

Universidade Federal do Rio de Janeiro

Escola Politécnica

Departamento de Eletrônica e de Computação

**Interface para Sistema de Visão e Dimensionamento 3D com
Câmeras Fotográficas Digitais de Alta Resolução**

Autor:

Daniel de Carvalho Cayres Pinto

Orientador:

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

Examinador:

Prof. Julio Cesar Boscher Torres, D. Sc.

Examinador:

Prof^a. Mariane Rembold Petraglia, Ph. D.

DEL

Fevereiro de 2011

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

Escola Politécnica – Departamento de Eletrônica e de Computação

Centro de Tecnologia, bloco H, sala H-217, Cidade Universitária

Rio de Janeiro – RJ CEP 21949-900

Este exemplar é de propriedade da Universidade Federal do Rio de Janeiro, que poderá incluí-lo em base de dados, armazená-lo em computador, microfilmá-lo ou adotar qualquer forma de arquivamento.

É permitida a menção, reprodução parcial ou integral e a transmissão entre bibliotecas deste trabalho, sem modificação de seu texto, em qualquer meio que esteja ou venha a ser fixado, para pesquisa acadêmica, comentários e citações, desde que sem finalidade comercial e que seja feita a referência bibliográfica completa.

Os conceitos expressos neste trabalho são de responsabilidade do(s) autor(es) e do(s) orientador(es).

DEDICATÓRIA

À minha família.

AGRADECIMENTO

Agradeço aos meus pais, aos meus irmãos e à Ana Cecilia por todo apoio. Aos meus amigos, como Diego Wanderley, Carlos Caldas, Trevor Dobbin e Ricardo Flach que me ajudaram durante todo o curso.

Agradeço também a todos os meus professores do Departamento de Eletrônica e Computação e colegas do curso de Engenharia Eletrônica e de Computação pelos ensinamentos e colaborações neste trabalho e na minha formação como engenheiro, em especial ao professor José Gabriel Rodriguez Carneiro Gomes, pela orientação neste trabalho e à professora Mariane Rembold Petraglia pela cooperação nas pesquisas e por viabilizar minha bolsa de iniciação científica no Laboratório de Processamento Analógico e Digital de Sinais.

RESUMO

Este trabalho apresenta uma interface para um sistema de visão 3D com câmeras fotográficas digitais de alta resolução, para realização da calibração de câmeras e dimensionamento tridimensional de objetos.

A visão tridimensional é baseada nas diferenças entre as imagens projetadas nas retinas dos dois olhos. Para simular esse efeito (chamado de estereoscopia) em monitores, podem ser utilizadas algumas técnicas, que possibilitam a exibição de imagens diferentes para cada olho. Neste trabalho são utilizadas duas câmeras de alta resolução para a obtenção das imagens de diferentes perspectivas e um monitor 3D por polarização da luz.

A precisão do dimensionamento de objetos utilizando pares de imagens depende de dois fatores: a eficiência da calibração e a qualidade das fotografias. A calibração, por sua vez, também apresenta melhores resultados para imagens de melhor resolução.

O trabalho busca desenvolver uma interface de visão 3D que permita que se tire o maior proveito da qualidade das câmeras utilizadas no projeto, tanto na visualização da imagem estereoscópica, quanto na calibração e dimensionamento de objetos nas fotografias. Também são estudados métodos lineares de calibração e dimensionamento, cujos resultados são apresentados ao final do trabalho.

Palavras-Chave: estereoscopia, visão 3D, visão computacional, dimensionamento 3D.

ABSTRACT

This work presents a 3D vision system interface with high resolution digital photographic cameras, for camera calibration and tridimensional sizing of objects.

Tridimensional vision is based on the differences between the images projected in retinas of both eyes. To simulate this effect (called stereoscopy) in monitors, we can apply some techniques which make possible the exhibition of different images to each eye. In this study, we use two high resolution cameras for obtaining the images in different perspectives and a 3D monitor with polarized light.

The object stereo sizing precision depends upon two factors: calibration efficiency and photo quality. The calibration, in turn, also presents better results for images with better resolution.

This study aims at developing a 3D vision interface which allows taking advantage of the quality of cameras used in the project, with respect to both the stereoscopic image viewing and the calibration and sizing of objects in the photos. We also study linear methods for calibration and sizing, whose results are presented at the end of this text.

Key-words: stereoscopy, 3D vision, computer vision, 3D sizing.

Sumário

1	Introdução	1
	1.1 - Tema	1
	1.2 - Delimitação	1
	1.3 - Justificativa	2
	1.4 - Objetivos	2
	1.5 - Metodologia	2
	1.6 - Descrição	3
2	Interface de Visão 3D e de Dimensionamento	5
	2.1 - Descrição Geral do Sistema	5
	2.2 - Visão Estereoscópica	6
	2.3 - Monitor 3D	8
	2.3.1 - Tipos de Monitor	8
	2.3.2 - Paralaxe	9
	2.4 - Aplicativo de Visualização	11
	2.5 - Aplicativo de Calibração e Dimensionamento	16
3	Modelagem Matemática	19
	3.1 - Conceitos de Geometria Projetiva	19
	3.2 - Modelo de Câmera <i>Pinhole</i>	20
	3.3 - Matriz de Projeção	20
	3.4 - Parâmetros Intrínsecos e Extrínsecos	22

	3.4.1 - Parâmetros intrínsecos	23
	3.4.2 - Parâmetros Extrínsecos	24
4	Calibração e Dimensionamento	27
	4.1 - Calibração	27
	4.1.1 - Restrições	27
	4.1.2 - Método Linear de Calibração	28
	4.2 - Dimensionamento	31
	4.2.1 - Método dos Multiplicadores de Lagrange	31
	4.2.2 - Método dos Mínimos Quadrados Ordinários	34
5	Resultados	36
	5.1 - Procedimentos de Calibração e Dimensionamento	36
	5.2 - Resultados	43
6	Conclusões	47
	Bibliografia	49

Lista de Figuras

2.1 – Estereoscopia	6
2.2 – Pontos homólogos e oclusão	7
2.3 – Visão estereoscópica	7
2.4 – Monitor 3D entrelaçado exibindo linhas vermelhas e verdes alternadas e o funcionamento dos óculos 3D	9
2.5 – Tipos de paralaxe	10
2.6 – Paralaxe divergente	11
2.7 – Interface exibindo uma imagem estereoscópica	11
2.8 – Gerenciamento de eventos do usuário	12
2.9 – Exibição das imagens para o usuário	12
2.10 – Troca de mensagens com o par de câmeras	13
2.11 – Gerenciador de eventos das câmeras	14
2.12 – <i>Grid</i> de calibração	16
2.13 – Aplicativo de calibração	17
2.14 – Aplicativo de Dimensionamento	18
3.1 – Modelo Pinhole	20
3.2 – Modelo geométrico da câmera	21
3.3 – Novo sistema de coordenadas do plano da retina	23
3.4 – Novo sistema de coordenadas do espaço	25
5.1 – Par de imagens para calibração com o <i>grid</i> mais próximo das câmeras .	37
5.2 – Par de imagens para calibração com o <i>grid</i> mais afastado das câmeras .	38
5.3 – Pontos de referência para calibração	39
5.4 – Par de imagens para dimensionamento com o <i>grid</i> mais próximo das câmeras	40
5.5 – Par de imagens para dimensionamento com o <i>grid</i> mais afastado das câmeras	41

Lista de Tabelas

2.1 – Funções da EDSK	15
5.1 – Resultados do dimensionamento com imagens de alta resolução	43
5.2 – Erros relativos do dimensionamento com imagens de alta resolução .	44
5.3 – Resultados dos dimensionamentos com imagens de baixa resolução ..	45
5.4 – Erros dos dimensionamentos com imagens de baixa resolução	46

Capítulo 1

Introdução

Este trabalho tem como tema a Visão Computacional 3D, utilizando um par de câmeras digitais de alta resolução e qualidade e um monitor 3D de quadros entrelaçados.

1.1 – Tema

O projeto é dividido em duas partes, a primeira parte é a programação da interface das câmeras com o computador e dos aplicativos de calibração e dimensionamento, visando aproveitar da melhor maneira possível a alta qualidade das imagens. A segunda parte é o estudo de métodos lineares de calibração das câmeras e de estimação de coordenadas 3D de pontos presentes nas imagens, permitindo o dimensionamento de segmentos no espaço.

A visão computacional em três dimensões é objeto de interesse de vários ramos da Ciência, tendo aplicações, por exemplo, na Biologia, Geografia, Física, Arquitetura, Medicina e em diversas áreas da Engenharia, em especial na Engenharia Eletrônica. O dimensionamento automático de objetos no espaço tem aplicações em sistemas de automação, inspeção industrial, robótica e outros. Utilizando câmeras de alta resolução e com lentes de boa qualidade, é possível obter resultados de calibração e dimensionamento excelentes.

1.2 – Delimitação

A interface gráfica para captura de pares de imagens desenvolvida neste trabalho é programada para utilizar câmeras Canon T1i em sistema operacional Windows.

As rotinas de calibração e dimensionamento utilizam métodos lineares de aproximação. Os procedimentos de obtenção das coordenadas dos pontos de referência (para a calibração) e pontos das extremidades de um segmento (para o

dimensionamento) são realizados manualmente pelo usuário do programa. O trabalho não trata de calibração e dimensionamento automáticos.

1.3 – Justificativa

Nos últimos anos, houve grande avanço tecnológico nas áreas relacionadas à visão estéreo, tanto na indústria do entretenimento, quanto na engenharia. Na visão computacional 3D, os problemas da calibração das câmeras e dimensionamento de objetos no espaço são de extrema importância. Um fator muito importante para a obtenção de resultados precisos é a definição das imagens utilizadas. O desenvolvimento das câmeras digitais de alta resolução e a evolução dos monitores 3D trazem mais qualidade para a imagem, permitindo maior exatidão na calibração e dimensionamento.

1.4 – Objetivos

O objetivo deste trabalho é apresentar uma interface para um sistema de visão 3D com câmeras digitais de alta resolução, que permitem a visualização da cena em três dimensões com alta definição, a captura simultânea de fotografias, calibração manual com boa precisão e dimensionamento de objetos no espaço.

Pretende-se também avaliar o desempenho de métodos lineares para a calibração e dimensionamento, comparando dois métodos diferentes de dimensionamento e testando a sensibilidade do sistema a ruídos na imagem.

1.5 – Metodologia

Para o desenvolvimento da interface gráfica dos softwares foram utilizadas as Classes Fundamentais da Microsoft (MFC) para a programação de todos os controles necessários, como botões, caixas de texto e caixas de imagem. Para o desenvolvimento da comunicação das câmeras com o computador, foram utilizadas funções da ferramenta de desenvolvimento fornecida pela Canon para as câmeras digitais, a EDSDK (EOS Digital Software Development Kit, sendo EOS sigla para Electro-Optical System).

A modelagem matemática das câmeras foi feita utilizando geometria projetiva, inicialmente considerando um modelo simplificado da câmera e em seguida desenvolvendo o modelo até chegar à forma final. O algoritmo de calibração foi desenvolvido utilizando o modelo matemático obtido com geometria projetiva e o método dos multiplicadores de Lagrange, que também foi utilizado para um dos algoritmos de dimensionamento. A segunda forma de resolver o problema do dimensionamento foi através do método dos mínimos quadrados ordinários.

1.6 – Descrição

No Capítulo 2 será detalhada a interface do sistema. Inicialmente será feita uma breve explicação do funcionamento da visão estereoscópica humana, serão explicados alguns métodos de exibição de imagens 3D a partir de duas imagens de duas dimensões, depois será feita a descrição do monitor utilizado no projeto. Em seguida, serão exibidos os dois aplicativos utilizados no projeto, o de visualização da imagem estereoscópica para o monitor utilizado com as câmeras de alta resolução e o de calibração e dimensionamento, onde são aplicados os métodos explicados nos capítulos seguintes.

O Capítulo 3 introduz alguns conceitos de Geometria Projetiva necessários para a modelagem das câmeras, que é o segundo assunto abordado nesse capítulo. A princípio é exposto um modelo simplificado que em seguida é desenvolvido até chegar à forma geral de representação. É introduzido o conceito de parâmetros intrínsecos e extrínsecos de uma câmera e seu papel na matriz de projeção.

Em seguida, no Capítulo 4, será apresentado um método linear para a calibração de câmeras. Nesse capítulo também são estudados dois métodos lineares para a estimação das coordenadas 3D de um ponto projetado nas duas imagens do par estereoscópico a partir das coordenadas 2D desse ponto nas duas imagens e, utilizando as coordenadas 2D de dois pontos, medir a distância tridimensional entre eles.

No Capítulo 5 serão apresentados os resultados obtidos utilizando os métodos de calibração e dimensionamento apresentados no capítulo anterior e os aplicativos descritos no Capítulo 2. São comparados os valores das dimensões de

segmentos iguais medidas pelos dois métodos de dimensionamento e serão analisadas as diferenças obtidas nos resultados quando é adicionado ruído ao valor das coordenadas 2D dos pontos de calibração e dos pontos das extremidades dos segmentos medidos. Também são comparadas as diferenças quando a calibração e dimensionamento são realizados sobre as imagens com resolução reduzida.

Capítulo 2

Interface de Visão 3D e de Dimensionamento

Neste capítulo será detalhado o desenvolvimento do aplicativo de interface das câmeras com o computador. Inicialmente serão apresentados alguns conceitos necessários para se compreender a visão tridimensional humana e, em seguida, será apresentado o funcionamento do *software*.

Também neste capítulo é explicado o funcionamento dos aplicativos de calibração e dimensionamento, mais especificamente da interface com o usuário. O tratamento matemático utilizado nessas duas etapas é descrito nos capítulos seguintes.

2.1 – Descrição Geral do Sistema

O sistema é formado por duas câmeras Canon T1i que se conectam ao computador por interface USB (Universal Serial Bus), permitindo o controle das câmeras via software. A imagem é exibida em um monitor 3D de alta definição, que utiliza o método de quadros entrelaçados e polarização da luz.

O projeto conta com dois aplicativos: um para a visualização em tempo real da imagem estereoscópica e captura simultânea de fotografias de alta resolução e outro para calibração do par de câmeras e dimensionamento tridimensional de segmentos presentes nas imagens.

A interface gráfica foi construída com as funções da MFC e, portanto, os programas funcionam exclusivamente na plataforma Windows. A interface do software de visualização consiste de três botões, um para capturar as imagens com as duas câmeras e construir a imagem estereoscópica e outros dois para ativar ou desativar o visor eletrônico. O programa de calibração e dimensionamento é formado por duas telas, uma para cada função, que são descritas nas próximas seções.

Para realizar a comunicação das câmeras com o computador foi utilizada interface USB e a ferramenta de desenvolvimento EDSDK fornecida pela Canon. Todas as ações são realizadas através de trocas de mensagens entre as câmeras e o computador.

2.2 – Visão Estereoscópica

Várias espécies de animais possuem o recurso da visão em três dimensões. Essa característica, chamada de visão estereoscópica (Figura 2.1), é resultado da capacidade do cérebro de dar noções de profundidade, distância e volume a partir das diferenças entre as imagens captadas pelos dois olhos.

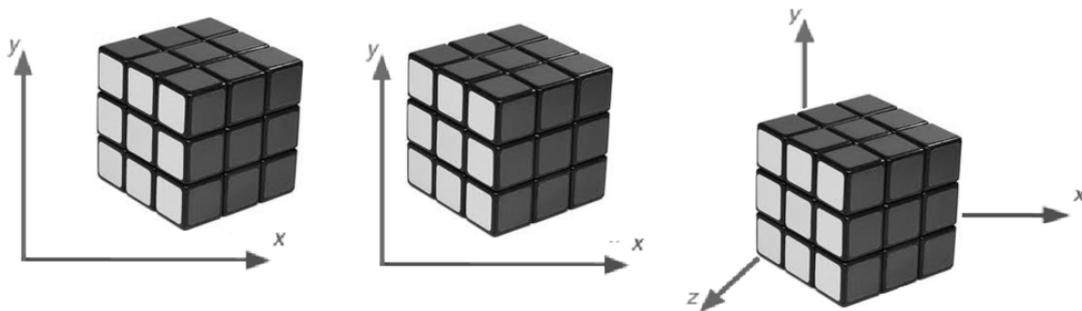


Figura 2.1 – Estereoscopia

Os olhos humanos são separados por uma distância horizontal de aproximadamente seis centímetros, por isso a imagem da cena observada projetada na retina esquerda é diferente da imagem projetada na retina direita. Alguns pontos podem ser observados através de um olho, mas não são visíveis para o outro por estarem oclusos, ou por estarem fora de seu ângulo de visão. Outros pontos são visualizados através de ambos os olhos, nesse caso as imagens desses pontos projetadas nas duas retinas são chamadas de pontos homólogos.

A Figura 2.2 mostra um exemplo em que estão presentes na cena um cubo e uma esfera. A esfera aparece na imagem captada pelo olho esquerdo, mas não pela captada pelo olho direito, diz-se que a esfera está oclusa para o olho direito, portanto não é possível obter pontos homólogos de sua imagem. Já o cubo é visível

nas duas perspectivas e as imagens de cada vértice do cubo que estão presentes em ambas as projeções são exemplos de pontos homólogos.

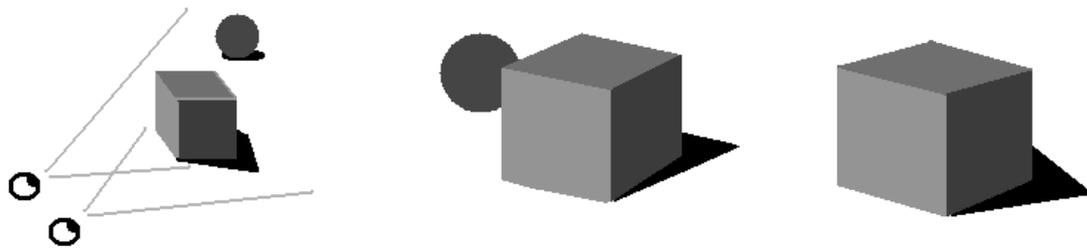


Figura 2.2 – Pontos homólogos e oclusão

A Figura 2.3 exibe um modelo geométrico da visão estereoscópica, onde D é a distância do objeto ao observador, f é a distância focal dos olhos, d é a distância entre os olhos, O é a origem do sistema de coordenadas da retina e x_1 e x_2 são as coordenadas horizontais da imagem do objeto nas duas retinas.

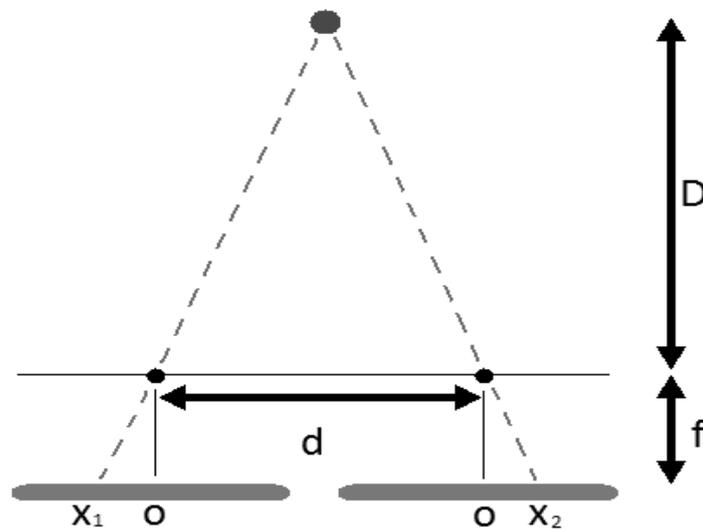


Figura 2.3 – Visão estereoscópica

Analisando a Figura 2.3, por semelhança de triângulos tem-se

$$\frac{d}{D} = \frac{x_2 - x_1}{f} \quad (2.1)$$

e

$$D = \frac{df}{X_2 - X_1}. \quad (2.2)$$

O termo $X_2 - X_1$, a distância horizontal entre as imagens de um ponto projetado nas retinas, é chamado de disparidade. Analisando a Equação (2.2) vemos que quanto maior a distância do objeto ao observador menor a disparidade. Um objeto no infinito resultaria em disparidade zero.

2.3 – Monitor 3D

Todo equipamento de exibição de imagens 3D tem o mesmo princípio básico, exibir uma imagem diferente para cada olho, de modo que a disparidade nas imagens determine a profundidade dos objetos no visor. Existem diferentes maneiras de produzir esse efeito, como será explicado a seguir.

2.3.1 – Tipos de Monitor

Uma técnica simples que é utilizada para exibir imagens tridimensionais em monitores, televisores e projetores comuns é o anáglifo, onde em cada uma das imagens é utilizado um filtro de uma cor (normalmente vermelho e ciano) e em seguida é produzida a imagem estereoscópica pela soma das outras duas. Utilizando óculos com lentes com aquelas cores, cada olho vê com mais intensidade apenas uma das imagens e o cérebro consegue formar a imagem 3D.

Com o avanço da tecnologia, foi possível criar uma técnica mais eficiente, que provoca menor distorção cromática e desconforto a quem assiste, fazendo uso de luz polarizada. As imagens esquerda e direita são exibidas com luz de polarizações diferentes, as lentes dos óculos deixam passar cada uma a luz polarizada em uma direção e, assim, cada olho recebe apenas uma das imagens. Os monitores com essa tecnologia geralmente exibem os quadros em linhas alternadas, ou seja, linhas pares para a imagem de um olho e linhas ímpares para a imagem do outro. A Figura 2.4 exibe o monitor 3D de quadros entrelaçados, que é

o utilizado no laboratório, exibindo alternadamente linhas verdes e vermelhas. A lente esquerda dos óculos mostra somente as linhas vermelhas e a lente direita mostra somente as linhas verdes.

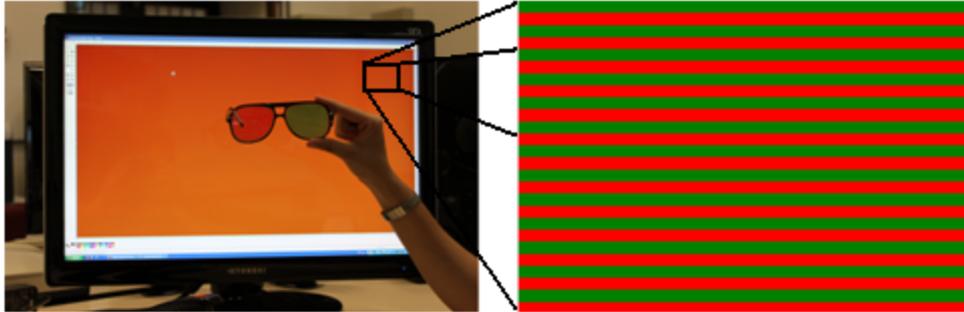


Figura 2.4 – Monitor 3D entrelaçado exibindo linhas vermelhas e verdes alternadas e o funcionamento dos óculos 3D

O inconveniente da utilização de luz polarizada é que para visualizar as imagens tridimensionais de maneira correta, o usuário precisa estar posicionado de forma específica na frente do monitor. O ângulo de visão do monitor polarizado é reduzido e, se não respeitado, a filtragem pelas lentes dos óculos não é realizada da maneira correta. Outro problema é que em imagens onde há muito contraste entre as cores, a imagem de um olho pode aparecer com intensidade reduzida no outro olho.

A solução para os problemas da tecnologia de luz polarizada é a utilização de monitores com óculos ativos de lentes LCD. Esses óculos impedem completamente a passagem de luz de uma das lentes alternadamente. Quando o quadro esquerdo é exibido, o olho direito é tampado, quando o quadro direito é exibido, a lente direita volta a permitir a passagem de luz e a lente esquerda passa a bloquear. Contudo, nesse caso, é necessário um sistema de sincronismo entre o monitor e os óculos.

2.3.2 – Paralaxe

A paralaxe é um conceito similar à disparidade, mas enquanto a disparidade é a diferença na horizontal entre as projeções de um mesmo ponto nas retinas, a

paralaxe é a diferença horizontal entre pontos homólogos projetados em uma tela [2].

Quando se trabalha com monitores 3D, deve-se então tomar os cuidados necessários com a paralaxe, para que a disparidade produzida por ela seja a adequada para o cérebro interpretar a imagem tridimensional de forma correta. Existem três tipos de paralaxe [2], ilustrados pela Figura 2.5:

- Paralaxe zero – Ocorre quando os pontos homólogos estão sobrepostos na tela, tem-se a sensação de que o ponto está no mesmo plano da projeção.
- Paralaxe positiva – Ocorre quando o ponto do quadro esquerdo está localizado à esquerda do ponto do quadro direito na tela. Nesse caso a posição simulada do ponto é atrás do plano da projeção.
- Paralaxe negativa – Ao contrário da positiva, acontece quando o ponto do quadro esquerdo está à direita do ponto do quadro direito. Produz a sensação de que o ponto está entre o observador e a tela.

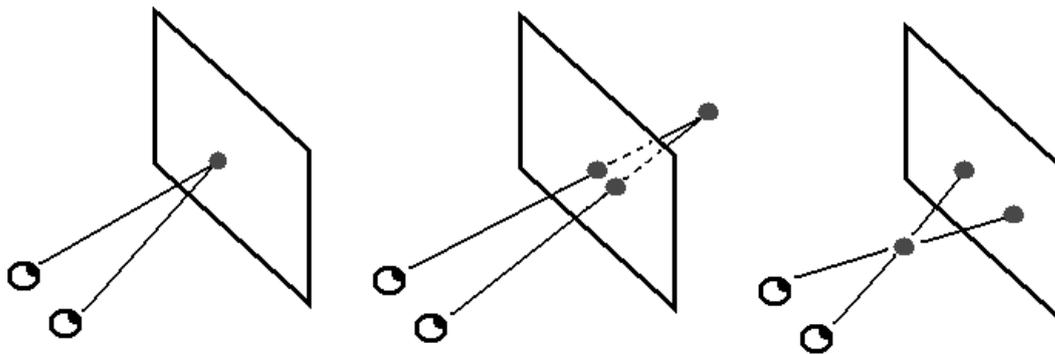


Figura 2.5 – Tipos de paralaxe

Para posicionar um ponto no infinito, a paralaxe deve ser igual à distância entre os olhos. Caso seja maior que essa distância, a disparidade produzida por essa paralaxe fica negativa, o que é claramente impossível analisando o modelo geométrico (Figura 2.6), portanto a imagem não converge.

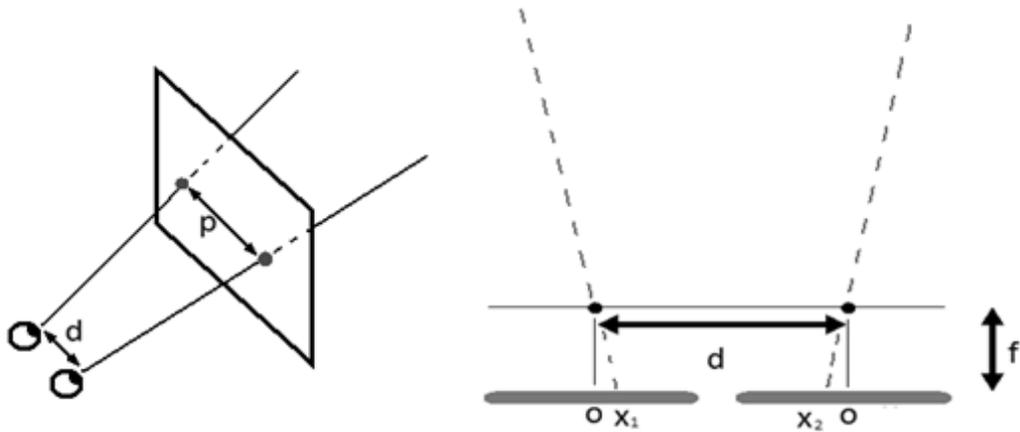


Figura 2.6 – Paralaxe divergente

2.4 – Aplicativo de Visualização

O aplicativo de visualização da imagem estereoscópica, que pode ser visto na Figura 2.7, é composto por três quadros nos quais são exibidas em tamanho reduzido as imagens de cada câmera e, com maior resolução, a imagem 3D e três botões, um para tirar uma fotografia com cada câmera e outros dois para ativar o visor eletrônico e desativá-lo.

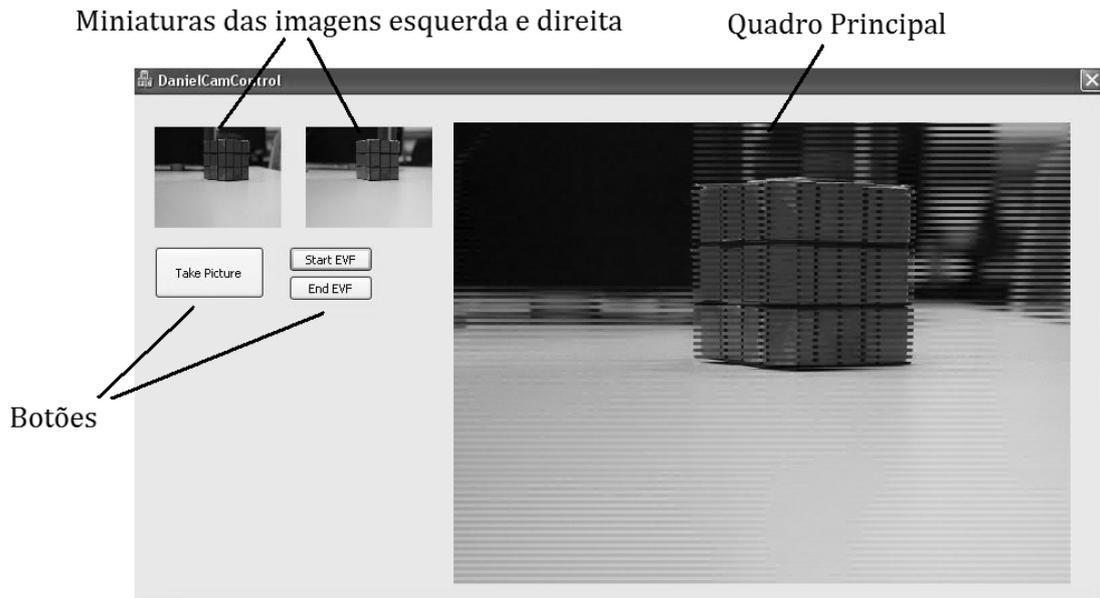


Figura 2.7 – Interface exibindo uma imagem estereoscópica

Ao iniciar o programa as câmeras são detectadas e, para cada uma, é criada a classe responsável por controlá-las. Em seguida é passado o ponteiro para as funções que irão tratar os eventos iniciados pelas câmeras. Após isso, se as conexões foram realizadas de maneira correta, é exibida a janela principal do programa.

O aplicativo pode ser dividido em duas partes: a interface entre o usuário e o *software* e a interface entre o *software* e as câmeras. A primeira parte é composta pelos elementos gráficos (botões e quadros onde são exibidas as imagens) e um gerenciador de eventos iniciados pelo usuário. Cada botão pressionado gera um evento que é tratado pelo aplicativo. Cada evento, por sua vez, inicia uma rotina de acordo com a Figura 2.8. As imagens exibidas nos quadros são resultado da rotina de *download* dos dados dos visores eletrônicos (*electronic viewfinder* – EVF), como mostrado na Figura 2.9.

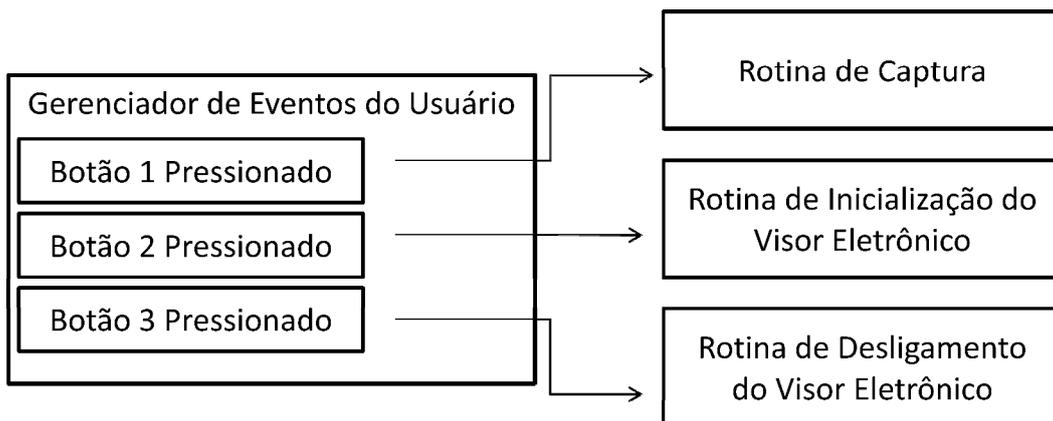


Figura 2.8 – Gerenciamento de eventos do usuário

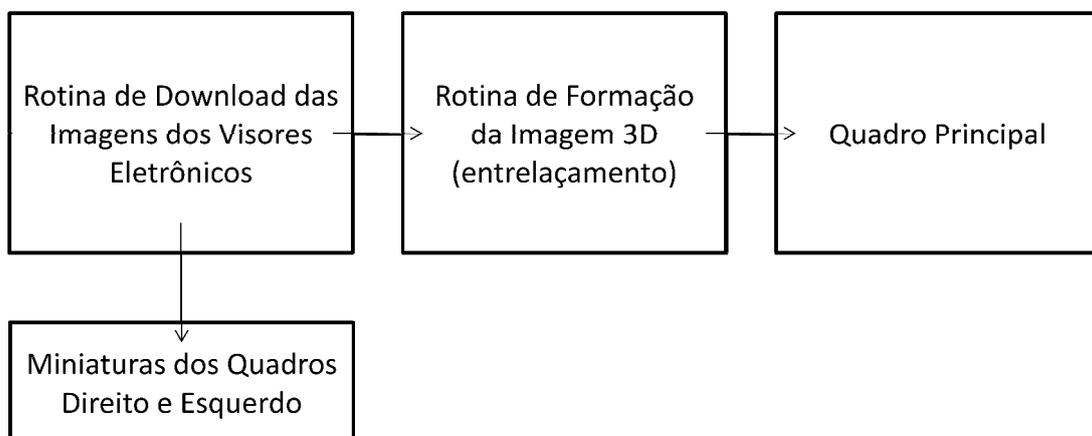


Figura 2.9 – Exibição das imagens para o usuário

A interface entre o programa e as câmeras foi desenvolvida utilizando as funções da biblioteca EDSK e as funções utilizadas no projeto são descritas na Tabela 2.1. As três rotinas iniciadas pelos eventos do usuário se comunicam com as câmeras por meio de trocas de mensagens, da mesma forma que as rotinas de *download* das fotografias e *download* das imagens dos visores eletrônicos (Figura 2.10).

Figura 2.10 – Troca de mensagens com o par de câmeras. As três primeiras rotinas enviam apenas uma mensagem para a câmera, enquanto que as últimas duas enviam mensagens e recebem de volta dados referentes às imagens capturadas, inclusive as próprias imagens eventualmente.

O botão de captura, quando pressionado, inicia a rotina de captura que envia uma mensagem para o par de câmeras solicitando o disparo das mesmas. A rotina de captura foi programada utilizando *threads*, de forma que as duas câmeras pudessem obter as fotografias simultaneamente, mas na prática há uma diferença de cerca de 100 milissegundos entre os disparos, devido provavelmente à estrutura da interface USB.

Logo após a captura das imagens, as câmeras enviam ao computador uma mensagem informando que há novos dados para serem baixados. Essa mensagem é tratada no gerenciador de eventos das câmeras (Figura 2.11), que, então, inicia a rotina de *download* das fotografias, solicitando o envio das imagens, que são recebidas e gravadas. As imagens gravadas são devidamente identificadas como pertencendo ao quadro esquerdo ou quadro direito, para serem usadas posteriormente nos aplicativos de calibração e dimensionamento.

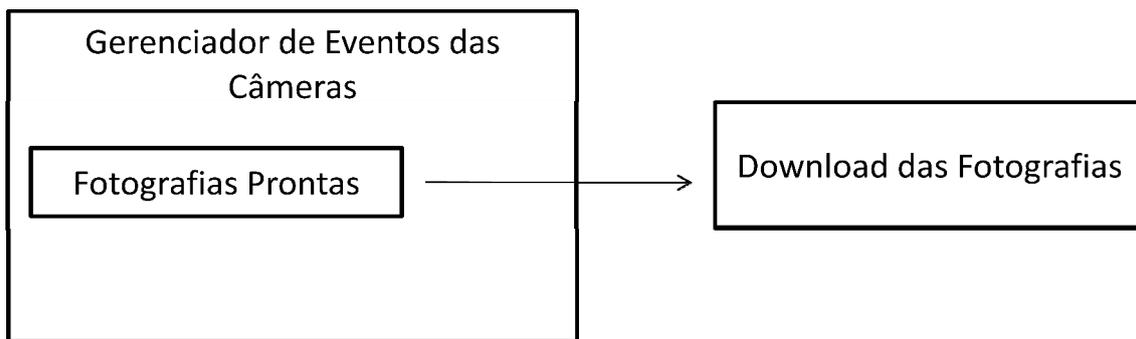


Figura 2.11 – Gerenciador de eventos das câmeras

O botão de inicialização do EVF inicia uma rotina que envia uma mensagem para as câmeras, solicitando que os visores eletrônicos sejam ligados. A partir desse momento, a rotina de *download* das imagens dos visores eletrônicos é iniciada e a imagem exata que é captada pelos sensores CMOS (*complementary metal-oxide silicon*) é enviada ao computador através da interface USB. Cada par de quadros das câmeras esquerda e direita é entrelaçado, de forma que nas linhas ímpares do quadro principal seja exibida a imagem da câmera esquerda e nas linhas pares a imagem da câmera direita. Sempre que o usuário move a janela na direção vertical, as linhas são reajustadas de modo que as imagens sejam sempre exibidas para os olhos corretos. Quando o visor eletrônico é desativado (pelo acionamento do respectivo botão), os sensores CMOS não recebem nenhuma informação visual, então os quadros não exibem mais o vídeo.

EdsInitializeSDK	Inicia a EDS SDK.
EdsGetCameraList	Obtém uma lista das câmeras conectadas.
EdsGetChildCount	Obtém o número de câmeras conectadas, para verificar se as duas câmeras estão ligadas corretamente.
EdsGetChildAtIndex	Seleciona uma câmera pelo seu índice (0 ou 1 no caso de duas câmeras) e associa a uma classe.
EdsOpenSession	Abre a sessão com uma câmera.
EdsSetObjectEventHandler	Determina onde serão tratados os eventos da câmera.
EdsSetPropertyData	Altera propriedades da câmera. Usado para ligar e desligar o visor eletrônico.
EdsSendCommand	Envia um comando para a câmera. É utilizado para efetuar o disparo.
EdsCreateMemoryStream	Aloca memória.
EdsCreateEvfImageRef	Cria uma referência para receber a imagem do visor eletrônico da câmera.
EdsDownloadEvfImage	Baixa a imagem do visor eletrônico.
EdsGetPropertyData	Lê propriedades da câmera.
EdsRelease	Libera memória.
EdsGetDirectoryItemInfo	Obtém informações sobre o arquivo de imagem
EdsCreateFileStream	Aloca memória para o arquivo de imagem.
EdsDownload	Baixa a fotografia.
EdsDownloadComplete	Realiza os últimos procedimentos após o download
EdsTerminateSDK	Finaliza a EDS SDK.
EdsGetPointer	Obtém o ponteiro para o primeiro endereço de memória de um determinado conjunto de dados. Utilizado para exibir as imagens na tela.
EdsGetLength	Obtém o tamanho de um vetor de um conjunto de dados. Usado para obter o tamanho das imagens a serem exibidas na tela.

Tabela 2.1 – Funções da EDS SDK

2.5 – Aplicativo de Calibração e Dimensionamento

A calibração é realizada em duas etapas, uma na qual o usuário marca na tela alguns pontos de referência em um *grid* de calibração (Figura 2.12) e a outra onde as coordenadas desses pontos de referência nas duas imagens e as coordenadas tridimensionais dos mesmos pontos são utilizadas para obter a matriz projetiva de cada câmera.

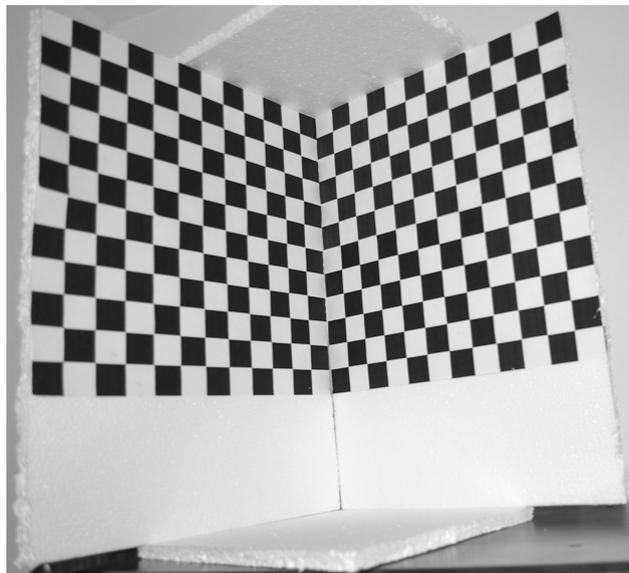


Figura 2.12 – *Grid* de calibração

A janela principal do aplicativo de calibração é formada por dezoito linhas (cada linha corresponde a um ponto de referência) de caixas de texto onde serão apresentadas as coordenadas dos pontos, um botão para carregar os dados do *grid* de calibração (coordenadas 3D dos pontos de referência) de um arquivo de texto e outro para iniciar os cálculos e um quadro no qual são exibidas as imagens direita e esquerda para a calibração (Figura 2.13).

Quando o botão de carregamento dos dados é pressionado, as informações das coordenadas 3D dos pontos de referência são carregadas e os valores da posição desses pontos nos eixos X, Y e Z são preenchidos nas devidas caixas de texto.

No quadro à direita é exibida inicialmente a imagem de calibração esquerda. Quando o usuário clica no primeiro ponto de referência, as coordenadas 2D desse ponto são preenchidas automaticamente e o quadro passa a mostrar a imagem de calibração direita. Esse processo se repete até o preenchimento das coordenadas dos dezoito pontos.

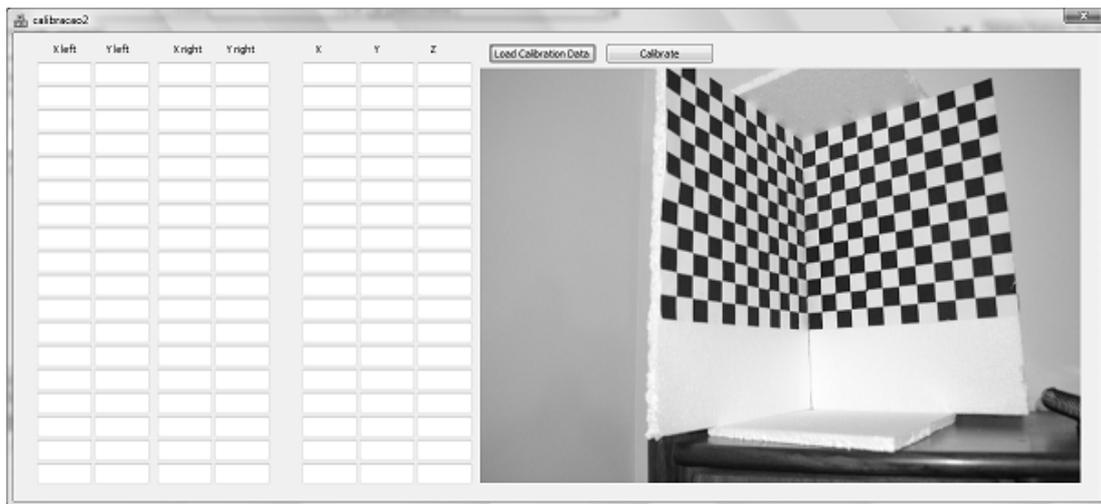


Figura 2.13 – Aplicativo de calibração

Quando o botão para iniciar a rotina de calibração é pressionado, todos os dados contidos nas caixas de texto são usados como parâmetros da função e as matrizes de projeção são calculadas. A rotina de calibração é apresentada detalhadamente no Capítulo 4 e o processo de escolha dos pontos de referência é detalhado no Capítulo 5.

Se o sistema já estiver calibrado, a rotina de dimensionamento poderá ser executada. A janela de dimensionamento (Figura 2.14) apresenta um quadro, onde são apresentadas alternadamente as duas imagens do par estereoscópico. O usuário marca na tela a primeira extremidade de um segmento que deseja medir nas imagens esquerda e direita e, em seguida, a segunda extremidade do segmento.

Pressionando o botão de dimensionamento, a rotina é executada, tendo como parâmetros as matrizes de projeção encontradas na etapa de calibração e as coordenadas das extremidades dos segmentos. O resultado da medida do segmento é exibido em uma caixa de texto ao lado do botão. A rotina de dimensionamento é explicada no Capítulo 4 e os resultados obtidos utilizando dois métodos diferentes são apresentados no Capítulo 5.



Figura 2.14 – Aplicativo de Dimensionamento

Capítulo 3

Modelagem Matemática

Neste capítulo será apresentado um modelo matemático para as câmeras utilizadas no sistema. Este modelo será utilizado para os métodos lineares de calibração do par de câmeras e dimensionamento de objetos na imagem. Para compreender os próximos tópicos deste capítulo é necessário introduzir alguns conceitos de Geometria Projetiva.

3.1 – Conceitos de Geometria Projetiva

Em um espaço projetivo de dimensão n , \mathcal{P}^n , um ponto é representado por um vetor de tamanho $n+1$, $\mathbf{x} = [x_1, \dots, x_{n+1}]^T$. Se existem dois pontos $\mathbf{x} = [x_1, \dots, x_{n+1}]^T$ e $\mathbf{y} = [y_1, \dots, y_{n+1}]^T$ e existe um escalar k diferente de zero, tal que $x_i = ky_i$, para $1 \leq i \leq n + 1$, então os vetores \mathbf{x} e \mathbf{y} representam o mesmo ponto, por definição.

O espaço de dimensão dois é chamado plano projetivo, \mathcal{P}^2 . Nele um ponto é definido por um vetor de três números $\mathbf{x} = [x_1, x_2, x_3]^T$. A reta, no plano projetivo, também é definida por um vetor de três elementos (u_1, u_2, u_3) , possuindo a seguinte equação:

$$\sum_{i=1}^3 u_i x_i = 0. \quad (3.1)$$

O espaço \mathcal{P}^3 é o espaço projetivo, no qual um ponto é definido por um vetor de quatro coordenadas $\mathbf{x} = [x_1, x_2, x_3, x_4]^T$. O plano no espaço projetivo é também definido por quatro coordenadas (u_1, u_2, u_3, u_4) , de acordo com a equação

$$\sum_{i=1}^4 u_i x_i = 0. \quad (3.2)$$

3.2 – Modelo de Câmera Pinhole

O modelo *pinhole* (em português buraco de agulha) é a representação mais simples de uma câmera fotográfica. Ele é composto de duas superfícies planas paralelas, uma das quais (plano focal) apresenta um pequeno orifício por onde os feixes de luz provenientes de um objeto podem passar projetando na outra superfície (plano da retina), a imagem desse objeto invertida, como visto na Figura 3.1.

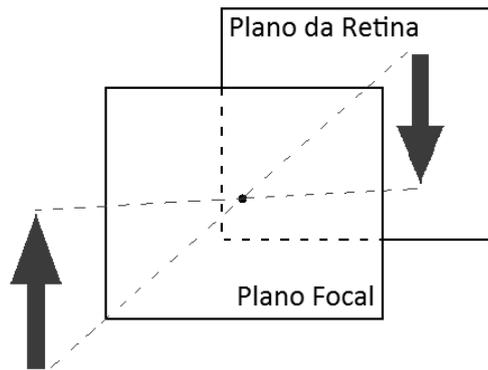


Figura 3.1 – Modelo *Pinhole*

3.3 – Matriz de Projeção

Para obter uma relação matemática entre as coordenadas 3D de um ponto e as coordenadas 2D da imagem desse ponto é necessário utilizar um modelo geométrico para a câmera. A Figura 3.2 apresenta esse modelo, onde o ponto c é o centro dos eixos do plano focal e o ponto C , o centro óptico, é a origem dos eixos das coordenadas 3D. Por esses dois pontos passa uma reta, chamada de eixo óptico, que é perpendicular aos planos focal e da retina.

O eixo z localiza-se sobre o eixo óptico e os outros dois eixos do espaço são situados sobre o plano focal. Analisando a figura, chega-se ao modelo matemático representado pela equação:

$$- - - (3.3)$$

E fazendo $u = U/S$ e $v = V/S$, pode-se escrever a Equação (3.3) na forma matricial:

$$(3.4)$$

Na Equação (3.3), U , V e S podem ser vistos como as coordenadas projetivas do ponto no plano da retina. Pode-se observar que se multiplicamos as coordenadas 3D do ponto M por um escalar k , as coordenadas U , V e S são multiplicadas pelo mesmo valor, mas como visto na Seção 3.1, se um vetor difere de outro por uma multiplicação de escalar, eles representam o mesmo ponto, então as novas coordenadas representam o mesmo ponto projetivo e as coordenadas euclidianas u e v também não são alteradas.

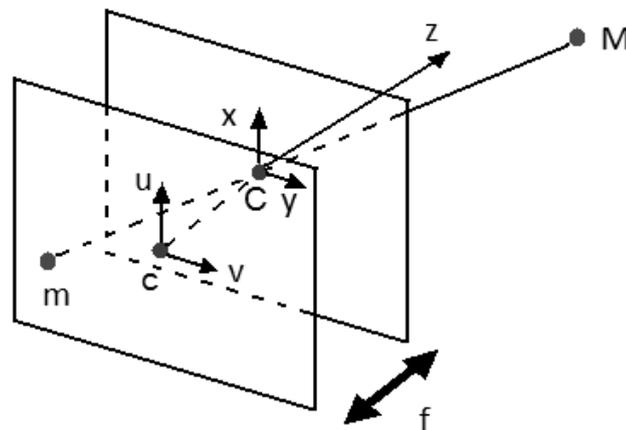


Figura 3.2 – Modelo geométrico da câmera

Pode-se utilizar também coordenadas projetivas para representar o ponto M , fazendo-se $x = X, y = U, z = Z$ e $T \neq 0$. E reescrevendo a Equação (3.4) tem-se:

$$(3.5)$$

Que pode ser reescrita como:

$$(3.6)$$

onde $\tilde{\mathbf{m}} = [U, V, S]^T$, $\tilde{\mathbf{M}} = [X, Y, Z, T]^T$ são as coordenadas projetivas do ponto na retina e no espaço respectivamente e $\tilde{\mathbf{P}}$ é a matriz de projeção, que recebe esse nome por levar as coordenadas de um ponto no espaço para as coordenadas de um outro ponto (sua projeção) em um plano.

Comparando as Equações (3.5) e (3.3) nota-se que utilizando geometria projetiva, um problema inicialmente não linear na geometria euclidiana, torna-se linear, sendo essa a grande vantagem de se utilizar a geometria projetiva neste trabalho.

Pode-se fazer uma interpretação geométrica das linhas da matriz $\tilde{\mathbf{P}}$. Sendo $\tilde{\mathbf{Q}}_i^T = [\mathbf{q}_i^T, q_{i4}]$, $i = 1, 2, 3$ as três linhas da matriz $\tilde{\mathbf{P}}$, pode-se escrever:

$$\tilde{\mathbf{P}} = \begin{bmatrix} \tilde{\mathbf{Q}}_1^T \\ \tilde{\mathbf{Q}}_2^T \\ \tilde{\mathbf{Q}}_3^T \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1^T & q_{14} \\ \mathbf{q}_2^T & q_{24} \\ \mathbf{q}_3^T & q_{34} \end{bmatrix}. \quad (3.7)$$

A equação $\tilde{\mathbf{Q}}_3^T \tilde{\mathbf{M}} = 0$ implica $Z = 0$, ou seja, representa o plano focal. Para $\tilde{\mathbf{Q}}_2^T \tilde{\mathbf{M}} = 0$, tem-se $V = 0$, que é o plano perpendicular ao plano da retina, tendo como interseção a reta $v = 0$. De forma semelhante, a equação $\tilde{\mathbf{Q}}_1^T \tilde{\mathbf{M}} = 0$ define o plano perpendicular ao plano da retina, tendo a reta $u = 0$ como interseção.

3.4 – Parâmetros Intrínsecos e Extrínsecos

Na seção anterior foi definida a matriz de projeção para o modelo ilustrado na Figura 3.2, onde os centros das coordenadas 3D e 2D estão sobre os planos focal e da retina, respectivamente, e ligados por uma reta perpendicular a esses planos. Nesta seção será explicado o que acontece quando há alterações nos eixos das coordenadas e essas alterações serão associadas a alguns parâmetros que são chamados de parâmetros intrínsecos e extrínsecos da câmera.

3.4.1 – Parâmetros Intrínsecos

No primeiro modelo de câmera analisado neste trabalho, foi considerado que a origem dos eixos de coordenadas 2D era a interseção do eixo óptico com o plano da retina e que a unidade em cada um dos eixos era a mesma que no sistema 3D. Agora será estudado o caso em que tanto a origem quanto a escala dos eixos no plano da retina são alterados.

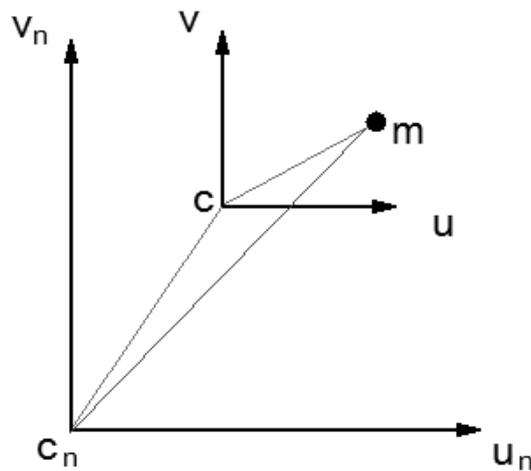


Figura 3.3 – Novo sistema de coordenadas do plano da retina

A Figura 3.3 ilustra a forma geral do novo sistema de coordenadas, onde c_n é o novo centro e u_n e v_n são as novas coordenadas, cujas unidades são as antigas multiplicadas por k_u e k_v , respectivamente. O novo vetor de coordenadas m_n de um ponto m no novo sistema, então, é a soma das coordenadas da antiga origem c (u_0 , v_0) com a antiga coordenada do ponto m , mas com as novas unidades, conforme:

$$(3.8)$$

onde $\mathbf{t} = [u_0, v_0]^T$.

E utilizando as coordenadas projetivas, pode-se escrever

$$(3.9)$$

Para obter a nova matriz de projeção $\tilde{\mathbf{P}}_n$, substitui-se $\tilde{\mathbf{m}}$ da Equação (3.9) por seu valor na Equação (3.6), obtendo-se então:

$$\tilde{\mathbf{m}}_n = \begin{bmatrix} \mathbf{s} & \mathbf{t} \\ \mathbf{0}_2^T & 1 \end{bmatrix} \tilde{\mathbf{P}}\tilde{\mathbf{M}}. \quad (3.10)$$

E, definindo $\tilde{\mathbf{H}} = \begin{bmatrix} \mathbf{s} & \mathbf{t} \\ \mathbf{0}_2^T & 1 \end{bmatrix}$:

$$\tilde{\mathbf{P}}_n = \tilde{\mathbf{H}}\tilde{\mathbf{P}}. \quad (3.11)$$

Utilizando a matriz $\tilde{\mathbf{P}}$ da Equação (3.5):

$$\tilde{\mathbf{P}}_n = \begin{bmatrix} -fk_u & 0 & u_0 & 0 \\ 0 & -fk_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \quad (3.12)$$

Fazendo $\alpha_u = -fk_u$ e $\alpha_v = -fk_v$, temos os quatro parâmetros intrínsecos da câmera (α_u , α_v , u_0 e v_0). O uso da palavra “intrínseco” vem do fato desses parâmetros não dependerem da posição da câmera no espaço.

3.4.2 – Parâmetros Extrínsecos

No modelo estudado na Seção 3.3, a origem do sistema de coordenadas 3D localizava-se sobre o plano focal, dois de seus eixos estavam contidos nesse plano e o terceiro eixo era perpendicular ao mesmo. Na prática, é difícil e desnecessário encontrar a posição exata do plano focal e também é mais cômodo posicionar o sistema de coordenadas de forma a facilitar a medição dos pontos de referência no *grid* de calibração. Portanto, é necessário analisar as mudanças que ocorrem na matriz de projeção quando os eixos do sistema 3D sofrem alterações em relação ao primeiro modelo.

A Figura 3.4 ilustra a forma geral do novo sistema de coordenadas, onde C_n é a nova origem desse sistema e x_n , y_n e z_n são as novas coordenadas do ponto M . O novo vetor \mathbf{M}_n que representa o ponto M é, então, a soma do segmento C_nC ,

representado pelo vetor \mathbf{r} , com as antigas coordenadas do ponto, mas com as unidades do novo sistema:

$$(3.13)$$

onde \mathbf{R} é a matriz de rotação e escalamento dos eixos.

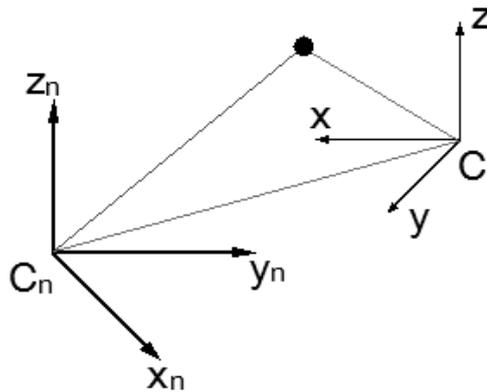


Figura 3.4 – Novo sistema de coordenadas do espaço

Utilizando coordenadas projetivas e definindo \mathbf{p} , a Equação (3.13) pode ser reescrita como:

$$\mathbf{p} = \mathbf{K} \mathbf{R} \mathbf{r} + \mathbf{t} \quad (3.14)$$

Comparando as Equações (3.14) e (3.6), obtém-se a nova matriz de projeção \mathbf{K} :

$$\mathbf{K} = \mathbf{K} \mathbf{R} \quad (3.15)$$

então,

$$\mathbf{p} = \mathbf{K} \mathbf{R} \mathbf{r} + \mathbf{t} \quad (3.16)$$

A matriz \mathbf{K} e o vetor \mathbf{t} são chamados parâmetros extrínsecos da câmera, pois ao contrário dos parâmetros intrínsecos dependem da posição e orientação da câmera no espaço, já que eles definem o deslocamento da origem, a rotação e o

escalamento do sistema de coordenadas 3D com relação ao modelo anterior onde a origem estava localizada no plano focal e os eixos eram dois paralelos e um perpendicular a esse plano.

Substituindo na Equação (3.16) a matriz $\tilde{\mathbf{P}}_n$ pela matriz de projeção da Equação (3.12), já levando em consideração os parâmetros intrínsecos da câmera, a forma geral da matriz de projeção é:

$$\tilde{\mathbf{P}} = \begin{bmatrix} \alpha_u \mathbf{r}_1 + u_0 \mathbf{r}_3 & \alpha_u t_x + u_0 t_z \\ \alpha_v \mathbf{r}_2 + v_0 \mathbf{r}_3 & \alpha_v t_y + v_0 t_z \\ \mathbf{r}_3 & t_z \end{bmatrix}. \quad (3.17)$$

Capítulo 4

Calibração e Dimensionamento

Neste capítulo são estudadas as técnicas de calibração e dimensionamento utilizadas no projeto. Primeiro será apresentada a técnica de calibração linear e, em seguida, os dois métodos também lineares de dimensionamento.

4.1 – Calibração

Neste trabalho, o objetivo final é conseguir medir, a partir de duas imagens capturadas por um par de câmeras, qualquer distância entre dois pontos que estejam presentes nas duas imagens.

O processo completo de calibração consiste em estimar todos os parâmetros intrínsecos e extrínsecos da câmera. Pode ser dividido em duas etapas: obtenção da matriz de projeção e obtenção dos parâmetros intrínsecos e extrínsecos a partir da matriz de projeção.

Para o cálculo da distância entre dois pontos, apenas a primeira etapa do processo de calibração será necessária.

4.1.1 – Restrições

Para estimar a matriz $\tilde{\mathbf{P}}$, é necessário saber as condições necessárias para que ela possa ser escrita como na Equação (3.17). Segundo [1], a Equação (3.7) pode ser escrita na forma da Equação (3.17) se e somente se forem satisfeitas as seguintes restrições:

$$\|\mathbf{q}_3\|^2 = 1 \quad (4.1)$$

e

$$(\mathbf{q}_1 \wedge \mathbf{q}_3) \cdot (\mathbf{q}_2 \wedge \mathbf{q}_3) = 0. \quad (4.2)$$

4.1.2 – Método Linear de Calibração

Utilizando as Equações (3.6) e (3.7), para um dado ponto \mathbf{M}_i , em coordenadas euclidianas, é possível obter duas equações lineares tendo \mathbf{q}_m e q_{m4} como variáveis. Realizando o produto do lado direito da primeira equação, pode-se escrever

$$U = \mathbf{q}_1^T \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + q_{14}T, \quad (4.3)$$

$$V = \mathbf{q}_2^T \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + q_{24}T, \quad (4.4)$$

$$S = \mathbf{q}_3^T \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + q_{34}T. \quad (4.5)$$

Fazendo $U = Su$ e $V = Sv$ nas Equações (4.3) e (4.4) e substituindo S por seu valor na Equação (4.5) e em seguida dividindo ambos os lados das equações resultantes por T, chega-se a duas equações lineares com coordenadas euclidianas:

$$\mathbf{q}_1^T \mathbf{M}_i - u_i \mathbf{q}_3^T \mathbf{M}_i + q_{14} - u_i q_{34} = 0 \quad (4.6)$$

e

$$\mathbf{q}_2^T \mathbf{M}_i - v_i \mathbf{q}_3^T \mathbf{M}_i + q_{24} - v_i q_{34} = 0 \quad (4.7)$$

Reunindo todas as variáveis em um só vetor $\mathbf{q} = [\mathbf{q}_1^T, q_{14}, \mathbf{q}_2^T, q_{24}, \mathbf{q}_3^T, q_{34}]^T$, as Equações (4.7) e (4.8) podem ser reescritas como:

$$\begin{bmatrix} \mathbf{M}_i & 1 & \mathbf{0}_3^T & 0 & -u_i \mathbf{M}_i & -u_i \\ \mathbf{0}_3^T & 0 & \mathbf{M}_i & 1 & -v_i \mathbf{M}_i & -v_i \end{bmatrix} \mathbf{q} = 0. \quad (4.8)$$

Para N pontos de referência, conseguem-se $2N$ equações lineares. Sendo \mathbf{A} a matriz $2N \times 12$ que depende das coordenadas 2D e 3D dos pontos de referência, tem-se o sistema

$$\mathbf{A}\mathbf{q} = 0. \quad (4.9)$$

Para evitar a solução trivial $\mathbf{q} = 0$, é necessária a utilização de alguma restrição. Pode-se utilizar a restrição $\|\mathbf{q}_3\|^2 = 1$ e o problema torna-se minimizar $\|\mathbf{A}\mathbf{q}\|^2$, sujeito a essa condição, que é um problema de minimização com restrição, o qual pode ser resolvido utilizando o método dos multiplicadores de Lagrange.

Na prática, para que os eixos do sistema 2D fossem perfeitamente ortogonais, seria necessário utilizar a restrição $(\mathbf{q}_1 \wedge \mathbf{q}_3) \cdot (\mathbf{q}_2 \wedge \mathbf{q}_3) = 0$ [1], mas como mesmo desconsiderando essa restrição o ângulo entre os eixos geralmente fica muito próximo de 90° e como não há forma fechada para o problema de minimização de $\|\mathbf{A}\mathbf{q}\|^2$ sujeito a ela, pode-se utilizar apenas a primeira restrição. Nesse caso, como os eixos do sistema 2D podem formar um ângulo diferente de 90° , acrescenta-se um novo parâmetro ao conjunto de parâmetros intrínsecos da câmera: o ângulo θ entre os eixos do sistema de coordenadas do plano da retina.

Sendo $\|\mathbf{q}_3\|^2 = 1$ a única restrição a ser considerada, deve-se minimizar $\|\mathbf{A}\mathbf{q}\|^2$ sujeito a essa restrição utilizando o método dos multiplicadores de Lagrange. Dada uma função $f(x,y)$ e a restrição $g(x,y) = c$, é introduzida a função

$$\Lambda(x, y, \lambda) = f(x, y) + \lambda(g(x, y) - c). \quad (4.10)$$

E o problema se torna minimizar a nova função $\Lambda(x,y,\lambda)$. Para isso deve-se resolver a equação

$$\nabla_{x,y,\lambda}\Lambda(x, y, \lambda) = 0 \quad (4.11)$$

onde $\nabla_{x,y,\lambda}\Lambda = \left(\frac{\partial\Lambda}{\partial x}, \frac{\partial\Lambda}{\partial y}, \frac{\partial\Lambda}{\partial\lambda}\right)$.

Para aplicar esse método ao problema em questão, primeiro divide-se a matriz \mathbf{A} em duas matrizes \mathbf{C} e \mathbf{D} , de modo que seja possível separar a variável \mathbf{q}_3 das outras variáveis, já que a restrição é uma função apenas de \mathbf{q}_3 :

$$\mathbf{A}\mathbf{q} = \mathbf{C} \begin{bmatrix} \mathbf{q}_1^T \\ q_{14} \\ \mathbf{q}_2^T \\ q_{24} \\ q_{34} \end{bmatrix} + \mathbf{D}\mathbf{q}_3. \quad (4.12)$$

Então, sendo \mathbf{q}_n o vetor $[\mathbf{q}_1, q_{14}, \mathbf{q}_2, q_{24}, q_{34}]^T$, o problema de minimizar $\|\mathbf{Aq}\|^2$ torna-se equivalente a minimizar

$$R = \|\mathbf{Cq}_n + \mathbf{Dq}_3\|^2 + \lambda(1 - \|\mathbf{q}_3\|^2). \quad (4.13)$$

Fazendo a derivada parcial da nova função com relação às variáveis \mathbf{q}_n e \mathbf{q}_3 , tem-se como resultado

$$\frac{\partial R}{\partial \mathbf{q}_n} = 2(\mathbf{C}^T \mathbf{Cq}_n + \mathbf{C}^T \mathbf{Dq}_3), \quad (4.14)$$

e

$$\frac{\partial R}{\partial \mathbf{q}_3} = 2(\mathbf{D}^T \mathbf{Dq}_3 + \mathbf{D}^T \mathbf{Cq}_n - \lambda \mathbf{q}_3). \quad (4.15)$$

E igualando as duas derivadas parciais à zero, encontra-se

$$\mathbf{q}_n = -(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T \mathbf{Dq}_3 \quad (4.16)$$

e

$$\mathbf{D}^T \mathbf{Dq}_3 + \mathbf{D}^T \mathbf{Cq}_n = \lambda \mathbf{q}_3. \quad (4.17)$$

Substituindo \mathbf{q}_n na Equação (4.17) de acordo com a Equação (4.16), a segunda equação pode ser reescrita como

$$(\mathbf{D}^T \mathbf{D} + \mathbf{D}^T \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T \mathbf{D})\mathbf{q}_3 = \lambda \mathbf{q}_3. \quad (4.18)$$

A Equação (4.18) mostra que \mathbf{q}_3 é um autovetor da matriz $\mathbf{E} = \mathbf{D}^T \mathbf{D} + \mathbf{D}^T \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T \mathbf{D}$ e λ é o autovalor associado a esse autovetor.

Substituindo \mathbf{q}_n na equação do problema inicial $\|\mathbf{Cq}_n + \mathbf{Dq}_3\|^2$ pelo seu valor na Equação (4.16), obtém-se

$$\|\mathbf{Cq}_n + \mathbf{Dq}_3\|^2 = \|-\mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T \mathbf{Dq}_3 + \mathbf{Dq}_3\|^2.$$

Desenvolvendo-se essa equação, pode-se reescrevê-la como

$$\|\mathbf{C}\mathbf{q}_n + \mathbf{D}\mathbf{q}_3\|^2 = \mathbf{q}_3^T \mathbf{D}^T (\mathbf{I} - \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T) \mathbf{D} \mathbf{q}_3.$$

O fator $\mathbf{D}^T (\mathbf{I} - \mathbf{C}(\mathbf{C}^T \mathbf{C})^{-1} \mathbf{C}^T) \mathbf{D} \mathbf{q}_3$ é equivalente ao lado esquerdo da Equação (4.18), portanto pode ser substituído por $\lambda \mathbf{q}_3$, resultando

$$\mathbf{q}_3^T \lambda \mathbf{q}_3 = \lambda \|\mathbf{q}_3\| = \lambda.$$

Então, o valor do erro para o problema da minimização de $\|\mathbf{A}\mathbf{q}\|^2$ com a restrição $\|\mathbf{q}_3\|^2 = 1$ é λ , autovalor da matriz \mathbf{E} . Sendo assim, para obter o menor erro possível, o vetor \mathbf{q}_3 deve ser o autovetor correspondente ao menor autovalor da matriz \mathbf{E} . E o vetor \mathbf{q}_n é obtido substituindo o valor de \mathbf{q}_3 na Equação (4.16).

Realizando esse processo para as câmeras esquerda e direita, são estimadas as duas matrizes de projeção $\tilde{\mathbf{P}}_e$ e $\tilde{\mathbf{P}}_d$ e, de posse delas, é possível realizar os procedimentos de dimensionamento.

4.2 – Dimensionamento

Tendo estimado as matrizes de projeção do par de câmeras, é possível encontrar a coordenada 3D de pontos presentes em duas imagens, uma obtida por cada câmera. Nesta seção são apresentados dois métodos lineares de dimensionamento.

4.2.1 – Método dos Multiplicadores de Lagrange

De forma semelhante ao que foi apresentado na Seção 4.1.2, o método dos multiplicadores de Lagrange pode ser utilizado também para obter a solução de um problema de estimar as coordenadas 3D de um ponto a partir de pelo menos duas projeções deste ponto em câmeras diferentes.

É necessário, então, escrever a equação das coordenadas 3D de um ponto na forma $\mathbf{A}\tilde{\mathbf{M}} = \mathbf{0}$. Fazendo novamente $U = Su$ e $V = Sv$ nas Equações (4.3) e (4.4) e substituindo S por seu valor na Equação (4.5), pode-se reescrevê-las como

$$u \left(\mathbf{q}_3^T \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + q_{34}T \right) = \mathbf{q}_1^T \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + q_{14}T \quad (4.21)$$

e

$$v \left(\mathbf{q}_3^T \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + q_{34}T \right) = \mathbf{q}_2^T \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + q_{24}T \quad (4.22)$$

E reorganizando as equações tem-se

$$\begin{bmatrix} \mathbf{Q}_1^T - u\mathbf{Q}_3^T \\ \mathbf{Q}_2^T - v\mathbf{Q}_3^T \end{bmatrix} \tilde{\mathbf{M}} = \mathbf{0}. \quad (4.23)$$

Sendo $\tilde{\mathbf{m}}_e$ e $\tilde{\mathbf{m}}_d$ os vetores das coordenadas projetivas de um mesmo ponto nas imagens da câmera esquerda e da câmera direita, respectivamente, e $\tilde{\mathbf{P}}_e$ e $\tilde{\mathbf{P}}_d$ as matrizes de projeção dessas câmeras, tem-se o par de equações

$$\tilde{\mathbf{m}}_e = \tilde{\mathbf{P}}_e \tilde{\mathbf{M}} \quad (4.24)$$

e

$$\tilde{\mathbf{m}}_d = \tilde{\mathbf{P}}_d \tilde{\mathbf{M}} \quad (4.25)$$

Cada equação pode ser reescrita na forma da Equação (4.23) e, usando as duas equações, chega-se à equação da forma $\mathbf{A}\tilde{\mathbf{M}} = \mathbf{0}$

$$\begin{bmatrix} \mathbf{Q}_{1e}^T - u_e \mathbf{Q}_{3e}^T \\ \mathbf{Q}_{2e}^T - v_e \mathbf{Q}_{3e}^T \\ \mathbf{Q}_{1d}^T - u_d \mathbf{Q}_{3d}^T \\ \mathbf{Q}_{2d}^T - v_d \mathbf{Q}_{3d}^T \end{bmatrix} \tilde{\mathbf{M}} = \mathbf{0}. \quad (4.26)$$

Utilizando a restrição $\|\tilde{\mathbf{M}}\|^2 = 1$ e aplicando o método dos multiplicadores de Lagrange, o problema de minimizar $\|\mathbf{A}\tilde{\mathbf{M}}\|^2$ torna-se equivalente a minimizar

$$R = \|\mathbf{A}\tilde{\mathbf{M}}\|^2 + \lambda(1 - \|\tilde{\mathbf{M}}\|^2). \quad (4.27)$$

Derivando a função R com relação a $\tilde{\mathbf{M}}$ e igualando o resultado a zero, encontra-se

$$\frac{\partial R}{\partial \tilde{\mathbf{M}}} = 2(\mathbf{A}^T \mathbf{A} \tilde{\mathbf{M}} - \lambda \tilde{\mathbf{M}}) = 0 \quad (4.28)$$

e, então,

$$\mathbf{A}^T \mathbf{A} \tilde{\mathbf{M}} = \lambda \tilde{\mathbf{M}}. \quad (4.29)$$

Pela Equação (4.29), λ é um autovalor da matriz $\mathbf{A}^T \mathbf{A}$ e $\tilde{\mathbf{M}}$ é o autovetor associado a esse autovalor. Desenvolvendo a expressão inicial, $\|\mathbf{A}\tilde{\mathbf{M}}\|^2$, tem-se

$$\|\mathbf{A}\tilde{\mathbf{M}}\|^2 = (\mathbf{A}^T \mathbf{A} \tilde{\mathbf{M}})^T \tilde{\mathbf{M}}.$$

E substituindo $\mathbf{A}^T \mathbf{A} \tilde{\mathbf{M}}$ por seu valor na Equação (4.29) tem-se:

$$\lambda \tilde{\mathbf{M}}^T \tilde{\mathbf{M}} = \lambda \|\tilde{\mathbf{M}}\|^2 = \lambda.$$

Esse resultado mostra que o erro da minimização é um autovalor da matriz $\mathbf{A}^T \mathbf{A}$. Portanto, o menor erro possível acontece quando $\tilde{\mathbf{M}}$ é o autovetor associado ao menor autovalor de $\mathbf{A}^T \mathbf{A}$.

Dado um segmento cujas extremidades são visíveis nas duas imagens do par estereoscópico, pode-se obter a estimativa das coordenadas projetivas das duas extremidades, representadas pelos vetores $\tilde{\mathbf{M}}_1$ e $\tilde{\mathbf{M}}_2$. Dividindo as três primeiras coordenadas pela quarta, encontram-se as coordenadas euclidianas $\mathbf{M}_1 = [x_1, y_1, z_1]^T$ e $\mathbf{M}_2 = [x_2, y_2, z_2]^T$. Então, o comprimento d do segmento é dado pela distância entre \mathbf{M}_1 e \mathbf{M}_2 :

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}. \quad (4.30)$$

4.2.2 – Método dos Mínimos Quadrados Ordinários

Um segundo método linear para estimar a distância tridimensional entre dois pontos é o método dos mínimos quadrados ordinários (MQO), que consiste em minimizar o quadrado do erro de estimação. Dada a variável \mathbf{Y} e a matriz \mathbf{X} de parâmetros relacionados a \mathbf{Y} , deve-se encontrar o vetor \mathbf{b} tal que o quadrado da norma do vetor de erro $\mathbf{e} = \mathbf{Y} - \mathbf{X}\mathbf{b}$ é mínimo.

Derivando $\|\mathbf{e}\|^2$ com relação a \mathbf{b} , encontra-se

$$\frac{\partial \|\mathbf{e}\|^2}{\partial \mathbf{b}} = \frac{\partial (\mathbf{Y}^T \mathbf{Y} - 2\mathbf{Y}^T \mathbf{X}\mathbf{b} + \mathbf{b}^T \mathbf{X}^T \mathbf{X}\mathbf{b})}{\partial \mathbf{b}} = -2\mathbf{X}^T \mathbf{Y} + 2\mathbf{X}^T \mathbf{X}\mathbf{b} \quad (4.31)$$

e igualando o resultado a zero, tem-se

$$-2\mathbf{X}^T \mathbf{Y} + 2\mathbf{X}^T \mathbf{X}\mathbf{b} = \mathbf{0} \quad (4.32)$$

e

$$\mathbf{b} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y}. \quad (4.33)$$

Das Equações (4.6) e (4.7), considerando \mathbf{M} como variável, segue a equação

$$\begin{bmatrix} \mathbf{q}_1^T - u_i \mathbf{q}_3^T \\ \mathbf{q}_2^T - v_i \mathbf{q}_3^T \end{bmatrix} \mathbf{M} = \begin{bmatrix} -q_{14} + u_i q_{34} \\ -q_{24} + v_i q_{34} \end{bmatrix}. \quad (4.34)$$

Para dois pontos de coordenadas 2D $\mathbf{m}_e = [u_1, v_1]^T$ e $\mathbf{m}_d = [u_2, v_2]^T$, a Equação (4.34) pode ser reescrita como:

$$\begin{bmatrix} \mathbf{q}_{1e}^T - u_e \mathbf{q}_{3e}^T \\ \mathbf{q}_{2e}^T - v_e \mathbf{q}_{3e}^T \\ \mathbf{q}_{1d}^T - u_d \mathbf{q}_{3d}^T \\ \mathbf{q}_{2d}^T - v_d \mathbf{q}_{3d}^T \end{bmatrix} \mathbf{M} = \begin{bmatrix} -q_{14e} + u_e q_{34e} \\ -q_{24e} + v_e q_{34e} \\ -q_{14d} + u_d q_{34d} \\ -q_{24d} + v_d q_{34d} \end{bmatrix}, \quad (4.35)$$

$$\mathbf{AM} = \mathbf{b}. \quad (4.36)$$

De acordo com a Equação (4.33), a solução da Equação (4.36) é dada por

$$\mathbf{M} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}. \quad (4.37)$$

Para um segmento qualquer cujas extremidades são visíveis nas duas imagens do par estereoscópico, é possível então estimar as coordenadas 3D dessas extremidades utilizando a Equação (4.37) e o comprimento do segmento é dado pela Equação (4.30).

Capítulo 5

Resultados

Neste capítulo são apresentados os resultados obtidos utilizando o aplicativo de dimensionamento, calibração, visualização e aquisição de imagens apresentados nos capítulos anteriores. Inicialmente são descritos os procedimentos de calibração e dimensionamento e, em seguida, são apresentados os resultados de uma série de dimensionamentos.

5.1 – Procedimentos de Calibração e Dimensionamento

Para realizar a calibração foi construído um *grid* formado por duas placas, cada uma com um padrão quadriculado preto e branco formado por doze linhas e doze colunas de quadrados com lados de três centímetros. Os dois padrões quadriculados se unem exatamente na junção das placas conforme a Figura 2.9.

A rotina de calibração foi realizada utilizando o software desenvolvido no projeto. As rotinas de calibração e dimensionamento desenvolvidas neste projeto seguem os conceitos explicados no Capítulo 4 e se baseiam na implementação apresentada em [3]. Foram utilizados dois pares de imagens (Figura 5.1 e Figura 5.2) para testar a calibração das câmeras em duas posições diferentes.

Os pontos do *grid* utilizados para a calibração foram marcados em vermelho na Figura 5.3. A partir desses pontos foi criado um arquivo de texto com as suas coordenadas tridimensionais. Considerou-se o eixo z a junção entre as duas placas, o eixo x a linha inferior que delimita o padrão quadriculado da placa à esquerda nas imagens e o eixo y a linha inferior que delimita o padrão quadriculado da placa à direita nas imagens.

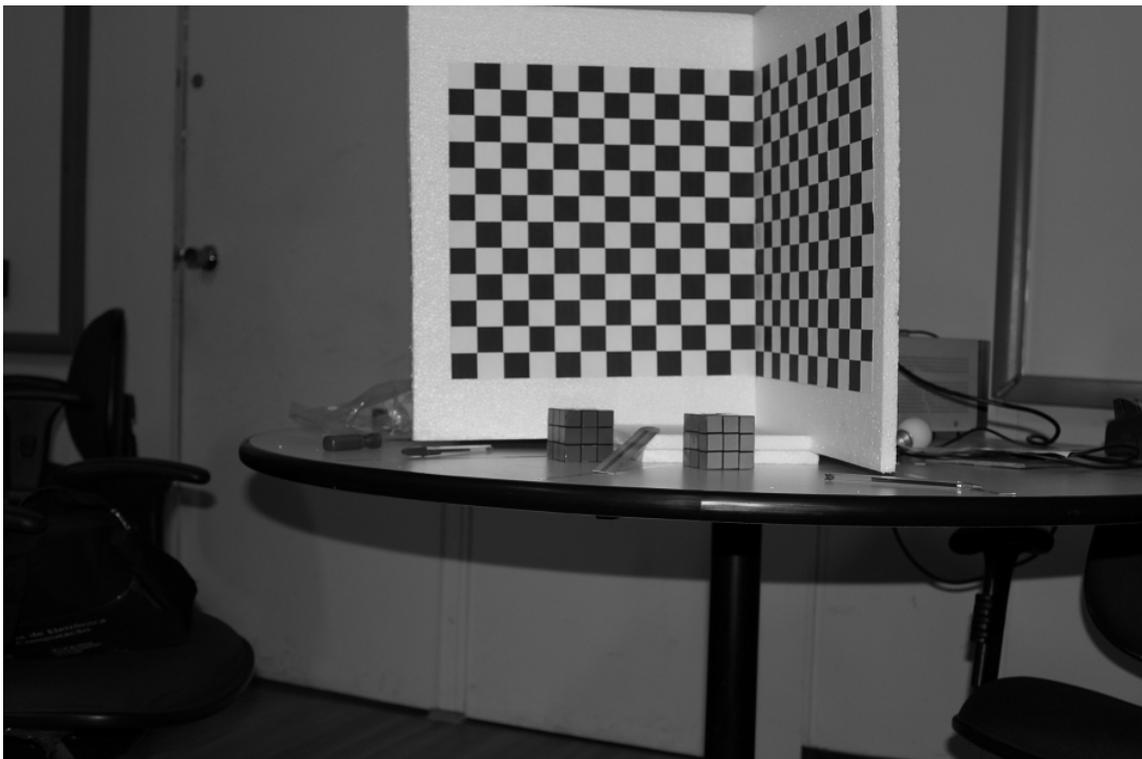
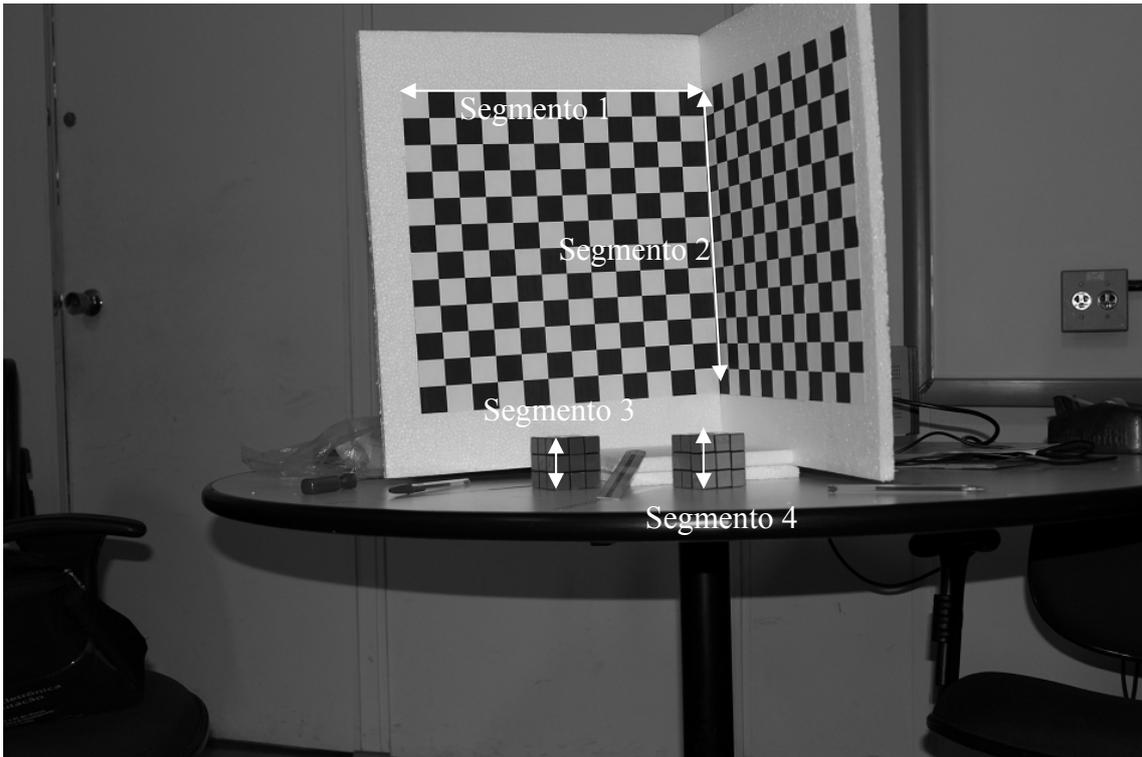


Figura 5.1 – Par de imagens para calibração com o *grid* mais próximo das câmeras

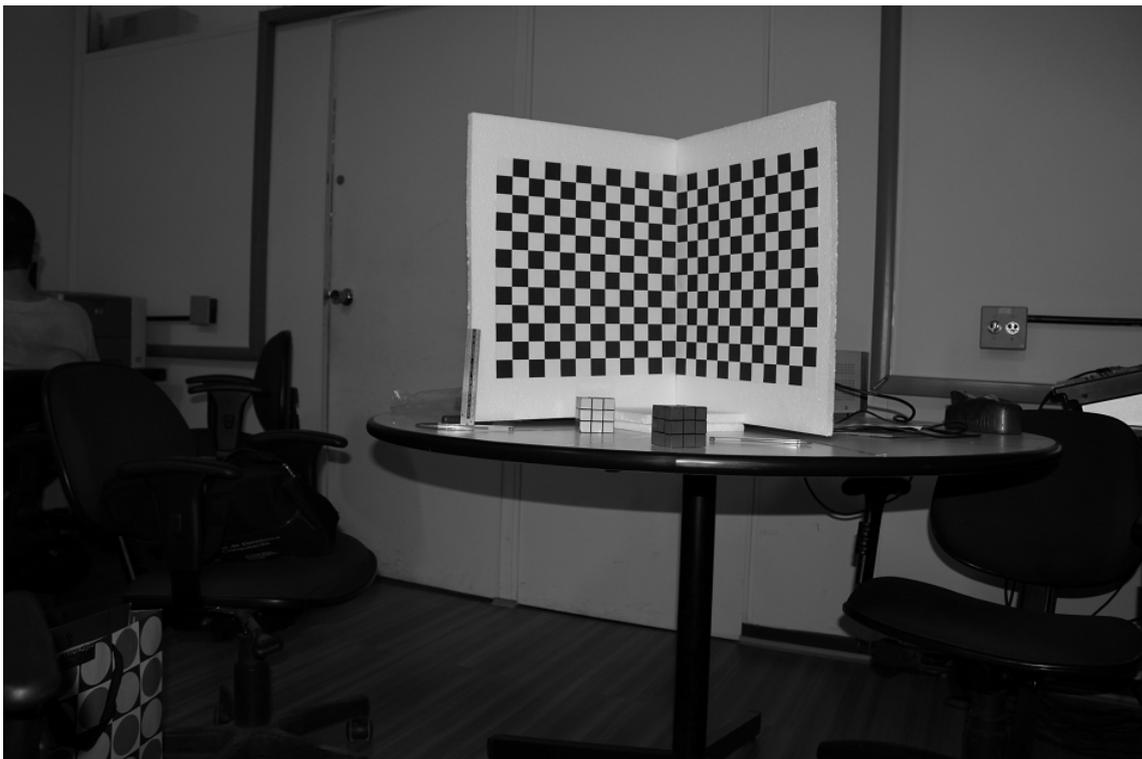


Figura 5.2 – Par de imagens para calibração com o *grid* mais afastado das câmeras

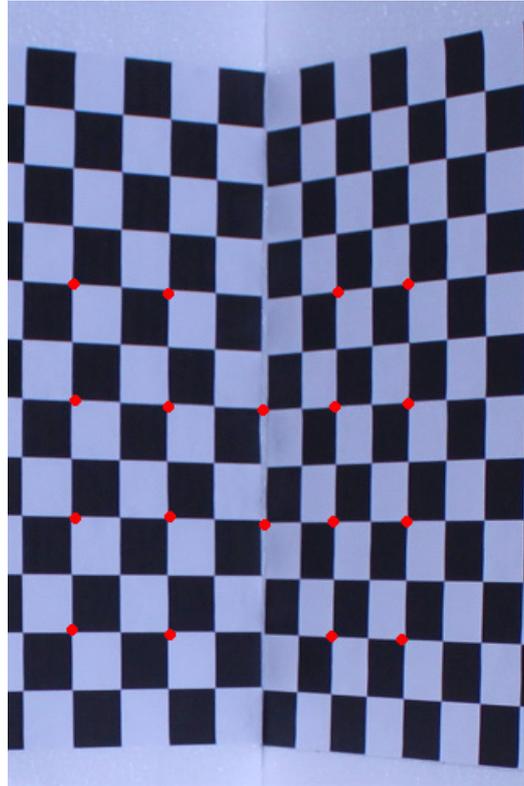


Figura 5.3 – Pontos de referência para calibração

O dimensionamento foi realizado utilizando as matrizes projetivas encontradas no processo de calibração descrito anteriormente e também foi feito utilizando o software desenvolvido no projeto (conforme o Capítulo 4 e as referências [1] e [3]).

Para testar a rotina de dimensionamento, foram utilizados quatro pares de imagens, sendo dois deles os mesmos utilizados para a calibração e os outros dois compostos por imagens sem a presença do *grid* de calibração (Figura 5.4 e Figura 5.5). Foram medidos segmentos do próprio *grid* e objetos que aparecem nas imagens. Esses segmentos estão marcados nas Figuras 5.1, 5.2, 5.4 e 5.5 e numerados de 1 a 14.



Figura 5.4 – Par de imagens para dimensionamento
com o *grid* mais próximo das câmeras



Figura 5.5 – Par de imagens para dimensionamento
com o *grid* mais afastado das câmeras

Utilizando imagens de alta resolução e realizando os procedimentos de marcação de pixels nas imagens manualmente, é possível ter uma boa precisão nas coordenadas 2D dos pontos de referência e das extremidades do segmento a serem medidos. Mas quando se realiza os mesmos procedimentos com imagens de menor resolução, ou câmeras de pior qualidade, ou usando técnicas de calibração e dimensionamento automático, a obtenção das coordenadas 2D dos pontos pode não ser tão precisa.

Para simular uma imprecisão nas medidas, que ocorreria no caso de imagens de baixa qualidade, as etapas de calibração e dimensionamento foram realizadas novamente, com as mesmas imagens, mas acrescentando às coordenadas de todos os pontos um número inteiro de pixels escolhido aleatoriamente no intervalo de -5 a 5. Mais especificamente, dois arquivos de calibração ideais foram criados manualmente, através da captura visual dos pontos de calibração do *grid*. Um arquivo foi criado a partir da Figura 5.1 e outro arquivo foi criado a partir da Figura 5.2. O primeiro arquivo de calibração foi usado para dimensionamento com os pares de imagens 1 e 3 (Figuras 5.1 e 5.4) e o segundo arquivo de calibração foi usado para dimensionamento com os pares de imagens 2 e 4 (Figuras 5.2 e 5.5). Para cada medida documentada nas tabelas da Seção 5.2, uma perturbação foi gerada independentemente das demais medidas e esta mesma perturbação foi aplicada ao arquivo de calibração referente ao par de imagens em questão. Este procedimento foi executado uma vez por linha, ou seja, as medidas documentadas nas Tabelas 5.1 a 5.4 *não são* valores médios, portanto não há desvio-padrão ou dispersão associados a estas medidas.

Para avaliar a vantagem do uso de imagens de alta resolução, todas as medidas mostradas nas Tabelas 5.1 e 5.2 foram repetidas para as mesmas imagens, só que com a resolução reduzida por fator $\sqrt{10}$ ao longo de cada direção da imagem. Os resultados obtidos com estas imagens de baixa resolução são mostrados nas Tabelas 5.3 e 5.4.

5.2 – Resultados

A tabela a seguir apresenta os resultados obtidos utilizando os aplicativos de calibração e dimensionamento apresentados durante o trabalho, para as imagens originais (de alta resolução). Para cada segmento medido é visto na tabela o seu comprimento medido com uma régua, e os resultados utilizando as duas diferentes técnicas de minimização para o dimensionamento (método dos multiplicadores de Lagrange e método dos mínimos quadrados - MQO) com e sem ruído nas coordenadas dos pixels dos pontos do *grid* e das extremidades dos segmentos medidos.

Os pares de imagens das Figuras 5.1, 5.2, 5.4 e 5.5 são chamados em todas as tabelas (5.1 a 5.4) de Par 1, Par 2, Par 3 e Par 4, respectivamente.

Segmento	Medida Exata (cm)	Sem Ruído		Com Ruído		Par
		Lagrange	MQO	Lagrange	MQO	
1	36	35,91	35,91	34,12	34,10	1
2	36	35,69	35,69	32,22	32,22	1
3	5,3	5,37	5,37	4,39	4,32	1
4	5,3	5,27	5,27	4,77	4,84	1
5	36	36,97	36,83	35,63	35,49	2
6	36	34,66	34,67	27,55	27,86	2
7	5,3	5,11	5,12	3,92	4,35	2
8	5,3	5,80	5,71	7,18	5,21	2
9	14,2	14,08	14,08	11,63	12,30	3
10	5,3	5,30	5,30	6,59	6,96	3
11	5,3	5,26	5,25	5,22	3,87	3
12	14,2	13,81	13,81	18,40	18,42	4
13	5,3	5,34	5,36	4,97	5,02	4
14	5,3	5,41	5,37	5,46	5,44	4

Tabela 5.1 – Resultados do dimensionamento com imagens de alta resolução

A Tabela 5.2 apresenta os erros relativos de cada medida. Analisando a tabela, nota-se que a diferença entre os resultados obtidos utilizando o método dos

multiplicadores de Lagrange e o método dos mínimos quadrados foi muito pequena. Os erros das medidas com ruído foram consideravelmente maiores que os erros das medidas sem ruído. É possível observar também que os erros nos Pares 1 e 3, quando a calibração foi realizada com o *grid* mais próximo das câmeras, foram menores que os erros nos Pares 2 e 4. Na Tabela 5.2 estão destacados, em negrito, os maiores erros relativos nas medições para cada método de minimização.

Segmento	Erro sem ruído (%)		Erro com ruído (%)		Par
	Lagrange	MQO	Lagrange	MQO	
1	0,25	0,25	5,22	5,28	1
2	0,86	0,86	10,50	10,50	1
3	1,32	1,32	17,17	18,49	1
4	0,57	0,57	10,00	8,68	1
5	2,69	2,31	1,03	1,42	2
6	3,72	3,69	23,47	22,61	2
7	3,58	3,40	26,04	17,92	2
8	9,43	7,74	35,47	1,70	2
9	0,85	0,85	18,10	13,38	3
10	0,00	0,00	24,34	31,32	3
11	0,75	0,94	1,51	26,98	3
12	2,75	2,75	29,58	29,72	4
13	0,75	1,13	6,23	5,28	4
14	2,08	1,32	3,02	2,64	4

Tabela 5.2 – Erros relativos do dimensionamento com imagens de alta resolução

A Tabela 5.3 exibe os resultados obtidos quando as etapas de calibração e dimensionamento foram repetidas, utilizando as imagens com resolução dez vezes menor.

Segmento	Medida Exata (cm)	Sem Ruído		Com Ruído		Par
		Lagrange	MQO	Lagrange	MQO	
1	36	36,26	34,28	32,68	32,87	1
2	36	36,08	36,13	32,79	32,79	1
3	5,3	5,49	5,44	7,30	7,27	1
4	5,3	5,33	5,35	7,67	6,81	1
5	36	36,07	35,50	29,80	29,69	2
6	36	36,25	36,32	35,50	40,55	2
7	5,3	5,70	5,45	6,17	6,13	2
8	5,3	5,49	5,20	7,07	6,61	2
9	14,2	15,24	15,40	12,60	12,69	3
10	5,3	5,54	5,51	4,62	4,29	3
11	5,3	5,52	5,22	4,43	4,42	3
12	14,2	14,20	14,13	19,83	19,55	4
13	5,3	5,08	5,13	3,57	3,51	4
14	5,3	5,41	5,41	3,24	3,22	4

Tabela 5.3 – Resultados dos dimensionamentos com imagens de baixa resolução

A Tabela 5.4 apresenta os erros relativos nas medidas para as imagens de resolução reduzida. O erro encontrado com imagens de menor resolução foi, em média, maior do que o erro no caso do dimensionamento com imagens de alta resolução. Por exemplo, o método baseado em multiplicadores de Lagrange gera erro relativo médio igual a 15,1% na presença de ruído em imagens de alta resolução, enquanto que este mesmo erro sobe para 22,9% no caso das imagens de baixa resolução. Assim como na Tabela 5.2, os maiores erros relativos estão destacados em cada coluna.

Segmento	Erro sem ruído (%)		Erro com ruído (%)		Par
	Lagrange	MQO	Lagrange	MQO	
1	0,72	4,78	9,22	8,69	1
2	0,22	0,36	8,92	8,92	1
3	3,58	2,64	37,74	37,17	1
4	0,57	0,94	44,72	28,49	1
5	0,19	1,39	17,22	17,53	2
6	0,69	0,89	1,39	12,64	2
7	7,55	2,83	16,42	15,66	2
8	3,58	1,89	33,40	24,72	2
9	6,62	8,45	11,27	10,63	3
10	4,53	3,96	12,83	19,06	3
11	4,15	1,51	16,42	16,60	3
12	0,00	0,49	39,65	37,68	4
13	4,15	3,21	32,64	33,77	4
14	2,08	2,08	38,87	39,25	4

Tabela 5.4 – Erros relativos do dimensionamento com imagens de baixa resolução

Capítulo 6

Conclusões

O aplicativo de visualização funcionou corretamente, permitindo ver as imagens em 3D no monitor entrelaçado de luz polarizada e fazer as capturas das fotografias de alta resolução. Utilizando o programa observou-se que a qualidade das imagens tridimensionais era melhor para objetos mais distantes das câmeras, A explicação para esse fato é que, devido ao tamanho das câmeras, a distância entre as duas lentes é aproximadamente o dobro da distância entre os olhos de uma pessoa. Dessa forma, objetos muito próximos às câmeras geram uma disparidade que não é natural para o sistema visual humano.

O *software* de calibração e dimensionamento também apresentou o funcionamento esperado. Utilizando as imagens de alta resolução obtidas com o aplicativo de visualização foi possível calibrar o par de câmeras e dimensionar segmentos das imagens.

Os resultados encontrados utilizando os dois métodos de dimensionamento foram muito próximos. Teoricamente o método dos multiplicadores de Lagrange apresenta o resultado com o menor erro possível, mas pelo método dos mínimos quadrados a complexidade da solução é menor, então ambos podem ser utilizados dependendo das exigências da aplicação.

Ao adicionar ruído às coordenadas 2D dos pontos de referência e dos pontos que delimitam o segmento medido, percebeu-se claramente que erros maiores ocorreram na presença de ruído. Os erros relativos máximos encontrados nas medidas sem ruído foram 9,4% para as imagens de alta resolução e 8,5% para as imagens de baixa resolução. Os erros relativos máximos encontrados nas mediadas com ruído foram 35,5% para as imagens de alta resolução e 44,7% para as imagens de baixa resolução.

Comparando os resultados obtidos com as imagens de alta e baixa resolução, percebe-se que os erros relativos para o caso de baixa resolução foram superiores na maioria das medidas (em trinta de cinquenta e seis medidas). Quando foi acrescentado ruído, o erro nas medidas obtidas com imagens de baixa

resolução foi menor que 10% em apenas 5 medidas contra 11 no caso das imagens de alta resolução.

Os testes, então, mostraram que a alta resolução das imagens e a boa qualidade das imagens têm importância na precisão da calibração e do dimensionamento.

Para trabalhos futuros é possível fazer uma análise de erros mais detalhada, e utilizar outros métodos de calibração e dimensionamento que permitam maior precisão. Pode-se também desenvolver técnicas para realização de calibração automática, eliminando o trabalho repetitivo da calibração manual.

Bibliografia

- [1] FAUGERAS, O., *Three Dimensional Computer Vision*. Massachusetts, The MIT Press, 1993.
- [2] RAPOSO, A. B., SZENBERG, F., GATTAS, M., CELES, W., *Visão Estereoscópica, Realidade Virtual, Realidade Aumentada e Colaboração*, Tecgraf – Grupo de Tecnologia em Computação Gráfica, Departamento de Informática, PUC-Rio, 2004.
- [3] SILVA, L. C., *Método Robusto para Calibração de Câmeras em Estereofotogrametria*, Tese de Doutorado, COPPE/UFRJ, agosto de 2003.