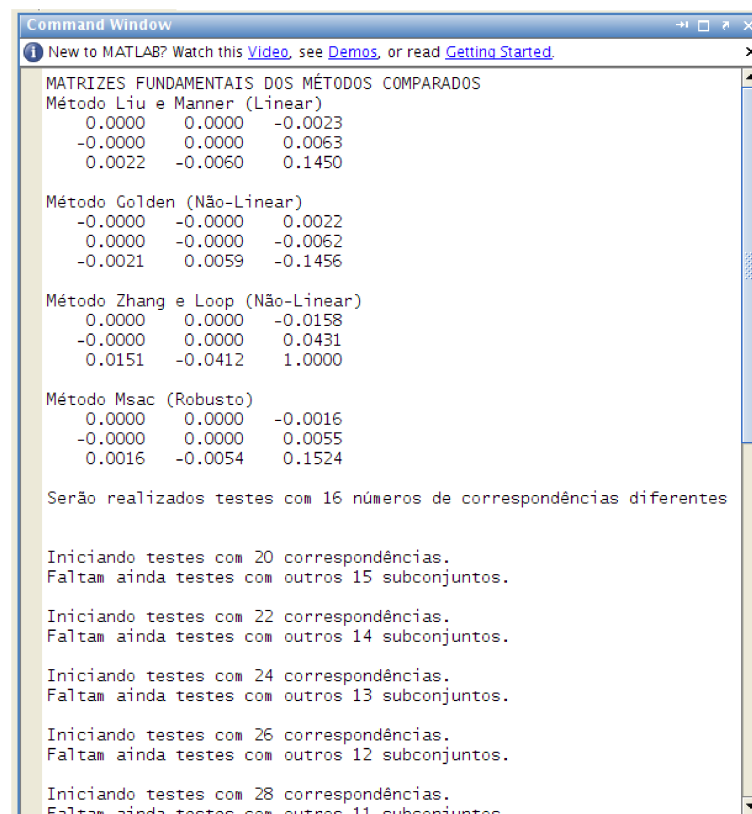


Figura 3.14: Resultados Textuais Gerados pelo Novo Ambiente da Simulação da Estimação da Matriz Fundamental no Ambiente Matlab

fundamental pode ser decomposta em seus epipolos e na homografia epipolar. Contudo, a homografia epipolar pertence ao espaço \mathbf{P}^2 , que não possui nenhuma

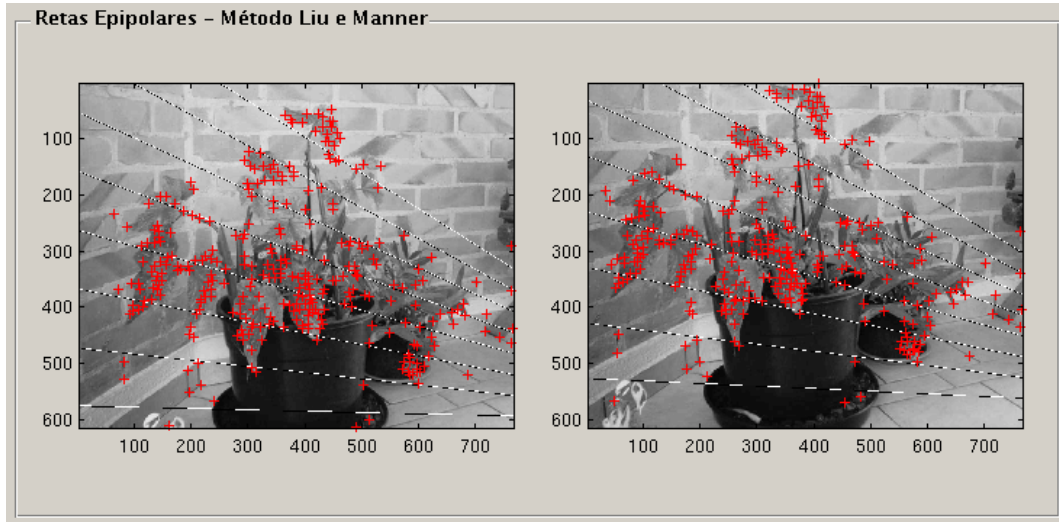


Figura 3.15: Exibição das Retas Epipolares Calculadas a partir da Estimação da Matriz Fundamental pelo Método Liu e Manner

métrica, tornando difícil definir uma medida de ajuste. Já os epipolos, apesar de poderem ser considerados parâmetros euclidianos, são muito suscetíveis a ruído e, por isso, muito difíceis de serem estimados com exatidão, sobretudo quando tendem ao infinito (LUONG; FAUGERAS, 1998). Além disso, o desempenho dos métodos é muito dependente de fatores, tais como, quantidade de correspondências utilizadas na estimação, distribuição de tais correspondências no plano de imagem e localização dos epipolos. Neste trabalho, é utilizada uma metodologia de testes semelhante à realizada por Hartley (1997). A mesma foi implementada no ambiente computacional proposto, ou seja:

a) Em cada experimento, são utilizadas N correspondências de pontos. Em todos os casos, considera-se que o conjunto de correspondências possua ruído com distribuição gaussiana de média zero e variância σ .

b) Dentre as j correspondências diferentes, S_j subconjuntos são selecionados aleatoriamente. Todos os subconjuntos possuem a mesma quantidade, X_j , de correspondências.

c) A cada iteração, cada método a ser avaliado utiliza todos os S_j subconjuntos escolhidos (um por vez) e estima as matrizes, \mathbf{F}_j , diferentes. Além disso, para cada método, é calculado o seguinte resíduo

$$\bar{r}_j^2 = \frac{1}{N} \sum_{i=1}^N d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{F}_j \tilde{\mathbf{m}}_{1i}) + d^2(\tilde{\mathbf{m}}_{1i}, \mathbf{F}_j^T \tilde{\mathbf{m}}_{2i}).$$

d) Após os S_j subconjuntos serem aplicados, a mediana \bar{R}_l^2 é calculada para

cada método, onde

$$\overline{R}_l^2 = \text{mediana}\{\overline{r}_1^2, \dots, \overline{r}_N^2\}.$$

e) Em seguida, o número de correspondências, X_j , é aumentado e outro subconjunto é escolhido aleatoriamente. Isso ocorre até que S_j subconjuntos sejam escolhidos. No total, X_j assume valores diferentes (dentro o intervalo de X_1 a X_j). Dessa forma, para cada par de imagens, cada método é testado $\sum_{i=1}^j X_j$ vezes, com diferentes números de correspondências.

Com essa metodologia, comparando-se os resíduos \overline{R}_l^2 obtidos por métodos diferentes, tem-se uma boa medida de qual deles estimou a matriz fundamental que melhor ajustou-se ao conjunto total de correspondências, ou seja, o método com menor resíduo é o que possui o melhor resultado (ZHANG, 1998). Dentre todos os métodos para calcular a matriz fundamental, o método proposto por Levenberg-Marquardt (LEVENBERG, 1944; MARQUARDT, 1963) foi utilizado sempre que um algoritmo de otimização era necessário, com a exceção ao método da Máxima Verossimilhança (Golden), no qual foi utilizado o método de Levenberg-Marquardt Dividido (FRANÇA et al., 2008).

Diversos testes foram realizados para avaliar, na prática, a exatidão dos métodos e compará-los. Na figura 3.14, é mostrada a execução da simulação da estimação da matriz fundamental pelo ambiente proposto, com as saídas no ambiente Matlab, na qual são apresentadas as matrizes fundamentais e os passos de testes realizados, de acordo com as configurações determinadas para a simulação. É notório que a proposta do novo ambiente facilita e traz uma motivação maior para a execução do processo de estimação da matriz fundamental, assim como a visualização dos resultados de formas textuais e gráficas, possibilitando ao usuário, o entendimento intuitivo da prática desse importante tópico da visão computacional, estimulando o aprendizado crítico. No próximo capítulo, são apresentados os resultados experimentais, no qual são comparados os principais métodos para estimar a matriz fundamental, utilizando-se do ambiente proposto.

4 Resultados e Discussões

Este capítulo apresenta as discussões relativas aos resultados obtidos das simulações realizadas para comparar a execução dos métodos para estimar a matriz fundamental, utilizando o ambiente computacional proposto, a partir de dados advindos de imagens reais e os gerados sinteticamente pelo próprio usuário.

4.1 Resultados com Imagens Reais

Para uma análise criteriosa, o ambiente computacional proposto fornece um canal de resultados de forma gráfica e textual. Os gráficos são gerados para uma análise do resíduo e número de iterações do conjunto de informações, o que caracterizando a exatidão e o custo computacional de cada método avaliado. O conjunto de testes com imagens reais foi realizado com as imagens mostradas na figura 4.1. Para obtenção dos resultados, seguiu-se os padrões definidos nos capítulos anteriores e por meio do processo natural para a estimação da matriz fundamental. As simulações foram baseadas no critério definido na seção 3.6 do capítulo 3.



Figura 4.1: Imagens Utilizadas nos Testes Realizados: a) câmera 1 (esquerda);
b) câmera 2 (direita)

Para tanto, foram utilizados 5 ensaios para cada subconjunto, com pontos selecionados aleatoriamente, para todos os métodos avaliados. Contudo, foram

gerados 31 subconjuntos diferentes, sendo o primeiro com 20 e o último com 50 correspondências. Os métodos não-lineares avaliados utilizaram-se de configurações padrão do *software*, que consiste em uma tolerância de 10^{-6} e um limite de 100 iterações. Já nos métodos robustos, o percentual de um subconjunto ser sorteado foi de 99%, o número de pontos do subconjunto é 8, o limiar do resíduo é de 4, que permite as comparações para detectar as falsas correspondências. O percentual de falsas correspondências é de 30%. Nota-se nos resultados obtidos, que o método não-linear Golden (Máxima Verossimilhança) (HARTLEY; ZISSERMAN, 2003) possui os melhores resultados no critério de exatidão em relação ao demais não-lineares. Já o método robusto MSAC (TORR; ZISSERMAN, 1998), resultou, para todos os subconjuntos analisados, os melhores resultados em comparação à todos os métodos avaliados, mas com resultados muito próximos ao RANSAC (FISCHLER; BOLLES, 1981) e, como utiliza-se de um método linear para o cálculo da matriz fundamental, também não necessita de iterações para chegar aos números finais. Já, para os métodos lineares o método proposto por TRAJKOVIC (TRAJKOVIC; HEDLEY, 1997), obteve os melhores resultados em termos de exatidão e custo computacional. No entanto, com resultados muito próximo aos demais métodos lineares. Na figura 4.2, é mostrada a simulação da compara-

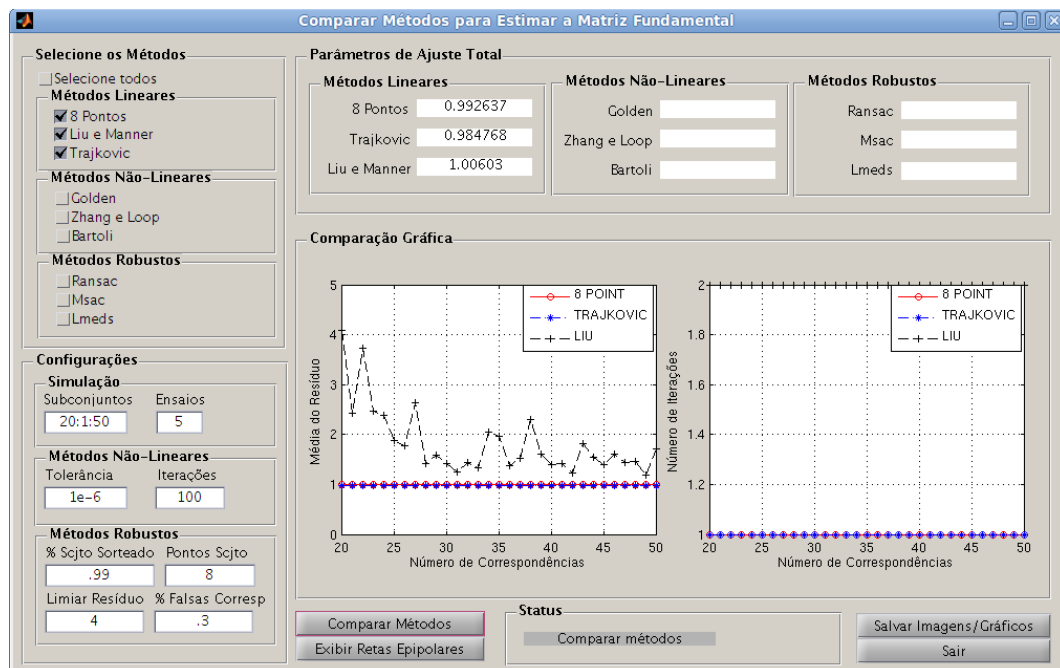


Figura 4.2: Resultados da Simulação da Matriz Fundamental dos Métodos Lineares com Imagens Reais

ção entre métodos lineares para estimar a matriz fundamental. Na figura 4.3, é mostrada a simulação da comparação entre métodos não-lineares para estimar a matriz fundamental. Na figura 4.4, é mostrada a simulação da comparação entre

métodos robustos para estimar a matriz fundamental.

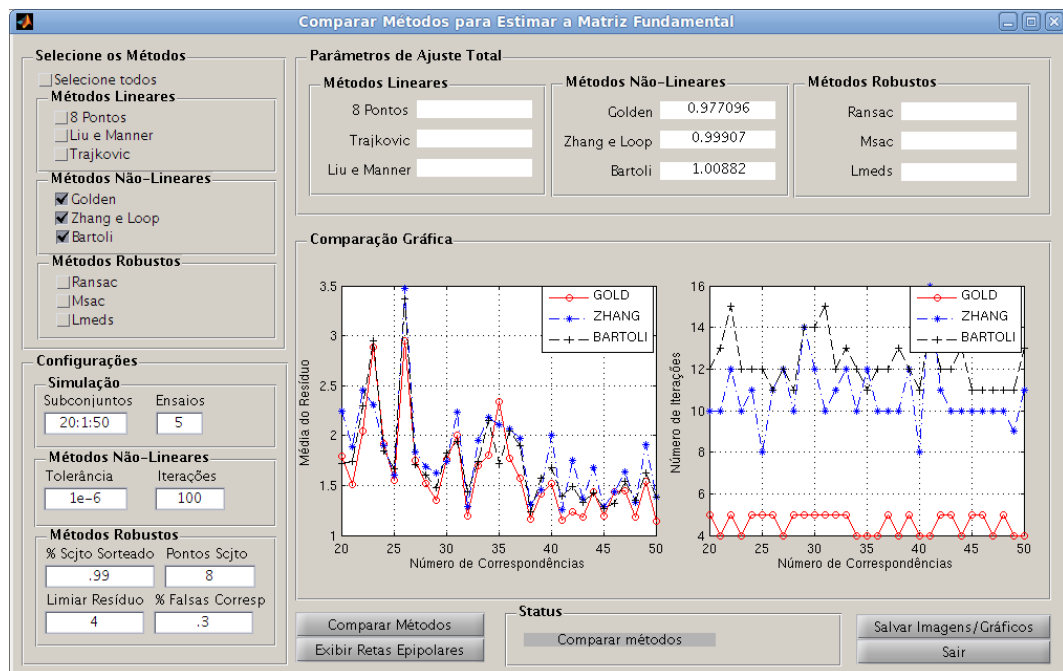


Figura 4.3: Resultados da Simulação da Matriz Fundamental dos Métodos Não-Lineares com Imagens Reais

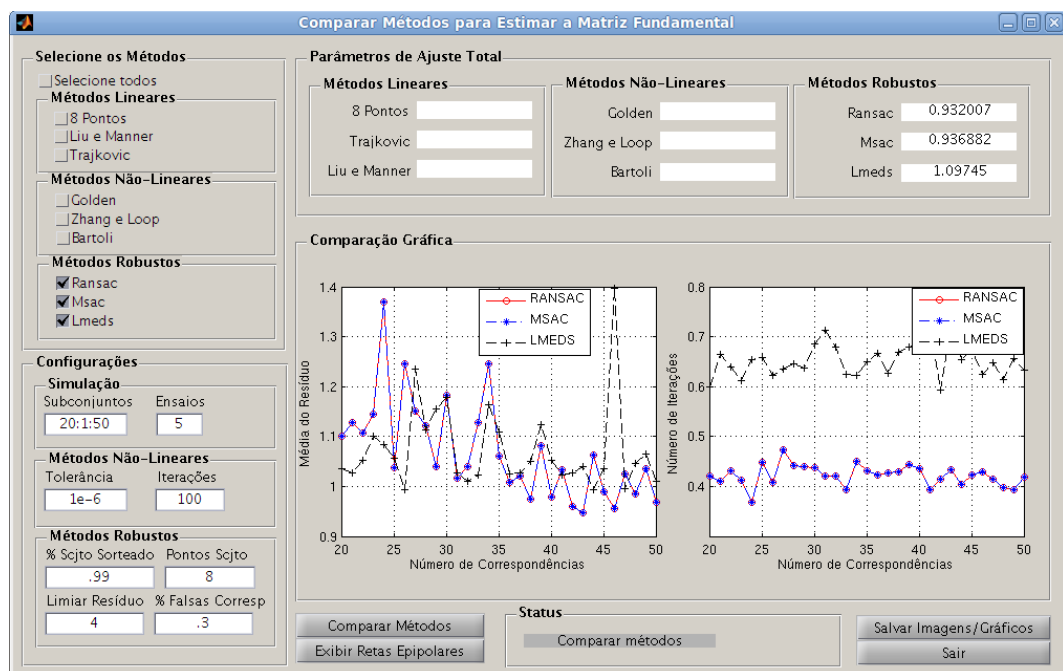


Figura 4.4: Resultados da Simulação da Matriz Fundamental dos Métodos Robustos com Imagens Reais

4.2 Resultados com Dados Sintéticos

A simulação para calcular a matriz fundamental com dados sintéticos, partiu-se da geração do “cubo” sintético gerado a partir da funcionalidade mostrada na figura 4.5. Nessa, nota-se que a largura da face do cubo tem 100 cm, o número

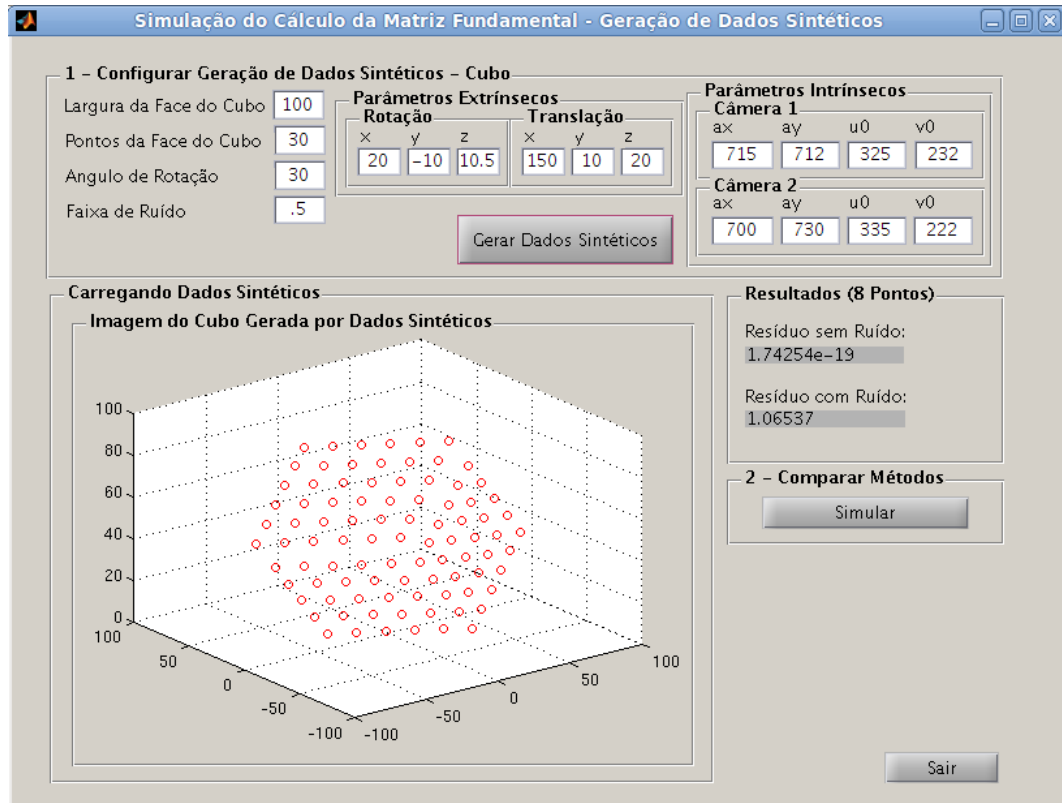


Figura 4.5: Interface de Simulação de Dados Sintéticos para Simulação do Cálculo da Matriz Fundamental

de pontos em cada face do “cubo” é de 30 e o ângulo de rotação do cubo é de 30 graus. No primeiro experimento, foi adicionado um ruído de variância (σ) igual a 0,5 de média zero. Os parâmetros intrínsecos são definidos como o padrão do ambiente computacional, para a câmera 1: $\alpha = 715$, $\beta = 712$, que são fatores de escala e $\mathbf{u}_0 = 325$, $\mathbf{v}_0 = 232$, para a câmera 2: $\alpha = 700$, $\beta = 730$, que são fatores de escala e $\mathbf{u}_0 = 335$, $\mathbf{v}_0 = 222$. Já os parâmetros extrínsecos são definidos para rotação: $\mathbf{x} = 20$, $\mathbf{y} = -10$ e $\mathbf{z} = 10,5$, que são valores associados aos ângulos. Para translação os ângulos de $\mathbf{x} = 150$, $\mathbf{y} = 10$ e $\mathbf{z} = 20$. Na figura 4.6, são observadas as comparações entre um método de cada classificação. No entanto, nota-se que o método não-linear da Golden (Máxima Verossimilhança) (HARTLEY; ZISSERMAN, 2003) possui uma melhor exatidão em termos de resíduo, mas tem um maior custo computacional em comparação aos demais métodos comparados. O método linear de Liu (LIU; MANNER, 2003) teve um resultado muito próximo ao método robusto MSAC (TORR; ZISSERMAN, 1998).

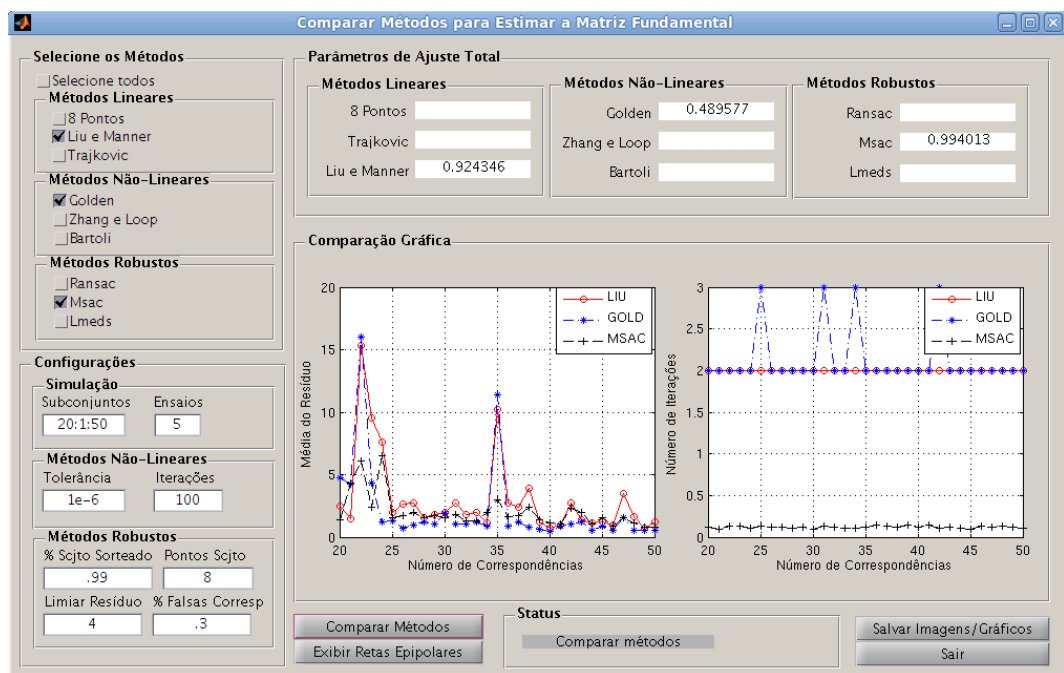


Figura 4.6: Resultados da Simulação da Matriz Fundamental com um Método de cada Classificação com Dados Sintéticos

5 Conclusão

Neste capítulo final do trabalho são apresentados os comentários das principais implicações diretas dos resultados obtidos neste trabalho. Os trabalhos futuros também são mencionados e há comentários das restrições que o projeto possui. Também são relatados os pontos em que houveram facilidade e dificuldade no desenvolvimento deste trabalho.

Apesar de existirem alguns *toolboxes* e ambientes com propósitos similares na literatura, este trabalho preenche uma lacuna existente na estimação da matriz fundamental, uma vez que, com o conjunto de ferramentas propostas, juntamente com o ambiente computacional interativo, é possível a geração de testes com configurações personalizadas, com os mais importantes métodos para o cálculo da matriz fundamental, fornecendo resultados a uma análise detalhada de forma intuitiva. Entretanto, o ambiente computacional mostrou-se bastante confiável em relação aos resultados apresentados e seu maior objetivo foi alcançado devido a alta interatividade com o usuário. Em testes realizados, os algoritmos não-lineares demonstraram os melhores resultados, embora tenham um maior custo computacional, baseado em iterações. Isso pode afetar diretamente o desempenho de um produto final que necessite de processamento em tempo real. Neste trabalho, a questão precisão do resultado dos algoritmos não foi observada na sua totalidade, por se tratar de um ponto sobre o qual já existem diversos estudos na literatura.

As maiores dificuldades encontradas no trabalho estão associadas à complexidade matemática que envolvem os métodos para o cálculo da matriz fundamental, devido a necessidade de uma alta abstração. As facilidades encontradas foram relacionadas ao ambiente de programação Matlab e também pela experiência, de trabalhos anteriores, do grupo de pesquisa de Visão Computacional da Universidade Estadual de Londrina.

Como trabalhos futuros, é interessante criar um módulo para inserção de novos métodos, para estimar a matriz fundamental, de forma dinâmica, ou seja, pela própria interface gráfica, com a finalidade de obter a comparação com os métodos já implementados. Atualmente isso é possível, mas o trabalho tem que

ser manual, ou seja, deve-se alterar o código fonte. Outro projeto interessante, que é relacionado ao tema, seria produzir uma ambiente computacional para simular a calibração de câmeras, por meio de dados reais e sintéticos, com os principais métodos proposto na literatura.

Anexo A – Artigo para Publicação - Um Ambiente Interativo para o Ensino da Geometria Epipolar

Resumo

Na visão binocular, quando o sistema não está calibrado, as informações geométricas das câmeras estão concentradas em uma matriz quadrada de ordem 3, conhecida como matriz fundamental. Esta torna todas as operações da visão binocular menos complexas e pode ser estimada a partir de um conjunto de correspondência de pontos. Devido a grande importância dessa matriz, existem diversos métodos disponíveis na literatura para estimá-la. O presente trabalho apresenta um ambiente interativo para MATLAB que permite a comparação entre esses métodos. Os algoritmos são divididos em três tipos: lineares, não-lineares e robustos. Por sua vez, a matriz fundamental é estimada a partir de um conjunto de correspondência de pontos extraídos de imagens fornecidas pelo próprio usuário. Todas as etapas envolvidas no processo são apresentadas de forma bastante intuitiva, permitindo ao estudante escolher quais métodos serão utilizados na estimação, bem como os parâmetros de configuração de cada um dos algoritmos. Isto, além de permitir o entendimento dos mesmos, facilita a confrontação dos métodos em termos de exatidão e número de iterações.

Palavras-chave: Correspondência de Pontos, Visão Computacional, Ensino de Engenharia, Matriz Fundamental.

A.1 Introdução

A visão computacional é uma área de estudo de grande importância, pois, sem nenhum contato físico, permite recuperar uma grande quantidade de informações. Em especial, quando informações relacionadas a distâncias e ângulos são necessárias, um conjunto binocular deve ser utilizado. Este é composto por duas câmeras que capturam pares de imagens de uma mesma cena, mas de pontos de

vista distintos. Por sua vez, quando o conjunto binocular não está calibrado, a única informação disponível é a geometria epipolar, que é completamente caracterizada através de uma matriz 3×3 , conhecida como matriz fundamental. Esta torna todas as operações envolvidas menos complexas e pode ser estimada a partir de um conjunto de correspondência de pontos. O presente trabalho apresenta um ambiente interativo para ensino da geometria epipolar baseado no MATLAB, no qual o estudante pode visualizar vários aspectos importantes dessa geometria. Além disso, todas as etapas envolvidas na estimação da matriz fundamental podem ser acompanhados passo a passo.

O MATLAB é um ambiente de programação que possui uma linguagem de alto nível. Tal linguagem permite resolver problemas, principalmente os que envolvem computação numérica, mais rapidamente que outras linguagens mais tradicionais como, por exemplo, C, C++ e FORTRAN. Os problemas são resolvidos em um ambiente fácil de usar, onde as soluções são expressas de maneira clara e intuitiva, possibilitando a sua utilização até mesmo por estudantes. Além disso, a estrutura do MATLAB pode ser facilmente estendida através da inclusão de funções correlacionadas e agrupadas em pacotes (*toolboxes*). De fato, existem inúmeros pacotes disponíveis comercialmente e outros com licenças livres. Por conta disso, ao longo de vários anos, o MATLAB tem sido utilizado com sucesso no ensino de diversas áreas da engenharia como, por exemplo, controle de sistemas (KNUDSEN, 2006; URAN; JEZERNIK, 2008; TEIXEIRA; ASSUNCAO; COVACIC, 2007; ALIANE, 2010), processamento digital de sinais (CAVICCHI, 2005), eletromagnetismo (SAGNARD, 2004; MAGISTRIS, 2005), máquinas elétricas (AYASUN; NWANKPA, 2005), robótica (POTA, 1992) e até mesmo sistemas de potência (CHAU, 1996; AYASUN; NWANKPA; KWATNY, 2006).

Já há na literatura outros pacotes de funções para MATLAB voltados à solução de problemas na área de visão computacional e robótica. Dentre esses, destacam-se os criados por Peter Corke (CORKE, 2005, 1996). Esses são coleções de funções que tem sido desenvolvidas por mais de uma década e abrangem algoritmos desde processamento de imagens até manipulação de braços robóticos. Além disso, as funções são versáteis o suficiente para serem utilizadas em aplicações de tempo real. No entanto, a sua utilização no controle de manipuladores é muito dependente do hardware usado pelo autor. Além disso, para utilização do pacote, é exigido do usuário um elevado nível de conhecimento prévio (tanto dos algoritmos utilizados como do ambiente do MATLAB). Por isso, eles não são indicados para uso na educação. De fato, dentre os pacotes de código aberto disponíveis para MATLAB, essa é uma característica predominante dentre os

voltados para visão de máquina (ASTROM et al., 1997) e robótica (VILA-ROSADO; DOMINGUEZ-LOPEZ, 2005). Por exemplo, Kanwal, Arif e Majeed (2003) disponibilizaram um conjunto de funções relacionadas a visão binocular, no qual imagens são capturadas, pontos são segmentados, a matriz fundamental é estimada e até mesmo uma reconstrução euclidiana de um objeto é realizada após a calibração do conjunto binocular. Contudo, trata-se apenas de uma coleção de algoritmos, ou seja, dentre outras limitações: a) não há nenhuma interface gráfica com o usuário que torne o uso das funções, bem como a sequência correta de execução; b) existe apenas um único algoritmo para cada problema proposto (o que torna difícil a percepção da importância de determinados aspectos do código devido a dificuldade de visualizar a solução do problema sem eles), e; c) o resultado da reconstrução é apresentado em um aplicativo externo ao MATLAB, baseado no padrão VRML (*Virtual Reality Modeling Language*).

A proposta apresentada neste trabalho assemelha-se ao proposto no recente artigo de Samper et al. (2010). Contudo, o trabalho de Samper concentra-se apenas na calibração das câmeras, sem abordar a geometria epipolar. Para isso, vários métodos de calibração são suportados, uma interface amigável foi apresentada e um conjunto de experimentos foi proposto (SAMPER et al., 2010). Contudo, o aplicativo não trabalha com calibração binocular e as funções que compõem o código dos algoritmos não podem ser executadas externamente, ou seja, de forma independente do ambiente gráfico proposto. Por outro lado, no presente trabalho, todas as funções utilizadas no cálculo da matriz fundamental, além das outras funções de apoio, podem ser executadas diretamente na linha de comando do MATLAB. Essas são divididas por categorias e possuem documentação própria. Isso foi feito para possibilitar a adaptação do pacote às necessidades dos educadores. Apesar disso, o pacote de funções possui uma interface com o estudante bastante amigável, na qual, a partir de um conjunto de correspondência de pontos extraídos de imagens fornecidas pelo próprio estudante, a matriz fundamental é estimada por diferentes métodos. Esses métodos são divididos em três categorias: lineares, não-lineares e robustos. Além disso, todas as etapas envolvidas no processo são apresentadas de forma bastante intuitiva, permitindo ao estudante escolher quais métodos serão utilizados na estimação, bem como os parâmetros de configuração de cada um deles. Isto, além de permitir o entendimento dos mesmos, facilita a confrontação dos métodos em termos de exatidão e número de iterações.

A.2 Notação

As coordenadas homogêneas de um ponto $\mathbf{m} = [u, v]^T$ são representadas por $\tilde{\mathbf{m}}$, isto é, $\tilde{\mathbf{m}} = [u, v, 1]^T$. Índices, se houverem, indicam a posição do ponto em um conjunto de pontos. Assim, com a notação adotada, a relação entre um ponto 3D, \mathbf{M} , e sua projeção, \mathbf{m} , considerando o modelo de uma câmera *pinhole* (HARTLEY; ZISSERMAN, 2003), é dada por

$$\tilde{\mathbf{m}} \simeq \mathbf{A} [\mathbf{I} \ \mathbf{0}] \tilde{\mathbf{M}}, \quad (\text{A.1})$$

onde \simeq indica que ambos os lados da equação podem diferir por uma constante não-nula, $\mathbf{0}$ é o vetor nulo e \mathbf{I} é a matriz identidade. Em especial, \mathbf{A} é uma matriz 3×3 que concentra todos os parâmetros intrínsecos da câmera.

As câmeras que pertencem a um conjunto binocular são referenciadas neste artigo como “câmera 1” e “câmera 2”. Sem perda de generalidade, supõe-se sempre que a “câmera 1” está no centro do sistema de coordenadas do ambiente. Além disso, no caso de matrizes, um índice indica a qual câmera elas se referem. Por exemplo, \mathbf{A}_1 são os parâmetros intrínsecos da “câmera 1”, enquanto que \mathbf{A}_2 é associada a “câmera 2”. Dessa forma, considerando a equação (A.1), para um conjunto binocular, tem-se

$$\tilde{\mathbf{m}}_1 \simeq \mathbf{A}_1 [\mathbf{I} \ \mathbf{0}] \tilde{\mathbf{M}} \quad \text{e} \quad \tilde{\mathbf{m}}_2 \simeq \mathbf{A}_2 [\mathbf{R} \ \mathbf{t}] \tilde{\mathbf{M}}, \quad (\text{A.2})$$

onde \mathbf{R} é uma matriz 3×3 que representa a rotação da câmera 1 em relação a 2, \mathbf{t} é um vetor que equivale a translação do centro da câmera 1 com relação a 2 e, geralmente, são referenciados como os parâmetros extrínsecos da câmera.

A equação anterior muitas vezes é expressa em função das matrizes de projeção das câmeras. Neste caso,

$$\mathbf{P}_1 = \mathbf{A}_1 [\mathbf{I} \ \mathbf{0}] \quad \text{e} \quad \mathbf{P}_2 = \mathbf{A}_2 [\mathbf{R} \ \mathbf{t}]. \quad (\text{A.3})$$

A.3 Preliminares

Quando o conjunto binocular não está calibrado, a geometria epipolar é a única restrição geométrica disponível. Tal geometria já foi descrita em numerosos artigos como, por exemplo, os trabalhos de Armangué e Salvi (2003) e Zhang (1998). Contudo, basicamente, a geometria epipolar pode ser entendida se for considerado o caso de duas câmeras, como apresentado na figura A.1. Nesta, \mathbf{C}_1 e \mathbf{C}_2 são, respectivamente, os centros ópticos da primeira e segunda câmeras. Então, dado

um ponto \mathbf{m}_1 na primeira imagem, I_1 , o ponto correspondente, \mathbf{m}_2 , na segunda imagem, I_2 , está restrito a uma reta chamada “reta epipolar” de \mathbf{m}_1 , representada na figura por l_2 . A reta l_2 é a intersecção do plano Π , definido por \mathbf{M} , \mathbf{C}_1 e \mathbf{C}_2 (chamado de plano epipolar), com o plano I_2 . Isto acontece porque o ponto \mathbf{m}_1 pode corresponder a qualquer ponto da reta $\overline{\mathbf{C}_1\mathbf{M}}$ e a projeção de $\overline{\mathbf{C}_1\mathbf{M}}$ em I_2 é a reta l_2 . Além disso, observa-se que todas as retas epipolares dos pontos de I_1 passam através de um ponto comum, \mathbf{e}_2 , em I_2 . Este ponto é conhecido como “epipolo”. O ponto \mathbf{e}_1 é a intersecção da reta $\overline{\mathbf{C}_1\mathbf{C}_2}$ com o plano I_2 . Finalmente, pode-se facilmente observar a simetria da geometria epipolar. O correspondente em I_1 de cada ponto \mathbf{m}_{2i} , sobre a reta l_{2i} , precisa pertencer a reta epipolar l_{1i} , que é a intersecção do mesmo plano Π_i com o plano I_1 . Todas as retas epipolares formam um conjunto contendo o epipolo \mathbf{e}_1 , que é a intersecção da reta $\overline{\mathbf{C}_1\mathbf{C}_2}$ com o plano I_1 .

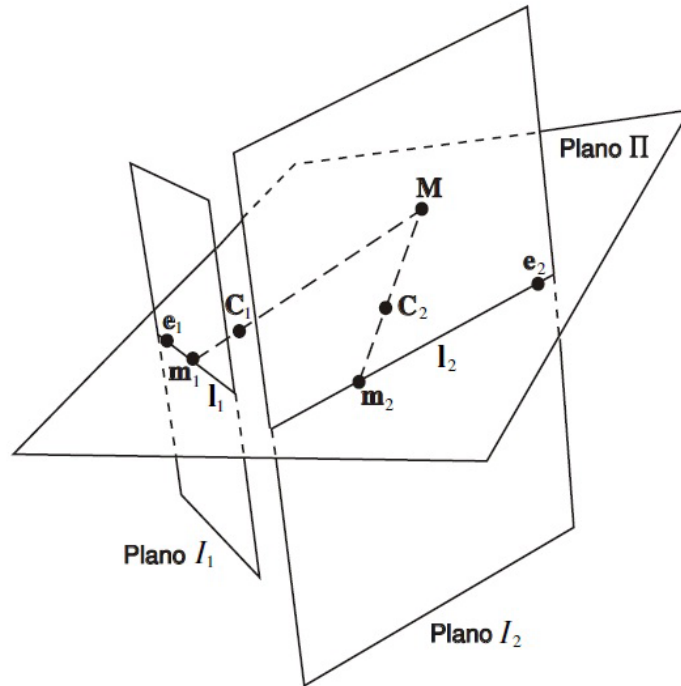


Figura A.1: Esboço da geometria epipolar.

Normalmente, todas as restrições impostas pela geometria epipolar são resumidas na seguinte equação,

$$\tilde{\mathbf{m}}_2^T \mathbf{F} \tilde{\mathbf{m}}_1 = 0, \quad (\text{A.4})$$

onde \mathbf{F} é uma matriz homogênea 3×3 , de posto dois e com sete graus de liberdade, conhecida como a “matriz fundamental” do conjunto binocular.

A equação (A.4) é uma restrição por trás de quaisquer duas imagens se estas são projeções em perspectiva de uma mesma cena. Geometricamente, $\mathbf{F}\tilde{\mathbf{m}}_1$ define a reta epipolar do ponto \mathbf{m}_1 no plano I_2 . Assim, a equação (A.4) não diz nada

além de que o ponto correspondente de \mathbf{m}_1 (em I_2), ou seja, \mathbf{m}_2 , está sobre a sua reta epipolar $\mathbf{l}_2 = \mathbf{F}\tilde{\mathbf{m}}_1$.

Da equação (A.4), vê-se que, dado um conjunto de correspondências ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$), é possível estimar a matriz fundamental. Como \mathbf{F} é homogênea e tem sete graus de liberdade, são necessárias no mínimo sete correspondências de pontos para estimar \mathbf{F} unicamente (ARMANGUÉ; SALVI, 2003). Contudo, devido ao ruído, na prática, são necessárias oito ou mais correspondências para estimar essa matriz com exatidão.

Quando o conjunto binocular não está calibrado e a matriz fundamental é desconhecida, não há nenhuma restrição geométrica disponível. Assim, dado um ponto na primeira imagem, o seu correspondente pode ser qualquer ponto da segunda imagem. Por isso, estabelecer correspondência de pontos nessas condições é uma tarefa de grande complexidade. Assim, para simplificar, inicialmente, são segmentados um conjunto de pontos em ambas as imagens. Isto reduz significativamente o espaço de busca.

Normalmente, os pontos segmentados são cantos, pois estes são (quase) invariantes a deslocamentos Euclidianos, enquanto que, por exemplo, planos e retas podem até desaparecer, dependendo do deslocamento de uma câmera em relação a outra. No presente trabalho, o algoritmo de detecção de cantos utilizado é o proposto por Harris e Stephens (1988). O algoritmo de Harris é muito popular devido a sua grande robustez com relação a rotações, diferenças de iluminação, ruído, etc. Este baseia-se na função de auto-correlação de um sinal, onde esta função mede mudanças locais do sinal em caminhos com pequenos deslocados e em todas as direções.

Uma vez que cantos tenham sido segmentados em ambas as imagens, um algoritmo de correspondência de cantos pode ser utilizado. Neste trabalho, o algoritmo de correspondência de Zhang et al. (1995) foi implementado computacionalmente. Este estabelece um conjunto de correspondência de pontos entre duas imagens sem fazer nenhuma suposição com relação a posição de uma câmera em relação a outra. Assim, as imagens podem ter sido capturadas por duas câmeras diferentes ou por uma mesma câmera, mas em instantes de tempo e posições diferentes. Inicialmente, baseado no grau de semelhança entre os cantos segmentados (calculado através de correlação cruzada) e usando técnicas de relaxação, é estabelecido um conjunto inicial de correspondência de pontos dentre os cantos segmentados. Em seguida, a matriz fundamental é estimada de forma robusta, usando o algoritmo LMedS (seção A.4.3.3). Com a matriz fundamental estimada,

a restrição epipolar é utilizada para descartar falsas correspondências.

Uma vez que o conjunto de correspondência de pontos tenha sido estabelecido, a matriz fundamental pode ser estimada usando um dos diversos métodos disponíveis no ambiente implementado.

A.4 Estimação da Matriz Fundamental

Nas seções seguintes, os algoritmos de estimação da matriz fundamental implementados no MATLAB e disponíveis no ambiente educacional desenvolvido são descritos. Estes foram classificados em três categorias: lineares, não-lineares e robustos.

A.4.1 Métodos Lineares

Considerando $\tilde{\mathbf{m}}_{1i} = [u_{1i}, v_{1i}, 1]^T$ e $\tilde{\mathbf{m}}_{2i} = [u_{2i}, v_{2i}, 1]^T$, a equação (A.4) pode ser reescrita como sendo

$$\mathbf{u}_i \mathbf{f} = 0, \quad (\text{A.5})$$

onde $\mathbf{u}_i = [u_{1i}u_{2i}, u_{1i}v_{2i}, u_{1i}, v_{1i}u_{2i}, v_{1i}v_{2i}, v_{1i}, u_{2i}, v_{2i}, 1]^T$,

$\mathbf{f} = [F_{11}, F_{21}, F_{31}, F_{12}, F_{22}, F_{32}, F_{13}, F_{23}, F_{33}]^T$ e F_{ij} é o elemento da i -ésima linha e j -ésima coluna de \mathbf{F} .

Agora, com n correspondências de pontos, tem-se

$$\mathbf{U}_n \mathbf{f} = \mathbf{0}, \quad (\text{A.6})$$

onde $\mathbf{U}_n = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n]^T$ e $\mathbf{0}$ é o vetor nulo.

De uma forma geral, a grande maioria dos métodos lineares de estimação da matriz fundamental baseiam-se na equação (A.6). A principal vantagem destes métodos é a simplicidade. Apesar disso, se alguns cuidados forem tomados, estes fornecem uma boa estimação de \mathbf{F} a um custo computacional bastante reduzido. Contudo, a matriz fundamental obtida por estes métodos possui geralmente posto três. Por isso, como discutido a seguir, deve-se impor a restrição de posto dois a posteriori.

A.4.1.1 Método dos 8 pontos

Na presença de ruído, não existe vetor \mathbf{f} que satisfaça a equação (A.6) exatamente. Contudo, com $n \geq 8$ correspondências de pontos, pode-se encontrar \mathbf{f} resolvendo

$$\min_{\mathbf{f}} \mathbf{U}_n \mathbf{f}, \text{ sujeito a } \|\mathbf{f}\| = 1. \quad (\text{A.7})$$

O método baseado na resolução da equação (A.7) é conhecido como “método dos 8 pontos” e é o método mais simples de estimar a matriz fundamental a partir de um conjunto de correspondências de pontos.

Como a maioria dos métodos lineares, a desvantagem evidente do algoritmo de 8 pontos é não garantir a restrição de posto dois. Assim, esta tem que ser imposta a posteriori. A forma mais conveniente de fazer isto é substituir a matriz estimada, $\widehat{\mathbf{F}}$, pela matriz \mathbf{F} que minimiza $\|\widehat{\mathbf{F}} - \mathbf{F}\|$, sujeito a $\det(\mathbf{F}) = 0$. Isto é equivalente a fazer $\mathbf{F} = \widehat{\mathbf{F}}$ e substituir o menor autovalor de \mathbf{F} por zero (FAUGERAS; LUONG, 2001).

O resultado do algoritmo dos 8 pontos depende fortemente do sistema de coordenadas utilizado para representar as coordenadas dos pontos \mathbf{m}_{1i} e \mathbf{m}_{2i} . Normalmente, as coordenadas destes pontos são dadas em relação ao canto superior esquerdo da imagem. Contudo, Hartley (1997) demonstrou que uma simples mudança neste referencial torna o problema melhor condicionado e, portanto, conduz a um resultado muito mais exato. Esta mudança pode ser realizada por meio de transformações projetivas aplicadas ao conjunto de correspondências. Do trabalho de Hartley (1997), sabe-se que essas transformações devem ser tais que, em cada uma das imagens, independentemente, o centróide do conjunto de pontos esteja na origem e a distância média de um ponto à origem seja $\sqrt{2}$. Esta simples mudança produz um aumento significativo na exatidão do algoritmo. Além disso, o custo computacional acrescentado ao problema devido a esta normalização é desprezível.

A.4.1.2 Método de Trajkovic

Este método também encontra \mathbf{F} resolvendo a equação (A.6). A diferença está na imposição da restrição de posto dois. O algoritmo de Trajkovic e Hedley (1997) impõe esta restrição calculando, inicialmente, o epipolo da matriz fundamental que resolve o problema (A.7). Em seguida, encontra \mathbf{F} que tem o mesmo epipolo e também tem posto dois. Segundo Trajkovic e Hedley (1997), este método tem performance superior ao de Hartley (1997), especialmente, quando há apenas um

número pequeno de correspondências de pontos disponíveis para montar a matriz \mathbf{U}_n .

A.4.1.3 Minimização da Função Sampson

É um método linear e iterativo proposto por Liu e Manner (2003), que estima a matriz \mathbf{F} que minimiza o “Erro de Sampson” (SAMPSON, 1982) da distância dos pontos às retas epipolares correspondentes. Contudo, faz-se uso da aproximação de primeira ordem para a função de custo geométrica e, portanto, converte o problema em uma minimização linear dos mínimos quadrados. Infelizmente, o método fornece bons resultados apenas quando as câmeras estão dispostas de forma que os epipolos estão próximos ao centro de imagem.

A.4.2 Métodos Não-Lineares

Como já descrito anteriormente, a restrição de posto dois não pode ser embutida a métodos lineares. Além disso, geralmente, tais métodos minimizam critérios sem significado físico. Para contornar essas limitações, algoritmos não-lineares foram propostos. Geralmente, a resolução de tais algoritmos exige uma estimação inicial da matriz fundamental. Devido a seu bom desempenho, o método dos 8 pontos normalizado é normalmente utilizado para isto. Sua função é fornecer uma solução inicial bem próxima de um mínimo local. Dessa forma, a solução dos problemas não-lineares é atingida após poucas iterações.

Em seguida, os métodos não-lineares para cálculo da matriz fundamental implementados neste trabalho e disponíveis no ambiente desenvolvido são descritos resumidamente.

A.4.2.1 Método da Máxima Verossimilhança

Como já discutido, a estimação da matriz fundamental necessita de um conjunto de correspondência de pontos ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$). Devido ao processo de obtenção dessas correspondências, é natural que exista ruído em suas coordenadas. Este ruído degrada a estimação de \mathbf{F} . Para contornar este problema, o método da máxima verossimilhança estima as coordenadas reais dos pontos de cada correspondência ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$), sujeito a equação (A.4). Para isso, além de uma estimação inicial das matrizes de projeção canônicas (HARTLEY; ZISSERMAN, 2003) do conjunto binocular, ou seja,

$$\mathbf{P}_1 = [\mathbf{I} \ \mathbf{0}] \text{ e } \mathbf{P}_2 = [\mathbf{H} \ \mathbf{t}], \quad (\text{A.8})$$

é indispensável também uma estimação inicial dos pontos 3D, \mathbf{M}_i , que satisfazem

$$\tilde{\mathbf{m}}_{1i} = \mathbf{P}_1 \tilde{\mathbf{M}}_i \text{ e } \tilde{\mathbf{m}}_{2i} = \mathbf{P}_2 \tilde{\mathbf{M}}_i. \quad (\text{A.9})$$

Com isso, dado um conjunto com n correspondência de pontos, encontra-se \mathbf{H} , \mathbf{t} e os n pontos \mathbf{M}_i que minimizam

$$\sum_{i=1}^n [d^2(\tilde{\mathbf{m}}_{1i}, \mathbf{P}_1 \tilde{\mathbf{M}}_i) + d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{P}_2 \tilde{\mathbf{M}}_i)], \quad (\text{A.10})$$

onde $d^2(\cdot, \cdot)$ representa o quadrado da distância euclidiana entre dois pontos. Em seguida, obtém-se a matriz fundamental fazendo $\mathbf{F} = [\mathbf{t}]_{\times} \mathbf{H}$.

A minimização da equação anterior envolve a solução de um problema de otimização não-linear. Na visão computacional, o algoritmo de Levenberg-Marquardt (LEVENBERG, 1944; MARQUARDT, 1963) é muito utilizado para solucionar tais problemas. Contudo, a complexidade computacional de tal algoritmo é N^3 , onde N é número de incógnitas a serem estimadas (FRANÇA et al., 2008). A equação (A.10) possui um número excessivo de incógnitas. Felizmente, observando as equações (A.8) e (A.9), fica claro que uma mudança nos elementos de \mathbf{H} ou \mathbf{t} altera as coordenadas dos pontos de todas as correspondências, enquanto que uma alteração nas coordenadas de algum ponto \mathbf{M}_i reflete-se apenas nas coordenadas dos pontos da i -ésima correspondência. Dessa forma, o jacobiano da função definida pela equação (A.10) tem uma estrutura esparsa e existe uma derivação do algoritmo Levenberg-Marquardt, popularizada por Hartley e Zisserman (2003) e referenciada por Levenberg-Marquardt Dividido, que toma vantagem da estrutura do problema para reduzir significativamente a complexidade do algoritmo.

Uma implementação do algoritmo Levenberg-Marquardt Dividido já foi implementada como um *script* para o MATLAB (FRANÇA et al., 2008) e, no presente trabalho, este *script* é utilizado para minimizar (A.10). Com isso, a complexidade do problema é reduzida para apenas N e, dessa forma, o método da máxima verossimilhança constitui um método conveniente e exato para estimação da matriz fundamental.

A.4.2.2 Método de Zhang e Loop

Dado um conjunto com n correspondências de pontos, Luong et al. (1993) propuseram um algoritmo no qual busca-se \mathbf{F} que minimiza

$$\sum_{i=1}^n [d^2(\mathbf{m}_{2i}, \mathbf{F} \tilde{\mathbf{m}}_{1i}) + d^2(\mathbf{m}_{1i}, \mathbf{F}^T \tilde{\mathbf{m}}_{2i})], \quad (\text{A.11})$$

com $d^2(\mathbf{x}, \mathbf{l})$ igual ao quadrado da distância euclidiana entre o ponto \mathbf{x} e a reta \mathbf{l} , e;

$$\mathbf{F} = \begin{bmatrix} 1 & a & -\alpha - \alpha\beta \\ b & c & -b\alpha - c\beta \\ -\lambda - b\mu & -a\lambda - c\mu & F_9 \end{bmatrix}, \quad (\text{A.12})$$

onde $a, b, c, \beta, \alpha, \lambda$ e μ são constantes que dependem de uma estimativa inicial de \mathbf{F} e $F_9 = \alpha(\lambda + b\mu) + \beta(a\lambda + c\mu)$. Por sua vez, este método tem a vantagem de estimar a matriz fundamental com apenas sete incógnitas e impor a restrição de posto dois. Contudo, a parametrização proposta pela equação (A.12) funciona bem apenas nas seguintes condições: os epipolos estão no infinito e a, b e c são todos menores que a unidade. Para contornar esse problema, Loop e Zhang (1999) propuseram transformações projetivas que agem em ambas as imagens e levam as correspondência de pontos para novos sistemas de coordenadas, nos quais essas condições são sempre satisfeitas.

A.4.2.3 Método de Bartoli

Neste método proposto por Bartoli e Sturm (2004), por meio da decomposição de valor singular (SVD), a matriz fundamental é representada como sendo $\mathbf{F} = \mathbf{U} \text{diag}(1, \sigma, 0) \mathbf{V}^T$, onde σ é a razão entre os dois autovalores de \mathbf{F} . Dessa forma, a matriz fundamental pode ser atualizada por meio das seguintes equações

$$\mathbf{U} \leftarrow \mathbf{U}\mathbf{R}(\theta_u), \mathbf{V} \leftarrow \mathbf{V}\mathbf{R}(\theta_v), \sigma \leftarrow \sigma + \delta,$$

onde $\mathbf{R}(\theta_i)$ representa matrizes de rotação baseadas nos ângulos de Euler. Assim, usando um algoritmo não-linear de otimização, é possível ajustar \mathbf{F} , iterativamente, de forma a minimizar a equação (A.11). A vantagem do método é que \mathbf{F} é representada com o mínimo de parâmetros possível (três para o vetor θ_u , três para θ_v e um para δ_σ).

A.4.3 Métodos Robustos

Devido a dificuldade de se estabelecer correspondência de pontos antes da geometria epipolar ser conhecida, espera-se que, em um conjunto de correspondências ($\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i}$), existam erros e ruído nas coordenadas dos pontos \mathbf{m}_{1i} e \mathbf{m}_{2i} . Isto permite classificar as correspondências em boas ou falsas. Normalmente, supõe-se que o ruído presente nas coordenadas dos pontos, \mathbf{m}_{1i} e \mathbf{m}_{2i} , das boas correspondências possui uma distribuição Gaussiana. Isso é uma suposição razoável, pois os erros nas coordenadas são de apenas poucos pixels. Os algoritmos para cálculo

da matriz fundamental são capazes de lidar com o ruído presente nas boas correspondências. Contudo, apenas algumas poucas falsas correspondências presentes no conjunto são capazes de degradar totalmente a estimação de \mathbf{F} . Neste caso, algoritmos robustos conseguem tratar com tais correspondências, detectando e retirando-as do conjunto. Desse modo, ao final do algoritmo, a matriz fundamental pode ser estimada com apenas boas correspondências.

A seguir, os algoritmos robustos implementados e disponibilizados no ambiente desenvolvido são descritos resumidamente.

A.4.3.1 RANSAC

É um dos métodos robustos mais conhecidos na literatura. Basicamente, o RANSAC pode ser resumido como segue. Dado um conjunto com N correspondências $c_i = (\mathbf{m}_{1i} \leftrightarrow \mathbf{m}_{2i})$, escolhe-se, aleatoriamente, n subconjuntos com $m \geq 8$ correspondências. Para cada subconjunto (indexado por j), estima-se a matriz fundamental, \mathbf{F}_j , e calcula-se o resíduo $r(\mathbf{F}_j, c_i)$ de todas as N correspondências, onde

$$r(\mathbf{F}_j, c_i) = d^2(\mathbf{m}_{2i}, \mathbf{F}_j \tilde{\mathbf{m}}_{1i}) + d^2(\mathbf{m}_{1i}, \mathbf{F}_j^T \tilde{\mathbf{m}}_{2i}). \quad (\text{A.13})$$

Em seguida, cada resíduo é comparado a um limiar r_{th} (escolhido empiricamente). Se $r(\mathbf{F}_j, c_i) < r_{th}$, a correspondência c_i é considerada uma boa correspondência. Após n iterações, retém-se a matriz fundamental, \mathbf{F}_{win} , que ajustou-se ao maior número de boas correspondências. Por último, estima-se \mathbf{F} com apenas as boas correspondências, identificadas por $r(\mathbf{F}_{win}, c_i) < r_{th}$.

A.4.3.2 MSAC

O algoritmo RANSAC procura a matriz \mathbf{F}_j que minimiza a função

$$\sum_{i=1}^N \mathcal{F}(r_{ji}),$$

onde $r_{ji} = r(\mathbf{F}_j, c_i)$ e

$$\mathcal{F}(r_{ji}) = \begin{cases} 0, & \text{se } r_{ji} \leq r_{th}, \\ 1, & \text{se } r_{ji} > r_{th}. \end{cases}$$

Na equação anterior, é evidente que, se r_{th} tiver um valor muito elevado, todas as correspondências serão consideradas boas. Neste caso, qualquer matriz \mathbf{F}_j teria a mesma pontuação, ou seja, $\sum_{i=1}^N \mathcal{F}(r_{ji})$ seria sempre igual a N . Por isso, Torr e Zisserman (1998) sugeriram uma discreta alteração na função $\mathcal{F}(r_{ji})$,

ou seja,

$$\mathcal{F}(r_{ji}) = \begin{cases} r_{ji}, & \text{se } r_{ji} \leq r_{th}, \\ r_{th}, & \text{se } r_{ji} > r_{th}. \end{cases}$$

Assim, cada boa correspondência contribui com um valor diferente e proporcional ao seu grau de ajuste a \mathbf{F}_j . Desse modo, mesmo considerando um número igual de boas correspondências, a função $\mathcal{F}(r_{ji})$ deve ter valores diferentes para matrizes fundamentais diferentes.

A.4.3.3 LMEDS

O algoritmo RANSAC e o MSAC só podem ser utilizados quando existe informação sobre a distribuição do ruído inserido no conjunto de correspondências (para que seja determinado o valor de r_{th}). Como discutido na seção anterior, essa informação é utilizada no critério que define se uma correspondência é boa ou não. Caso tal informação não esteja disponível, pode-se ainda estimar \mathbf{F} utilizando o método LMEDS. Este é muito semelhante ao RANSAC. A principal diferença é que uma matriz \mathbf{F}_j é considerada boa quando a mediana do resíduo é pequena. Assim, após \mathbf{F}_j ser estimada, calcula-se a mediana dos resíduos, r_{ji} , de todas as N correspondências. Após n iterações, retém-se a matriz fundamental, \mathbf{F}_{win} , que obteve a menor mediana. Por último, estima-se \mathbf{F} com apenas as boas correspondências, identificadas por $r(\mathbf{F}_{win}, c_i) < \sigma_{th}$, onde, do trabalho de Rousseeuw (1987),

$$\sigma_{th} = 1,4826[1 + 5/(N - m)]\sqrt{r_{med}},$$

onde r_{med} é a menor mediana do resíduo, dentre todos os n subconjuntos.

O algoritmo LMEDS tem uma grande desvantagem, ou seja, ele não consegue estimar a matriz fundamental corretamente se existirem mais de 50% de falsas correspondências.

A.5 A Construção do Ambiente de Simulação para Estimar a Matriz Fundamental

O ambiente computacional proposto neste trabalho tem como característica possibilitar ao usuário inexperiente uma rápida compreensão dos métodos para estimar a matriz fundamental de um conjunto binocular. O desenvolvimento deste ambiente computacional foi baseado no MATLAB, com alguns módulos de programação na linguagem C, por meio das funções MEX (*Matlab External File*). Na sua implementação, procurou-se adotar boas práticas de programação, tais

como a empregabilidade dos conceitos de usabilidade, modularidade e reusabilidade (PRESSMAN, 2006). O *toolbox* está organizado em pacotes e cada pacote possui um conjunto de funções para realizar uma determinada tarefa. No entanto, o usuário executa o processo completo, para estimar a matriz fundamental, por meio do ambiente interativo. Contudo, caso seja necessário, todas as funções utilizadas no cálculo da matriz fundamental, além das outras funções de apoio, podem ser executadas diretamente na linha de comando do MATLAB.

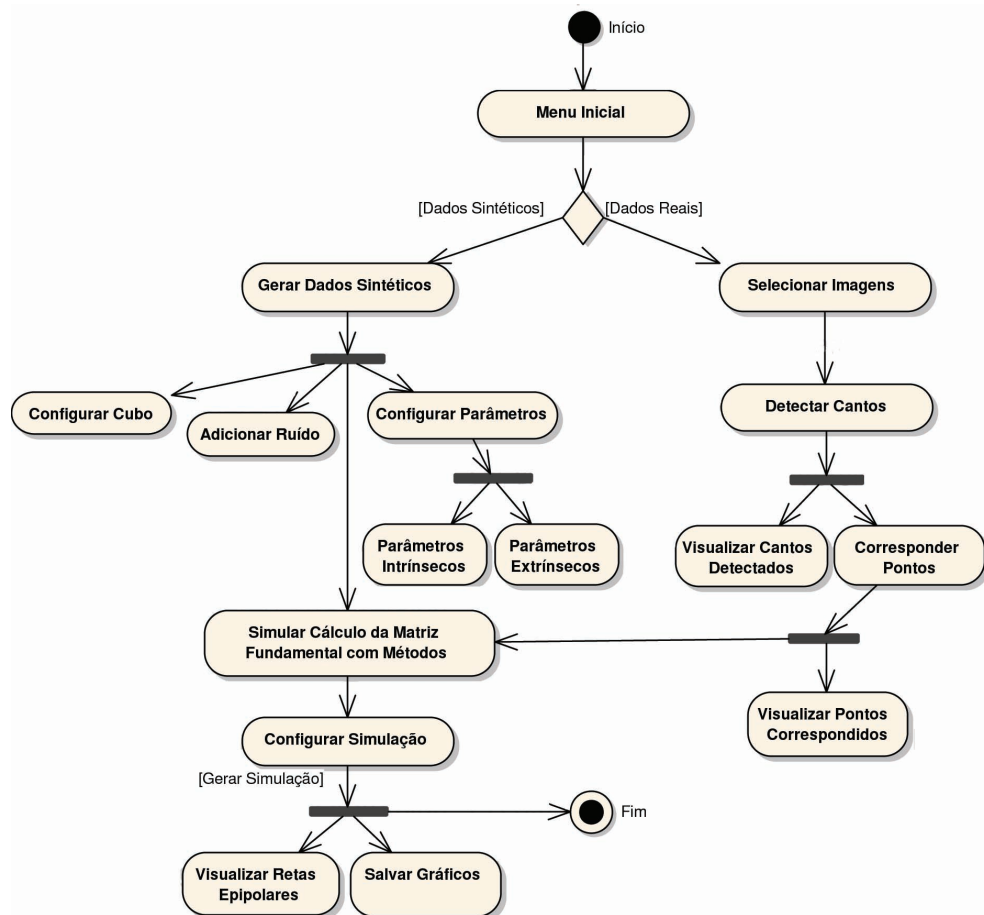


Figura A.2: Atividades do Ambiente Computacional Proposto

Na construção do ambiente interativo, seguiu-se o processo natural para estimar a matriz fundamental de um sistema binocular não-calibrado. Na figura A.2, é apresentado o diagrama de atividades do ambiente proposto. Nota-se que há uma sequência lógica para estimar a matriz fundamental e cada passo é necessário para o sucesso do passo subsequente.

Comparar o desempenho de dois ou mais métodos para cálculo da matriz fundamental não é uma tarefa óbvia. Isto é devido a dois fatores. Primeiro, a matriz fundamental pode ser decomposta em seus epipolos e na homografia epipolar. Contudo, a homografia epipolar pertence ao espaço \mathbf{P}^2 , que não possui nenhuma

métrica, tornando difícil definir uma medida de ajuste. Já os epipolos, apesar de poderem ser considerados parâmetros euclidianos, são muito susceptíveis a ruído e, por isso, muito difíceis de serem estimados com exatidão, sobretudo quando tendem ao infinito (LUONG; FAUGERAS, 1998). Além disso, o desempenho dos métodos é muito dependente de fatores, tais como, quantidade de correspondências utilizadas na estimação, distribuição de tais correspondências no plano de imagem e localização dos epipolos. Neste trabalho, é utilizada uma metodologia de testes semelhante à realizada por Hartley (1997). A mesma foi implementada no ambiente computacional proposto, ou seja:

a) Em cada experimento, são utilizadas N correspondências de pontos. Em todos os casos, considera-se que o conjunto de correspondências possua ruído com distribuição gaussiana de média zero e variância σ .

b) Dentre as j correspondências diferentes, S_j subconjuntos são selecionados aleatoriamente. Todos os subconjuntos possuem a mesma quantidade, X_j , de correspondências.

c) A cada iteração, cada método a ser avaliado utiliza todos os S_j subconjuntos escolhidos (um por vez) e estima as matrizes, \mathbf{F}_j , diferentes. Além disso, para cada método, é calculado o seguinte resíduo

$$\bar{r}_j^2 = \frac{1}{N} \sum_{i=1}^N d^2(\tilde{\mathbf{m}}_{2i}, \mathbf{F}_j \tilde{\mathbf{m}}_{1i}) + d^2(\tilde{\mathbf{m}}_{1i}, \mathbf{F}_j^T \tilde{\mathbf{m}}_{2i}).$$

d) Após os S_j subconjuntos serem aplicados, a mediana \bar{R}_l^2 é calculada para cada método, onde

$$\bar{R}_l^2 = \text{mediana}\{\bar{r}_1^2, \dots, \bar{r}_N^2\}.$$

e) Em seguida, o número de correspondências, X_j , é aumentado e outro subconjunto é escolhido aleatoriamente, isso ocorre até que S_j subconjuntos sejam escolhidos. No total, X_j assume valores diferentes (dentro o intervalo de X_1 a X_j). Dessa forma, para cada par de imagens, cada método é testado $\sum_{i=1}^j X_j$ vezes, com diferentes números de correspondências.

Com essa metodologia, comparando-se os resíduos \bar{R}_l^2 obtidos por métodos diferentes, tem-se uma boa medida de qual deles estimou a matriz fundamental que melhor ajustou-se ao conjunto total de correspondências, ou seja, o método com menor resíduo é o que possui o melhor resultado (ZHANG, 1998). Dentre todos os métodos para calcular a matriz fundamental, o método proposto por Levenberg-Marquardt (LEVENBERG, 1944; MARQUARDT, 1963) foi utilizado sempre que um

algoritmo de otimização era necessário, com a exceção ao método da Máxima Verossimilhança (Golden), no qual foi utilizado o método de Levenberg-Marquardt Dividido (FRANÇA et al., 2008).

Diversos testes foram realizados para avaliar, na prática, a exatidão dos métodos e compará-los. Na figura A.8, é mostrada a execução da simulação da estimação da matriz fundamental pelo ambiente proposto, com as saídas no ambiente Matlab, na qual são apresentadas as matrizes fundamentais e os passos de testes realizados, de acordo com as configurações determinadas para a simulação. É notório que a proposta do novo ambiente facilita e traz uma motivação maior para a execução do processo de estimação da matriz fundamental, assim como a visualização dos resultados de formas textuais e gráficas, possibilitando ao estudante, o entendimento intuitivo da prática desse importante tópico da visão computacional, estimulando o aprendizado crítico. Na próxima seção, são demonstrados os resultados experimentais, no qual são comparados os principais métodos para estimar a matriz fundamental, utilizando-se do ambiente proposto.

A.6 Um Estudo de Caso

Para estimar a matriz fundamental de um conjunto binocular, é demonstrado neste trabalho a simulação de um processo. Para tanto, é utilizado um par de imagens reais, capturada por câmeras do modelo *pinhole*¹. Contudo, também foi simulado dados sintéticos. Na sequência é explicado, detalhadamente, a utilização do ambiente interativo.

A.6.1 Geração de Dados Sintéticos

Essa funcionalidade permite a geração de dados sintéticos para a simulação do cálculo da matriz fundamental, para isso, é gerado um “cubo” sintético, no qual o usuário fornece algumas informações para sua criação, tal como, a largura da face do cubo, a quantidade de pontos de cada face e o ângulo de rotação relacionados aos eixos x, y e z. Entretanto, também, deve-se configurar os parâmetros intrínsecos e extrínsecos das câmeras simuladas. Para os parâmetros intrínsecos, é possível informar a coordenada do centro da projeção, representada por \mathbf{u}_0 e \mathbf{v}_0 e, também, os dados relativos ao fator de escala α (\mathbf{ax}) e β (\mathbf{ay}) para, respectivamente, os eixos x e y. Já os parâmetros extrínsecos, definem a rotação e

¹Consiste numa maneira de ver uma imagem real por meio de uma câmara escura. De um pequeno orifício onde a luz é captada para dentro da camera e sofrendo um movimento de inversão, a imagem é projetada para a parede oposta ao orifício ao contrário

translação da câmera 2 em relação a câmera 1, informados pelo usuário. A translação é definida pelos ângulos relacionados aos eixos x , y e z . Por outro lado, para a rotação, é gerada uma matriz quadrada de ordem 3, que representa as rotações no espaço 3D como um produto de 3 rotações sucessivas em torno dos eixos x , y e z , baseada nos ângulos de Euler (HARTLEY; ZISSERMAN, 2003) e fornecidos pelo usuário.

Além das informações relacionadas às câmeras, pode-se adicionar ruídos aos dados. Para isso, deve-se informar a variância (σ) de média zero aos pontos sintéticos \mathbf{m}_1 e \mathbf{m}_2 , associados, respectivamente, às câmeras 1 e 2. Enfim, a função estima a matriz fundamental com os dados sintéticos, com e sem ruído, por meio do método dos 8 pontos (HARTLEY, 1997) e, em seguida, é calculado o ajuste de ambos os dados gerados, de acordo com Zhang (1998). Um exemplo da geração de dados sintéticos é mostrado na figura A.3.

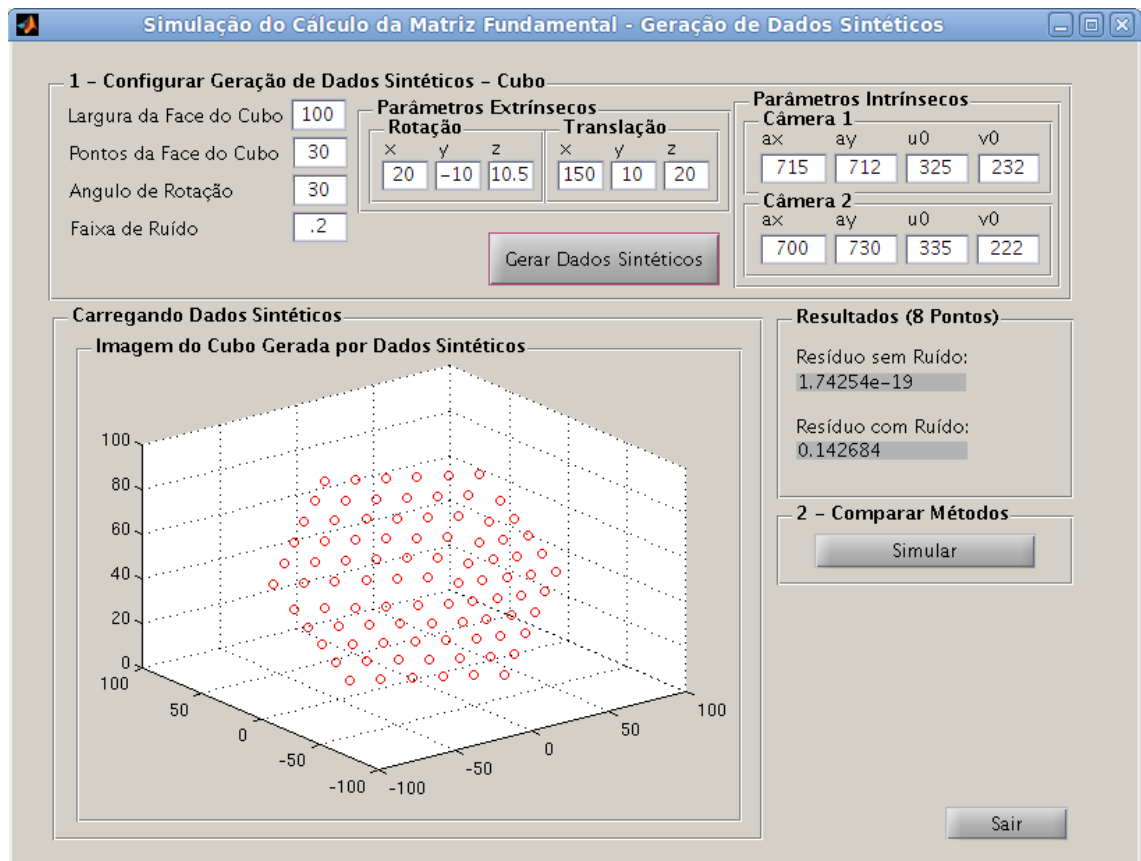


Figura A.3: Geração de Dados Sintéticos a Partir de um Cubo Projetado

De acordo com os testes realizados, mostrado na figura A.3, observa-se que o cubo foi gerado com 30 pontos por face, com uma rotação de 30 graus. A projeção da imagem leva em consideração apenas 3 faces, tem-se um total de 90 pontos identificados. Para tanto, foi aplicado um ruído, σ , de mediana 0,2. Nota-se que o cálculo preliminar da matriz fundamental, utilizando-se do método dos 8

pontos, obtêm-se um valor de ajuste mínimo para os dados sintéticos sem ruído e um valor muito maior para os dados com ruído. É notável que não foi necessário utilizar-se dos métodos para identificação de cantos e correspondência de pontos, uma vez que os pontos são gerados sinteticamente. Então, é possível simular a comparação dos métodos para o cálculo da matriz fundamental.

A.6.2 Selecionar Imagens

Inicialmente, o estudante deve selecionar as imagens do conjunto binocular. A tarefa é simples, basta acionar por meio dos botões “Câmera 1” e “Câmera 2” na tela principal do ambiente. As imagens devem ter uma boa resolução para a simulação e fatores como iluminação deve ser levado em consideração, afim de se obter bons resultados. Contudo, as imagens são mostradas em espaços visíveis à execução da tarefa. Na figura A.4, é apresentado o ambiente com as imagens PLANTA carregadas. Em seguida, o botão para detecção de cantos é habilitado.

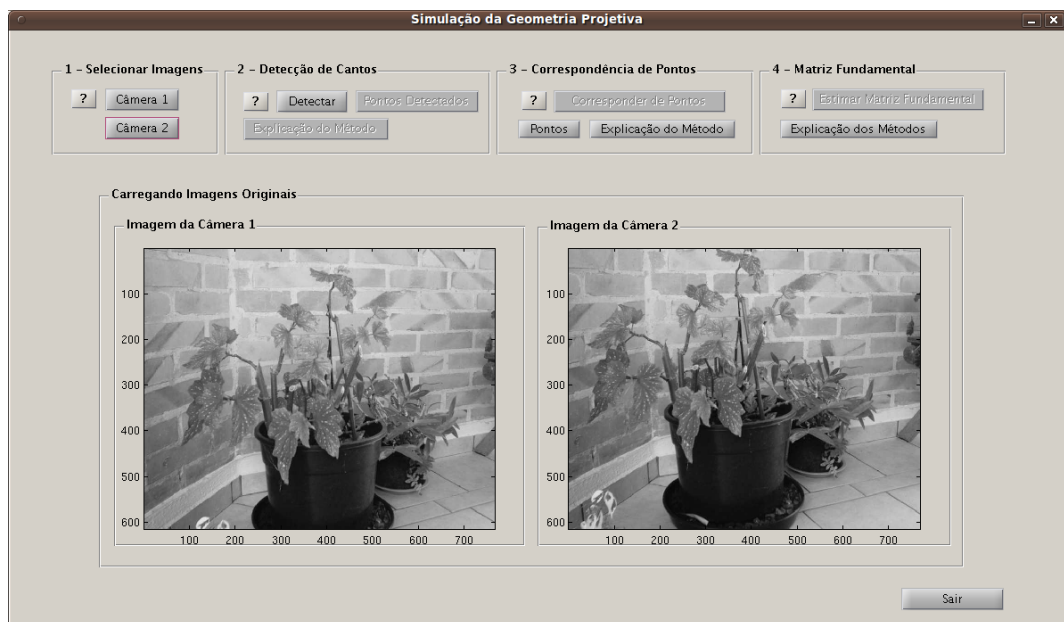


Figura A.4: Carregando as Imagens PLANTA no Ambiente Interativo

A.6.3 Detectar Cantos

A detecção de cantos é uma tarefa importante para a estimação da matriz fundamental, uma vez que é subsídio para a fase de correspondência de pontos. Para isso, basta acionar o botão “Detectar Cantos” na tela principal do ambiente. Quando executado, as imagens com os cantos identificados, por meio do método proposto por Harris e Stephens (1988), substituirão as imagens carregadas inicialmente. Então, são habilitados os botões para a visualização dos pontos

detectados e para efetuar a correspondência de pontos. Na figura A.5, observa-se que os cantos detectados são assinalado por um círculo.

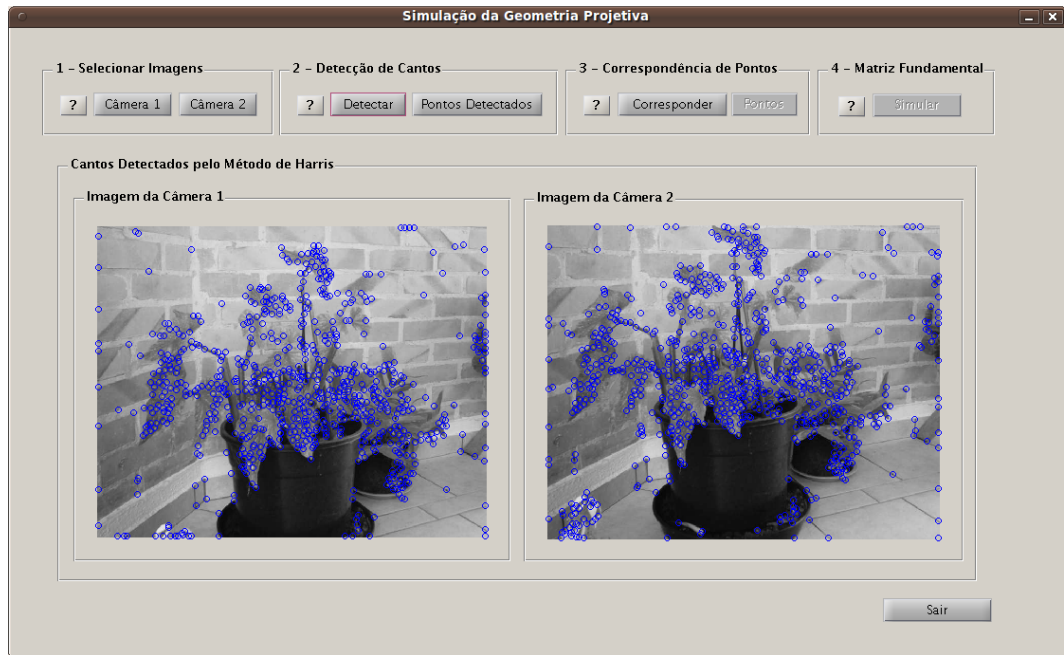


Figura A.5: Execução da Detecção de Cantos das Imagens PLANTA

A.6.4 Corresponder Pontos

O algoritmo de correspondência de pontos foi proposto por Zhang et al. (1995). Este processa um conjunto de pontos correspondidos, a mediana do ajuste e a estimativa inicial da matriz fundamental (HARTLEY, 1997). A execução do método para correspondência de pontos é mostrado na figura A.6. As imagens são assinaladas por “flechas” e indicam o deslocamento de um ponto em relação ao seu correspondente na outra imagem. Uma vez extraídos os pontos correspondidos, é possível então calcular a matriz fundamental. Por fim, é habilitado o botão para estimar a matriz fundamental, possibilitando ao estudante a simulação de vários métodos (lineares (HARTLEY, 1997; TRAJKOVIC; HEDLEY, 1997; LIU; MANNER, 2003), não-lineares (HARTLEY; ZISSERMAN, 2003; ZHANG; LOOP, 2001; BARTOLI; STURM, 2004) e robustos (FISCHLER; BOLLES, 1981; TORR; ZISSERMAN, 1998; ROUSSEEUW, 1987)).

A.6.5 Comparar Métodos para Estimar Matriz Fundamental

Esta funcionalidade do ambiente interativo proposto tem grande importância para o entendimento dos estudantes, pois permite escolher quais métodos serão utili-

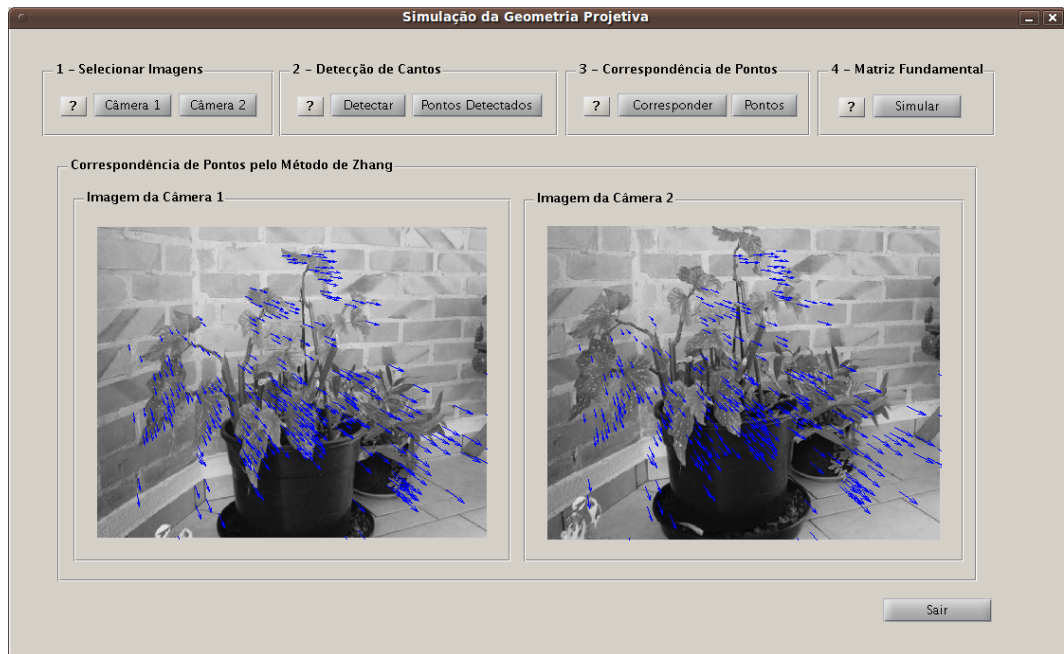


Figura A.6: Execução da Correspondência de Pontos

zados na estimação, bem como os parâmetros de configuração de cada um deles. Isto, além de permitir o entendimento dos mesmos, facilita a confrontação dos métodos em termos de exatidão e número de iterações, por meio de resultados gráficos e textuais. Na figura A.7, é apresentada uma comparação entre os métodos para a estimação da matriz fundamental. No entanto, é possível efetuar várias execuções com configurações variadas, pois, nesse ambiente, são disponibilizadas algumas opções para configurações individuais dos métodos não-lineares e robustos. Entretanto, também é possível configurar a geração da simulação e testes à serem realizados. Como resultado, são apresentadas as medianas dos ajustes de cada método calculado, os gráficos comparativos e as respectivas matrizes fundamentais calculadas no próprio console do MATLAB. Então, é possível também visualizar as retas epipolares e epipolos dos métodos calculados, assim como salvar as imagens geradas. Na figura A.7, é demonstrado o resultado da simulação para o cálculo da matriz fundamental. Já na figura A.8, são apresentados os resultados no modo texto. Para tanto, foram utilizados 5 ensaios para cada subconjunto, com pontos selecionados aleatoriamente, para todos os métodos avaliados. Contudo, foram gerados 16 subconjuntos diferentes, sendo o primeiro com 20 e o último com 50 correspondências. Os dois métodos não-lineares avaliados utilizaram-se de configurações específicas, que consistem em uma tolerância de 10^{-6} e um limite de 100 iterações. Já nos métodos robustos, o percentual de um subconjunto ser sorteado foi de 99%, o número de pontos do subconjunto é 8, o limiar do resíduo é de 4, que permite as comparações para

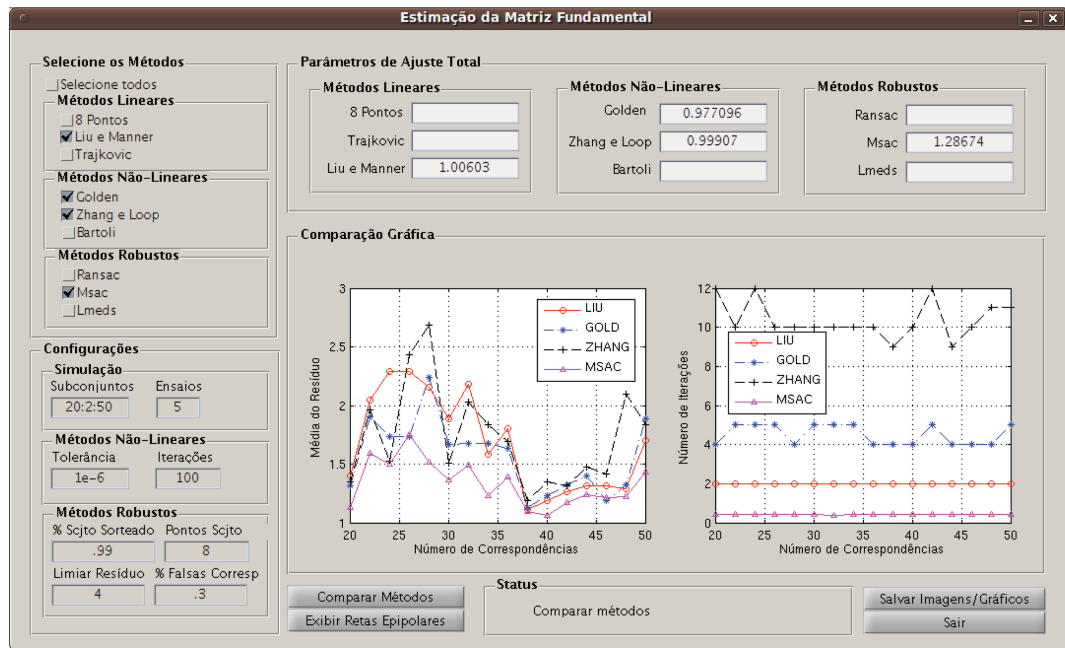


Figura A.7: O Novo Ambiente Estimando a Matriz Fundamental dos Métodos Lineares, Não-Lineares e Robustos

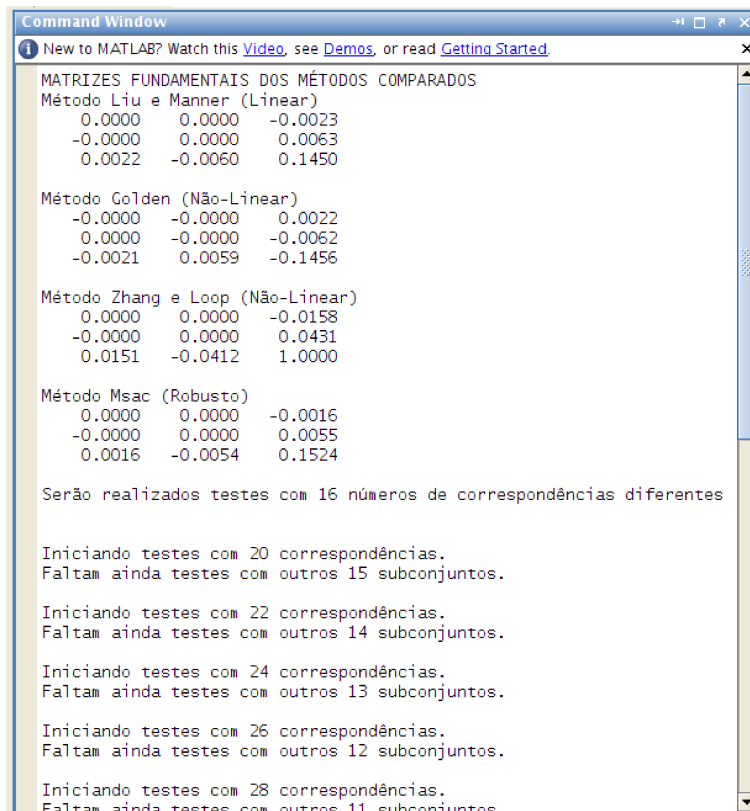


Figura A.8: Resultados Textuais Gerados pelo Novo Ambiente da Simulação da Estimação da Matriz Fundamental no Ambiente Matlab

detectar as falsas correspondências. O percentual de falsas correspondências é de 30%. Nota-se nos resultados obtidos, que o método não-linear Golden (Máxima Verossimilhança) (HARTLEY; ZISSERMAN, 2003) possui os melhores resultados no

critério de exatidão em relação ao método não-linear de Zhang (ZHANG; LOOP, 2001). Já o método robusto MSAC (TORR; ZISSERMAN, 1998), resultou, para todos os subconjuntos analisados, os melhores resultados em comparação à todos os métodos avaliados e, como utiliza-se de um método linear para o cálculo da matriz fundamental, também não necessita de iterações para chegar aos números finais. Enfim, o método linear de Liu (LIU; MANNER, 2003), obteve os piores resultados em comparação aos métodos simulados.

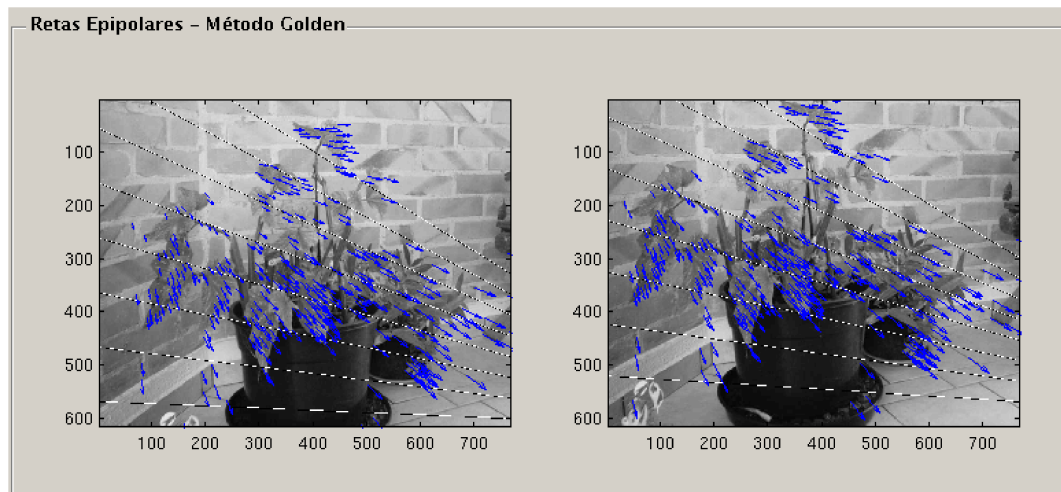


Figura A.9: Exibição das Retas Epipolares Calculadas a partir da Estimação da Matriz Fundamental pelo Método Golden

O ambiente computacional permite, sem mesmo sair da tela, uma nova simulação com combinações diferentes de métodos para estimar a matriz fundamental. Uma vez simulado, o estudante poderá visualizar as retas epipolares e epipolos gerados por cada método, na figura A.9 são demonstradas as figuras PLANTA com os respectivos desenhos das retas epipolares.

A.7 Conclusão

Apesar da disponibilidade de alguns *toolboxes* e ferramentas com propósitos similares, o ambiente computacional proposto nesse trabalho preenche uma lacuna no estudo da geometria epipolar de sistemas não-calibrados, uma vez que contribuem para uma análise detalhada dos resultados gerados da estimação da matriz fundamental de um conjunto binocular. No entanto, também possibilita a geração de testes com configurações personalizadas, permitindo uma otimização nas simulações para comparação entre os métodos. O ambiente computacional se mostrou bastante confiável em relação aos resultados apresentados, e seu maior objetivo foi alcançado devido a alta interatividade, facilitando aos alunos e pesquisadores

um estudo prático da geometria epipolar. De acordo com o estudo de caso, os algoritmos não-lineares, que calculam a matriz fundamental, demonstraram valores mais exatos nos experimentos, embora tenham um maior custo computacional. Contudo, tal estimação, ainda carece de uma técnica que obtenha resultados exatos a um custo computacional aceitável.

Referências

- ALIANE, N. A matlab/simulink-based interactive module for servo systems learning. *IEEE Transactions on Education*, v. 53, n. 2, p. 265–271, 2010.
- ALVES, E. G. *Desenvolvimento de Algoritmos de Calibração de Câmeras para Aplicações em Robótica*. Dissertação (Mestrado) — Universidade Estadual de Londrina, 2007.
- ARMANGUÉ, X.; SALVI, J. Overall view regarding fundamental matrix estimation. *Image and Vision Computing*, v. 21, p. 205–220, 2003.
- ASTROM, K.; HEYDEN, A.; KAHL, F.; BERTHILSSON, R.; SPARR, G. A computer vision toolbox. In: *Proceedings of Nordic MATLAB Conference*. [S.l.: s.n.], 1997.
- AYASUN, S.; NWANKPA, C. O. Induction motor tests using matlab/simulink and their integration into undergraduate electric machinery courses. *IEEE Transactions on Education*, v. 48, n. 1, 2005.
- AYASUN, S.; NWANKPA, C. O.; KWATNY, H. G. Voltage stability toolbox for power system education and research. *IEEE Transactions on Education*, v. 49, n. 4, p. 432–442, 2006.
- BARTOLI, A.; STURM, P. Nonlinear estimation of the fundamental matrix with minimal parameters. *IEEE Trans. on pattern analysis and machine intelligence*, v. 26, n. 3, p. 426–432, 2004.
- BARTOLI, A.; STURM, P.; HORAUD, R. Projective structure and motion from two views of a piecewise planar scene. In: *8th International Conference on Computer Vision*. Vancouver, Canada: [s.n.], 2001. v. 1, p. 593–598.
- BOUFAMA, B. S.; MOHR, R. A stable and accurate algorithm for computing epipolar. *International Journal of Pattern Recognition and Artificial Intelligence*, v. 12, n. 6, p. 817–840, 1998.
- CAVICCHI, T. J. Experimentation and analysis: Siglab/matlab data acquisition experiments for signals and systems. *IEEE Transactions on Education*, v. 48, n. 3, 2005.
- CHAU, K.-T. A software tool for learning the dynamic behavior of power electronics circuits. *IEEE Transactions on Education*, v. 39, n. 1, 1996.
- CHOJNACKI, W.; BROOKS, M.; HENGEL, A. van den; GAWLEY, D. Revisiting hartley’s normalized eight-point algorithm. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, v. 25, n. 9, p. 1172–1177, 2003.
- CORKE, P. I. A robotics toolbox for matlab. *IEEE Robotics and Automation Magazine*, 1996.

- CORKE, P. I. The machine vision toolbox. *IEEE Robotics and Automation Magazine*, 2005.
- CSURKA, G.; ZELLER, C.; ZHANG, Z.; FAUGERAS, O. Characterizing the uncertainty of the fundamental matrix. *Computer Vision and Image Understanding*, v. 68, n. 1, p. 18–36, 1997.
- FAUGERAS, O.; LUONG, Q. T. The geometry of multiple images: The laws that govern the formation of multiple images of a scene and some of their applications. *MIT Press, Cambridge, MA*, 2001.
- FISCHLER, M.; BOLLES, R. Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography. *Communications of ACM*, v. 24, n. 6, p. 381–385, 1981.
- FRANÇA, J. A. de. *Calibração de um Sistema de Visão Estéreo: de Correspondências de Pontos à Reconstrução Euclideana*. Tese (Doutorado) — Universidade Federal de Santa Catarina, Florianópolis-Brasil, 2005.
- FRANÇA, J. A. de; FRANÇA, M. B. d. M.; KOYAMA, M. H.; SILVA, T. P. da. Uma implementação do algoritmo levenberg-marquardt dividido para aplicações em visão computacional. *Semina: Ciências Exatas e Tecnológicas*, v. 26, n. 2, p. 195–202, 2008.
- GODOY, H. A. S. A imagem tridimensional e o audiovisual. *Associação Brasileira de Cinema*, online, 2005. Disponível em: <www.abcine.org.br>.
- HARRIS, C.; STEPHENS, M. A combined corner and edge detector. In: . Manchester: [s.n.], 1988. p. 147–151.
- HARTLEY, R. In defence of the eight point algorithm. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, online, v. 19, n. 6, p. 580–593, 1997. Disponível em: <<http://rsise.anu.edu.au/hartley/My-Papers.html>>.
- HARTLEY, R.; ZISSERMAN, A. *Multiple View Geometry in Computer Vision*. [S.l.]: Cambridge University Press, 2003.
- KANWAL, S.; ARIF, M.; MAJEED, T. Machine vision toolbox for three-dimensional structure estimation from images. In: *Multi Topic Conference*. [S.l.: s.n.], 2003. p. 16–20.
- KNUDSEN, M. H. Experimental modeling of dynamic systems: An educational approach. *IEEE Transactions on Education*, v. 49, n. 1, 2006.
- LEVENBERG, K. A method for the solution of certain non-linear. *Quarterly of Applied Math*, v. 2, p. 164–168, 1944.
- LIU, B.; MANNER, R. A linear iterative least-squares method for estimating the fundamental matrix. *IEEE International Symposium on Signal Processing and its Applications*, online, p. 17–20, 2003. Disponível em: <<http://citeseer.ist.psu.edu/559963.html>>.
- LOOP, C.; ZHANG, Z. Computing rectifying homographies for stereo vision. *Conference on Computer Vision and Pattern Recognition*, 1999.

- LUONG, Q. T.; DERICHE, R.; FAUGERAS, O.; PAPADOPOULOU, T. On determining the fundamental matrix: Analysis of different methods and experimental results. [S.l.], online, v. 27, 1993. Disponível em: <<http://www.inria.fr/rrrt/rr-1894.html>>.
- LUONG, Q.-T.; FAUGERAS, O. The fundamental matrix: theory, algorithms, and stability analysis. *The International Journal of Computer Vision*, online, v. 17, n. 1, p. 43–76, 1996. Disponível em: <<http://citeseer.ist.psu.edu/luong95fundamental.html>>.
- LUONG, Q. T.; FAUGERAS, O. D. On the determination of epipoles using cross-ratios. *Computer Vision and Image Understanding*, online, v. 71, n. 1, p. 1–18, 1998. Disponível em: <<http://citeseer.ist.psu.edu/>>.
- MAGISTRIS, M. d. A matlab-based virtual laboratory for teaching introductory quasi-stationary electromagnetics. *IEEE Transactions on Education*, v. 48, n. 1, 2005.
- MARIOTTINI, G. L.; PRATTICCHIZZO, D. Egt for multiple view geometry and visual servoing. *IEEE Robotics and Automation Magazine*, v. 12, n. 4, p. 26–39, 2005. ISSN 1070-9932.
- MARQUARDT, D. W. An algorithm for least-squares estimation of nonlinear parameters. *J. Soc. Indust. Appl. Math*, v. 11, n. 2, p. 431–441, 1963.
- MATHWORKS, I. *Matlab - The Language of Technical Computing*. [S.l.], 2010. Disponível em: <www.mathworks.com/products/matlab>.
- POTA, H. R. A prototype flexible robot arm: An interdisciplinary undergraduate project. *IEEE Transactions on Education*, v. 35, n. 1, 1992.
- PRESSMAN, R. S. *Engenharia de Software*. 6. ed. [S.l.]: Mcgraw Hill - ARTMED, 2006.
- ROUSSEEUW, P. J. *Robust Regression and Outlier Detection*. [S.l.]: Wiley, 1987.
- SAGNARD, F. Educational graphical interfaces to learn about radiation and propagation of electromagnetic waves. *IEEE Transactions on Education*, v. 47, n. 3, 2004.
- SAMPER, D.; SANTOLARIA, J.; MAJARENA, A. C.; AGUILAR, J. J. Comprehensive simulation software for teaching camera calibration by a constructivist methodology. *Measurement*, v. 43, n. 5, p. 618 – 630, 2010. ISSN 0263-2241. IMEKO XIX World Congress Part 1 - Advances in Fundamental and Applied Metrology.
- SAMPSON, P. D. Fitting conics sections to very scattered data: An iterative refinement of the bookstein algorithm. *Computer Graphics and Image Processing*, v. 18, n. 1, p. 97–108, 1982.
- SCARAMUZZA, D.; SIEGWART, R. A practical toolbox for calibrating omnidirectional cameras. *Vision Systems: Applications*, Vienna, Austria, p. 608, 2007.

- TEIXEIRA, M. C. M.; ASSUNCAO, E.; COVACIC, M. R. Proportional controllers: Direct method for stability analysis and matlab implementation. *IEEE Transactions on Education*, v. 50, n. 1, 2007.
- TORR, P.; MURRAY, D. The development and comparison of robust methods for estimating the fundamental matrix. *International Journal of Computer Vision*, online, v. 24, n. 3, p. 271–300, 1997. Disponível em: <<http://citeseer.ist.psu.edu/torr97development.html>>.
- TORR P.; FITZGIBBON, A. Invariant fitting of two view geometry or in defiance of the eight point algorithm. *British Machine Vision Conference*, online, 2003. Disponível em: <<http://wwwcms.brookes.ac.uk/philiptorr/papers.htm>>.
- TORR, P. H. S.; ZISSERMAN, A. Robust computation and parametrization of multiple view relations. *Narosa Publishing House*, online, ICCV6, p. 727–732, 1998. Disponível em: <<http://wwwcms.brookes.ac.uk/philiptorr/>>.
- TORR, P. H. S.; ZISSERMAN, A. Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, online, v. 78, n. 1, p. 138–156, 2000. Disponível em: <<http://wwwcms.brookes.ac.uk/philiptorr/>>.
- TRAJKOVIC, M.; HEDLEY, M. An efficient method for the estimation of the fundamental matrix. *Proc. Digital Image and Vision Computing - Techniques and Application*, New Zealand, p. 925–935, 1997.
- URAN, S.; JEZERNIK, K. Virtual laboratory for creative control design experiments. *IEEE Transactions on Education*, v. 51, n. 1, 2008.
- VILA-ROSADO, D. N.; DOMINGUEZ-LOPEZ, J. A. A MATLAB toolbox for robotic manipulators. *Mexican International Conference on Computer Science*, IEEE Computer Society, Los Alamitos, CA, USA, p. 256–265, 2005. ISSN 1550-4069.
- WU, H. H. P.; YU, Y. H. Projective rectification with reduced geometric distortion for stereo vision and stereoscopic video. *Journal of Intelligent and Robotic Systems*, 2005.
- ZHANG, Z. Determining the epipolar geometry and its uncertainty: A review. *Internat. J. Computer Vision*, v. 27, p. 161–195, 1998.
- ZHANG, Z.; DERICHE, R.; FAUGERAS, O.; LUONG, Q. tuan. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, v. 78, p. 87–119, 1995.
- ZHANG, Z.; LOOP, C. Estimating the fundamental matrix by transforming image points in projective space. *Computer Vision and Image Understanding*, v. 82, n. 2, p. 174–180, 2001.