



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO, DE CIÊNCIAS EXATAS E EDUCAÇÃO
DEPARTAMENTO DE ENG. DE CONTROLE, AUTOMAÇÃO E COMPUTAÇÃO
CURSO DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Bruno Bueno Bronzeri

**Scanner 3D com Técnica de Luz Estruturada para Reconstrução Geométrica
de Objetos**

Blumenau
2026

Bruno Bueno Bronzeri

**Scanner 3D com Técnica de Luz Estruturada para Reconstrução Geométrica
de Objetos**

Trabalho de Conclusão de Curso de Graduação em Engenharia de Controle e Automação do Centro Tecnológico, de Ciências Exatas e Educação da Universidade Federal de Santa Catarina como requisito para a obtenção do título de Engenheiro de Controle e Automação.

Orientador: Prof. Marcos Vinícius Matsuo, Dr.

Blumenau

2026

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.
Dados inseridos pelo próprio autor.

Bueno Bronzeri, Bruno
Scanner 3D com Técnica de Luz Estruturada para
Reconstrução Geométrica de Objetos / Bruno Bueno Bronzeri ;
orientador, Marcos Vinicius Matsuo, 2026.
86 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Campus Blumenau,
Graduação em Engenharia de Controle e Automação, Blumenau,
2026.

Inclui referências.

1. Engenharia de Controle e Automação. 2. Visão
Computacional. 3. Reconstrução Geométrica Tridimensional.
4. Escaneamento 3D. I. Vinicius Matsuo, Marcos. II.
Universidade Federal de Santa Catarina. Graduação em
Engenharia de Controle e Automação. III. Título.

Bruno Bueno Bronzeri

**Scanner 3D com Técnica de Luz Estruturada para Reconstrução Geométrica
de Objetos**

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de “Engenheiro de Controle e Automação” e aprovado em sua forma final pelo Curso de Graduação em Engenharia de Controle e Automação.

Blumenau, 08 de janeiro de 2026.

Banca Examinadora:

Prof. Marcos Vinicius Matsuo, Dr.
Universidade Federal de Santa Catarina

Prof. Alex Fabiano Bueno, Dr.
Universidade Federal de Santa Catarina

Profa. Sheila Santisi Travessa, Dra.
Universidade Federal de Santa Catarina

*Dedico esse trabalho à minha família, meus pais,
meus companheiros de curso e a todos que con-
tribuíram com a minha formação.*

AGRADECIMENTOS

Agradeço, em primeiro lugar, à minha mulher, que esteve ao meu lado desde o início dessa trajetória, oferecendo todo o apoio necessário e contribuindo, direta e indiretamente, para que eu completasse essa etapa da minha vida.

Agradeço aos meus pais, que tiveram papel fundamental em minha jornada acadêmica, com muito apoio e incentivo ao longo da graduação.

Por fim, agradeço aos professores da universidade que contribuíram com seus ensinamentos ao longo da minha formação e, em especial, ao professor que me orientou neste projeto, Dr. Marcos Matsuo, por toda a sua contribuição, disponibilidade e dedicação.

RESUMO

Este trabalho apresenta o desenvolvimento e a validação de um sistema de escaneamento tridimensional baseado em luz estruturada por linha de laser, voltado à reconstrução geométrica de objetos a partir de imagens bidimensionais. O sistema combina aquisição sincronizada e modelagem geométrica para a geração de nuvens de pontos e malhas 3D. A abordagem proposta fundamenta-se na calibração precisa dos elementos do sistema e na reconstrução tridimensional por triangulação. Os resultados experimentais demonstram que o sistema é capaz de reconstruir de forma consistente a geometria de diferentes objetos, evidenciando a influência da resolução angular na densidade da nuvem de pontos e na complexidade da malha gerada. A análise dos parâmetros de calibração destaca a importância da estabilidade dos modelos geométricos para a confiabilidade da reconstrução 3D, enquanto as limitações do sistema e possibilidades de aprimoramento são discutidas.

Palavras-chave: Reconstrução tridimensional; Luz estruturada; Visão computacional; Nuvem de pontos; *Scanner 3D*

ABSTRACT

This work presents the development and validation of a three-dimensional scanning system based on line-structured light, aimed at the geometric reconstruction of objects from two-dimensional images. The proposed system combines synchronized data acquisition and geometric modeling to generate point clouds and 3D meshes. The approach is grounded on accurate calibration of the system components and on triangulation-based three-dimensional reconstruction. Experimental results demonstrate that the system is capable of consistently reconstructing the geometry of different objects, highlighting the influence of angular resolution on point cloud density and mesh complexity. Furthermore, the analysis of calibration parameters emphasizes the importance of geometric model stability for reliable 3D reconstruction, while system limitations and directions for future work are discussed.

Keywords: Three-dimensional reconstruction; Structured light; Computer vision; Point cloud; 3D Scanner.

LISTA DE FIGURAS

Figura 1 – Taxonomia das técnicas de aquisição de informação 3D.	18
Figura 2 – Classificação das técnicas ópticas de aquisição de informação tridimensional.	18
Figura 3 – Triangulação por interseção de linha no plano.	21
Figura 4 – Representação Paramétrica de linhas e raios.	22
Figura 5 – (a) Representação paramétrica do plano. (b) Representação implícita do plano.	22
Figura 6 – Minimização para determinação da interseção.	24
Figura 7 – Representação de câmera estenopeica.	24
Figura 8 – Modelo de Projeção.	26
Figura 9 – Semelhança de Triângulos.	26
Figura 10 – Plano da Imagem com Grade de <i>Pixels</i>	28
Figura 11 – Exemplo de Aruco.	31
Figura 12 – Tabuleiro ChAruCo.	32
Figura 13 – (a) ChArUco visto de frente com chaves identificadas. (b) ChArUco visto em perspectiva com chaves identificadas.	33
Figura 14 – Exemplos de imagens do ChArUco em diferentes poses para calibração de câmera.	34
Figura 15 – (a) Grade sem distorção de lente. (b) Grade com distorção do tipo Barril. (c) Grade com distorção do tipo Almofada.	35
Figura 16 – (a) Vista da câmera com distorção da lente. (b) Vista da câmera sem distorção da lente.	35
Figura 17 – Esquemático da cena do <i>scanner</i>	38
Figura 18 – Câmera IP Hikvision.	39
Figura 19 – Módulo laser 5mW com alinhamento em linha.	40
Figura 20 – Mesa rotativa de ABS com polia dentada acoplada a rolamento.	41
Figura 21 – (a) Microcontrolador ESP32. (b) Driver para motor de passo DRV8825.	42
Figura 22 – Cena do <i>scanner</i> vista da lateral direita da câmera.	43
Figura 23 – Diagrama da disposição do componentes no sistema.	43
Figura 24 – Diagrama do funcionamento do sistema.	45
Figura 25 – Exemplos de imagens do ChArUco para calibração.	46
Figura 26 – Exemplos de retornos da identificação de ChArUcos para calibração.	48
Figura 27 – Exemplos de imagens do ArUco sobre a mesa rotativa.	50
Figura 28 – Representação 3D da estimação da circunferência que descreve a mesa e do eixo de rotação.	51
Figura 29 – Exemplos de imagens do ChArUco para estimação do plano do laser.	52

Figura 30 – Representação do plano estimado a partir do sistema de coordenadas da câmera.	54
Figura 31 – (a) Imagem para reconstrução sem processamento de região. (b) Imagem para reconstrução com processamento de região.	56
Figura 32 – Definição da região de interesse.	56
Figura 33 – (a) Nuvem de pontos de objeto cúbico (200 imagens). (b) Nuvem de pontos de objeto cúbico (1600 imagens).	61
Figura 34 – (a) Malha 3D de objeto cúbico (200 imagens). (b) Malha 3D de objeto cúbico (1600 imagens).	64
Figura 35 – Visualização 3D da calibração completa do sistema (câmera, plano da mesa rotativa e eixo de rotação).	67
Figura 36 – Visualização 3D da calibração completa do sistema (câmera, plano da mesa rotativa e plano do laser).	68
Figura 37 – Objetos utilizados para validação do <i>scanner</i>	69
Figura 38 – Cubo elaborado em <i>SolidWorks</i> e impresso em 3D.	70
Figura 39 – (a) Nuvem de pontos do Cubo. (b) Reconstrução 3D do Cubo.	70
Figura 40 – Malha do cubo com cotas de validação de dimensões.	71
Figura 41 – Pirâmide elaborada em <i>SolidWorks</i> e impressa em 3D.	73
Figura 42 – (a) Nuvem de pontos da Pirâmide. (b) Reconstrução 3D da Pirâmide.	73
Figura 43 – Malha da pirâmide com cotas de validação de dimensões.	74
Figura 44 – Ampulheta elaborada em <i>SolidWorks</i> e impressa em 3D.	76
Figura 45 – (a) Nuvem de pontos da Ampulheta. (b) Reconstrução 3D da Ampulheta.	76
Figura 46 – Abajur esférico.	77
Figura 47 – (a) Nuvem de pontos do Abajur. (b) Reconstrução 3D do Abajur.	78
Figura 48 – Cantil.	79
Figura 49 – (a) Nuvem de pontos do Cantil. (b) Reconstrução 3D do Cantil.	79

LISTA DE QUADROS

Pré-processamento e extração da máscara binária do laser	57
Extração subpixel por centróide	58
Triangulação por interseção entre raio de visão e plano do laser	59

LISTA DE TABELAS

Tabela 1 – Influência do número de imagens na calibração da câmera.	66
Tabela 2 – Comparação entre dimensões reais e reconstruídas do cubo em milímetros	71
Tabela 3 – Erros absolutos (mm) em cinco reconstruções independentes do cubo. .	71
Tabela 4 – Estatísticas descritivas dos erros absolutos por eixo para o cubo.	72
Tabela 5 – Objeto: cubo. Influência do número de imagens na geração da nuvem de pontos e malha 3D.	72
Tabela 6 – Comparação entre dimensões reais e reconstruídas da pirâmide em milímetros	74
Tabela 7 – Erros absolutos (mm) em cinco reconstruções independentes da pirâmide.	74
Tabela 8 – Estatísticas descritivas dos erros absolutos para a pirâmide.	75
Tabela 9 – Objeto: pirâmide. Influência do número de imagens na geração da nuvem de pontos e malha 3D.	75
Tabela 10 – Objeto: ampulheta. Influência do número de imagens na geração da nuvem de pontos e malha 3D.	75
Tabela 11 – Objeto: abajur. Influência do número de imagens na geração da nuvem de pontos e malha 3D.	77
Tabela 12 – Objeto: cantil. Influência do número de imagens na geração da nuvem de pontos e malha 3D.	78

SUMÁRIO

1	INTRODUÇÃO	14
1.1	OBJETIVO GERAL	15
1.2	OBJETIVOS ESPECÍFICOS	15
1.3	ORGANIZAÇÃO DO TRABALHO	16
2	FUNDAMENTAÇÃO TEÓRICA	17
2.1	DIGITALIZAÇÃO 3D E APLICAÇÕES	17
2.1.1	Aplicações Industriais	19
2.1.2	Aplicações Metrológicas	20
2.2	PRINCÍPIOS DA LUZ ESTRUTURADA E TRIANGULAÇÃO	20
2.2.1	Reconstrução por Triangulação	20
<i>2.2.1.1</i>	<i>Interseção de Linha no Plano e Entre Linhas</i>	<i>23</i>
2.3	GEOMETRIA DA CÂMERA	24
2.3.1	Câmera Estenopeica (<i>Pinhole</i>)	24
2.3.2	Formação da Imagem	25
2.3.3	Modelo da Câmera	26
2.4	CALIBRAÇÃO DE CÂMERA	29
2.4.1	Tabuleiro de Calibração (ChArUco)	31
2.4.2	Resultados da Calibração	33
<i>2.4.2.1</i>	<i>Lentes e Matriz de Distorção</i>	<i>34</i>
3	DESENVOLVIMENTO	37
3.1	CONCEPÇÃO DO SISTEMA	37
3.2	ARQUITETURA DE <i>HARDWARE</i>	38
3.2.1	Componentes Utilizados	39
<i>3.2.1.1</i>	<i>Câmera IP</i>	<i>39</i>
<i>3.2.1.2</i>	<i>Emissor de Luz Estruturada</i>	<i>40</i>
<i>3.2.1.3</i>	<i>Mesa Rotativa e Motor de Passo</i>	<i>40</i>
<i>3.2.1.4</i>	<i>Microcontrolador (ESP32) e Driver DRV8825</i>	<i>41</i>
3.2.2	Integração dos Componentes	42
3.3	ARQUITETURA DE <i>SOFTWARE</i>	44
3.4	CALIBRAÇÃO DA CÂMERA	45
<i>3.4.0.1</i>	<i>Aquisição das Imagens de Calibração</i>	<i>46</i>
3.4.1	Detecção do Padrão ChArUco	46
3.4.2	Estimação dos Parâmetros Intrínsecos da Câmera	47
3.5	ESTIMAÇÃO DO SISTEMA DE REFERÊNCIA DA MESA ROTATIVA	49
3.5.1	Método de Estimação do Plano e Eixo Z da Mesa	49
3.5.2	Estimação do Plano do Laser	51
3.6	PROCESSO DE AQUISIÇÃO DE DADOS DE ESCANEAMENTO	54

3.7	RECONSTRUÇÃO DA NUVEM DE PONTOS	55
3.7.1	Pré-processamento e Extração da Linha do Laser	57
3.7.2	Extração Subpixel por Centroide	57
3.7.3	Suavização do Perfil ao Longo da Imagem	58
3.7.4	Triangulação por Interseção Raio-Plano	58
3.7.5	Transformações de Referencial e Composição Global da Nuvem	59
3.7.5.1	<i>Rotação Incremental e Empilhamento das Seções</i>	60
3.7.5.2	<i>Exportação, Visualização e Persistência da Nuvem de Pontos</i>	60
3.8	PÓS-PROCESSAMENTO E GERAÇÃO DE MALHAS	62
3.8.1	Pós-processamento da Nuvem de Pontos e Geração da Malha	62
3.8.2	Exportação do Modelo Final	63
4	RESULTADOS E DISCUSSÕES	65
4.1	VALIDAÇÃO DA CALIBRAÇÃO GEOMÉTRICA	65
4.1.1	Resultados Numéricos da Calibração da Câmera	65
4.2	VISUALIZAÇÃO DA CALIBRAÇÃO COMPLETA DO SISTEMA	67
4.3	RESULTADOS DA RECONSTRUÇÃO TRIDIMENSIONAL	68
4.3.1	Objetos Avaliados e Condições de Aquisição	68
4.3.2	Análise e Comparação Final	69
4.3.2.1	<i>Objeto I - Cubo</i>	69
4.3.2.2	<i>Objeto II - Pirâmide</i>	72
4.3.2.3	<i>Objeto III - Ampulheta</i>	75
4.3.2.4	<i>Objeto IV - Abajur</i>	76
4.3.2.5	<i>Objeto V - Cantil</i>	78
4.4	DISCUSSÃO, SUGESTÕES E MELHORIAS	79
5	CONCLUSÃO	81
	REFERÊNCIAS	83
	APÊNDICE A – Repositório do Código-Fonte	86

1 INTRODUÇÃO

Escaneamento 3D (*3D scanning*) é uma técnica cada vez mais difundida na atualidade, devido à sua praticidade, confiabilidade e precisão na obtenção de formas e dimensões de objetos. Essa tecnologia é essencial para aplicações industriais, científicas e comerciais. Entre suas principais áreas de uso, destacam-se: impressão 3D, controle de qualidade, medicina, *design* industrial, engenharia reversa, jogos e animações, além da preservação de patrimônio histórico.

Dentre as diversas técnicas existentes de digitalização 3D, a *luz estruturada* (*structured light*) é uma das mais utilizadas, principalmente por permitir alta precisão com potencial de baixo custo, quando comparada a soluções baseadas em *laser* ou sensores de tempo de voo. Essa técnica consiste na projeção de padrões ópticos como faixas codificadas (*gray code*, *phase shifting*, etc.) sobre a superfície do objeto. As distorções observadas, capturadas por uma ou mais câmeras, permitem reconstruir a geometria da superfície por meio de triangulação óptica.

Diversas empresas de renome oferecem soluções comerciais baseadas em luz estruturada. Exemplos incluem a Creafom, Cognex, Artec 3D, Shining 3D e ainda a HP. Embora essas soluções apresentem altíssima qualidade e precisão, seus custos ainda são elevados, o que limita o acesso em contextos de pesquisa, educação e *hobby*. Como consequência, tem crescido o interesse por projetos alternativos e de baixo custo que buscam replicar os princípios dessas soluções comerciais. Esses projetos geralmente utilizam componentes acessíveis e modulares, como *Arduino*, *ESP32*, *Raspberry Pi*, câmeras IP ou webcams, projetores convencionais e motores de passo.

O mercado global de digitalização 3D tem apresentado crescimento acelerado. De acordo com relatórios da *Markets and Markets*, o setor de *3D scanning* movimentou mais de 5 bilhões de dólares em 2023, com projeções de crescimento contínuo impulsionado por avanços na automação industrial, manufatura digital e aplicações em saúde. No setor educacional e *maker*, a democratização de componentes e *softwares* e bibliotecas *open-source*, como o *OpenCV* (com o módulo `structured_light`) e o *Blender/Meshlab*, tem favorecido o desenvolvimento de soluções funcionais e acessíveis, incentivando a inovação e a disseminação do conhecimento (Han et al., 2025).

A construção de sistemas de digitalização tridimensional apresenta desafios significativos relacionados à precisão e ao custo dos componentes utilizados. Câmeras de baixo custo, embora amplamente disponíveis, geralmente oferecem menor resolução e sensibilidade óptica, além de apresentarem lentes com maior distorção geométrica, o que exige calibrações mais rigorosas e suscetíveis a ruídos. Da mesma forma, projetores e emissores de luz estruturada podem introduzir imprecisões caso não apresentem uniformidade e estabilidade adequadas. Diante desses fatores, o desenvolvimento de um *scanner 3D* de baixo custo, mas com resultados satisfatórios em termos de fidelidade geométrica, torna-se

um desafio relevante. Nesse contexto, o presente trabalho busca estudar, aplicar e aprimorar métodos de calibração, reconstrução e processamento que permitam alcançar um equilíbrio entre custo e desempenho.

1.1 OBJETIVO GERAL

O objetivo deste projeto é desenvolver e validar um *scanner 3D* baseado em técnicas de visão computacional, utilizando uma câmera, uma bandeja rotativa (acoplada a um motor de passo) e um emissor de luz estruturada para a geração de nuvens de pontos tridimensionais de objetos físicos.

1.2 OBJETIVOS ESPECÍFICOS

Para o desenvolvimento deste projeto, foi necessário seguir uma série de etapas que envolvem desde a montagem da estrutura física até a implementação dos algoritmos de processamento de imagem e reconstrução tridimensional. Assim, o trabalho tem como finalidade não apenas construir o sistema físico de escaneamento 3D baseado em luz estruturada, mas também abordar, compreender e aplicar os conceitos teóricos e práticos indispensáveis ao seu funcionamento.

De forma mais detalhada, os objetivos específicos incluem:

- Projetar e montar a estrutura física do *scanner 3D*, composta por uma câmera, um emissor de luz estruturada e um sistema de rotação controlado por motor de passo;
- Realizar a calibração da câmera para correção das distorções ópticas e determinação dos parâmetros intrínsecos e extrínsecos do sistema de captura;
- Determinar o eixo de rotação da bandeja, correspondente ao eixo Z , e garantir seu alinhamento com o sistema de coordenadas da câmera;
- Estimar e modelar matematicamente o plano de incidência da luz estruturada, a partir da calibração geométrica entre câmera e emissor;
- Desenvolver o algoritmo de sincronização entre a rotação da bandeja e a aquisição das imagens, assegurando uma captura ordenada e precisa;
- Implementar o processamento das imagens adquiridas para a detecção da linha de luz estruturada e reconstrução da nuvem de pontos tridimensional do objeto escaneado;
- Validar o desempenho do sistema construído por meio de testes experimentais com objetos de diferentes formas e superfícies, avaliando a precisão e a consistência dos resultados obtidos.

1.3 ORGANIZAÇÃO DO TRABALHO

Este trabalho está estruturado para conduzir a leitura desde os fundamentos teóricos necessários até a implementação completa do sistema de digitalização tridimensional proposto. No Capítulo 2 apresentam-se os conceitos fundamentais de visão computacional, geometria projetiva, modelos de câmera, técnicas de calibração e princípios de reconstrução 3D por luz estruturada. Esses elementos constituem a base teórica indispensável para o entendimento das metodologias empregadas no desenvolvimento do *Scanner 3D*.

O Capítulo 3 descreve o desenvolvimento do sistema, abrangendo desde sua concepção física até as etapas de processamento computacional. São detalhadas a montagem do dispositivo, os componentes utilizados, os procedimentos de calibração da câmera, a estimação do plano do laser e o método de triangulação empregado para a reconstrução da geometria do objeto. Também são apresentadas as etapas finais de pós-processamento, incluindo a geração da nuvem de pontos e a construção de malhas triangulares.

No Capítulo 4, são expostos e discutidos os resultados experimentais obtidos. Primeiramente, realiza-se a validação do sistema por meio do escaneamento de objetos com dimensões conhecidas, avaliando sua acurácia. Em seguida, apresentam-se digitalizações de objetos com geometrias diversas, com o objetivo de demonstrar a capacidade do sistema em preservar detalhes morfológicos e representar adequadamente superfícies reais.

Por fim, o Capítulo 5 apresenta as conclusões gerais do trabalho, destacando as contribuições obtidas, as limitações identificadas durante o desenvolvimento e sugestões de melhorias e trabalhos futuros, tanto no *hardware* quanto nas etapas de processamento e reconstrução tridimensional.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo tem como finalidade apresentar a fundamentação teórica necessária para o entendimento e desenvolvimento deste trabalho, abordando tópicos como a história do escaneamento 3D e algumas técnicas relevantes existentes, bem como fundamentos de câmeras, padrões de luz, métodos de calibração e discussão acerca dos parâmetros geométricos que precisam ser determinados para realização da triangulação de informação e geração de nuvens de pontos.

2.1 DIGITALIZAÇÃO 3D E APLICAÇÕES

A digitalização tridimensional constitui uma das bases tecnológicas mais relevantes para a reprodução geométrica de objetos físicos no espaço digital. Seu desenvolvimento está diretamente associado à evolução dos sensores ópticos e dos métodos de reconstrução geométrica, que, ao longo das últimas décadas, tornaram-se ferramentas fundamentais em áreas como engenharia reversa, controle dimensional, design industrial e preservação de patrimônio histórico-cultural.

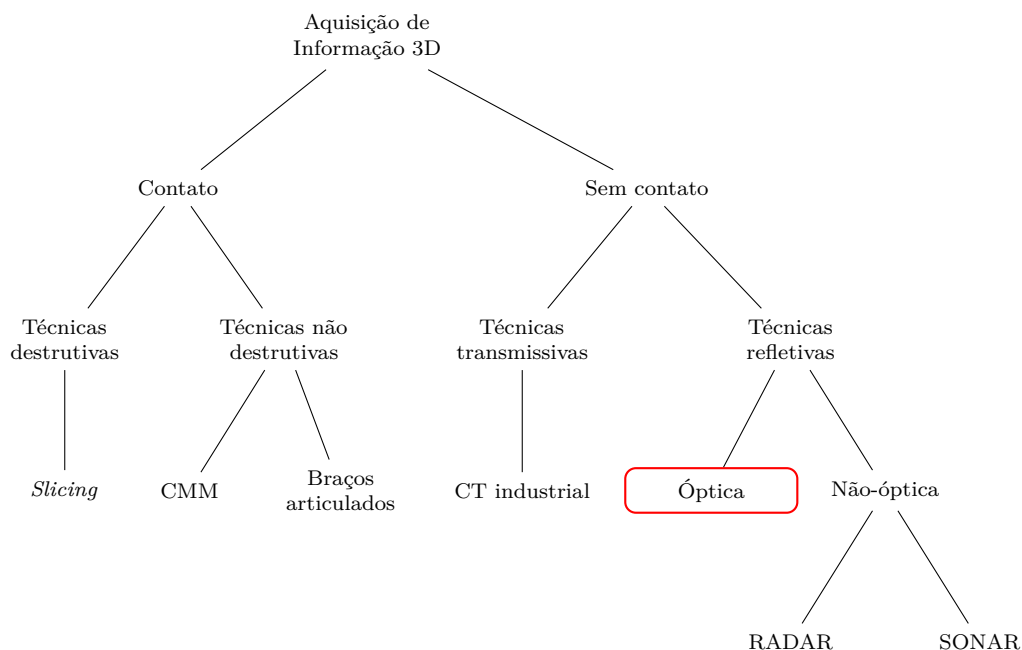
O desenvolvimento dos sensores de medição de distância teve início nas décadas de 1970 e 1980, com os primeiros dispositivos de varredura baseados em técnicas de contato mecânico. Esses sistemas, embora precisos, apresentavam limitações significativas de velocidade, custo e aplicabilidade, especialmente em superfícies complexas. Com o avanço da eletrônica e da óptica nas décadas seguintes, surgiram os primeiros dispositivos capazes de realizar medições tridimensionais sem contato, inaugurando uma nova geração de métodos de digitalização 3D (Blais, 2004).

As técnicas de digitalização tridimensional podem ser classificadas, de modo geral, em métodos de contato e métodos sem contato. Os métodos de contato incluem sistemas como máquinas de medição por coordenadas (*Coordinate Measuring Machines – CMM*) e braços articulados, nos quais um apalpador físico realiza a medição ponto a ponto sobre a superfície do objeto. Já as técnicas sem contato subdividem-se em transmissivas e refletivas, podendo ainda empregar princípios ópticos ou não ópticos (como ultrassom e radar). No caso das técnicas ópticas, estas se destacam pela elevada precisão, velocidade de aquisição e pela capacidade de capturar superfícies complexas sem deformação física do objeto (Sansoni, 2009).

Dentro do domínio das técnicas ópticas, é possível identificar duas abordagens principais: as *passivas* e as *ativas*. As técnicas passivas, como a fotogrametria e a reconstrução estéreo, dependem exclusivamente da luz ambiente refletida pelo objeto, sendo amplamente utilizadas em aplicações de grande escala. Por outro lado, as técnicas ativas fazem uso de uma fonte de iluminação controlada — como um feixe de laser ou um padrão projetado — para determinar as coordenadas tridimensionais por meio de princípios de triangulação, interferometria ou luz estruturada (Geng, 2011; Liu et al., 2019).

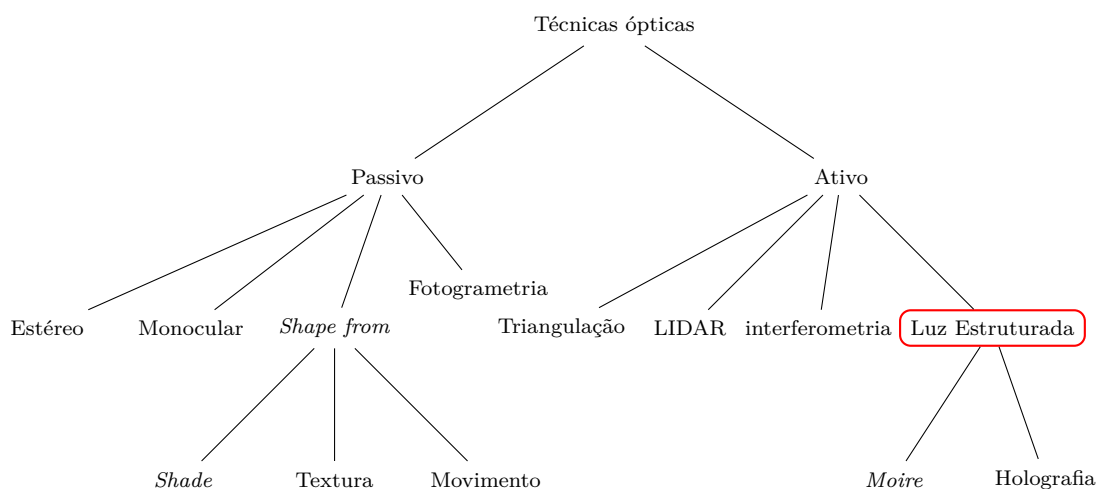
A Figura 1 apresenta um esquema geral das principais categorias de técnicas de digitalização tridimensional. Esse tipo de representação permite compreender como os diferentes métodos se organizam de acordo com o tipo de contato, natureza da luz utilizada e princípio de medição adotado. O diagrama também serve de base conceitual para classificar a metodologia deste trabalho, que emprega uma técnica ativa e óptica, baseada em projeção de luz estruturada, como explicita a Figura 2.

Figura 1 – Taxonomia das técnicas de aquisição de informação 3D.



Fonte: Adaptado de (Rocchini et al., 2001).

Figura 2 – Classificação das técnicas ópticas de aquisição de informação tridimensional.



Fonte: Adaptado pelo de (Rocchini et al., 2001).

A consolidação dessas técnicas, conforme destacado anteriormente, foi impulsionada tanto pelo avanço dos sensores de imagem (como câmeras CMOS e CCD de alta resolução)

quanto pela maior capacidade de processamento computacional, que viabilizou algoritmos de calibração, reconstrução e fusão de nuvens de pontos cada vez mais precisos. Assim, a digitalização 3D tornou-se uma ferramenta essencial na interface entre o mundo físico e o digital, permitindo desde aplicações industriais de metrologia até a criação de modelos virtuais para simulação e manufatura aditiva.

2.1.1 Aplicações Industriais

A digitalização 3D representa uma ferramenta estratégica em ambientes industriais que exigem inspeção não destrutiva e medição de alta resolução. Entre as principais aplicações destacam-se: a engenharia reversa, o controle dimensional de peças, a integração com processos de manufatura aditiva e a automação da inspeção óptica em linhas de produção (Sansoni, 2009).

Na engenharia reversa, a digitalização tridimensional é utilizada para reconstruir modelos digitais de peças existentes, possibilitando sua reprodução, modificação ou integração em novos projetos de produto (Liu et al., 2019). Em muitos casos, esse processo substitui ou complementa desenhos técnicos antigos ou inexistentes, permitindo a reengenharia de componentes cuja geometria complexa seria de difícil modelagem manual. Essa aplicação é especialmente relevante em setores como o automotivo, aeroespacial e metalmeccânico.

Além disso, a digitalização 3D vem sendo integrada a processos de manufatura aditiva (*Additive Manufacturing*), em que os modelos digitalizados servem como base para impressão 3D de protótipos e peças funcionais (Geng, 2011). Essa integração fecha o ciclo entre aquisição, modelagem e reprodução física, constituindo o que se denomina *Digital Twin* ou gêmeo digital — um paradigma de representação virtual de sistemas físicos utilizado para simulação, otimização e manutenção preditiva.

No contexto da automação industrial, sensores de visão 3D também são empregados em robôs de inspeção, manipulação e montagem, permitindo o reconhecimento espacial de objetos e o posicionamento preciso em ambientes de produção. Conforme destaca (Blais, 2004), a incorporação de sensores tridimensionais em linhas automatizadas permite substituir medições manuais por processos ópticos sem contato, reduzindo erros humanos e aumentando a produtividade.

De modo geral, a digitalização tridimensional deixou de ser uma tecnologia restrita a laboratórios de metrologia para se tornar uma ferramenta amplamente adotada na manufatura inteligente. Sua integração com sistemas de controle, robótica e inteligência artificial tem consolidado o conceito de Indústria 4.0, em que o fluxo contínuo de dados tridimensionais serve de base para decisões automatizadas e sistemas de produção autônomos.

2.1.2 Aplicações Metrológicas

A digitalização tridimensional desempenha papel central em aplicações metrológicas e de documentação de patrimônio histórico, graças à sua capacidade de capturar geometrias complexas com elevada precisão e sem contato físico. No campo da metrologia industrial, os sistemas ópticos baseados em luz estruturada e triangulação a laser têm sido amplamente empregados na inspeção dimensional de peças e componentes, permitindo medições rápidas e automatizadas com precisão na ordem de micrômetros (Sansoni, 2009; Blais, 2004; Remondino et al., 2008). Tais sistemas substituem ou complementam equipamentos convencionais, como máquinas de medição por coordenadas (*CMM*), possibilitando a comparação direta entre o modelo digital obtido e o projeto de referência.

Além do contexto industrial, a digitalização tridimensional tem se mostrado uma ferramenta valiosa na preservação e restauração de bens culturais e históricos. Destaca-se seu uso em projetos de documentação de obras de arte, esculturas e edificações, nos quais os modelos 3D obtidos permitem o registro detalhado de formas e texturas sem risco de dano ao patrimônio (Sansoni, 2009). Essa abordagem possibilita tanto a análise e conservação física das peças quanto a criação de réplicas digitais para estudo, exposição virtual ou reconstituição de elementos degradados.

Dessa forma, a digitalização 3D consolida-se como uma tecnologia de dupla relevância: ao mesmo tempo em que contribui para o avanço da metrologia industrial, também desempenha papel fundamental na preservação do patrimônio histórico e cultural.

2.2 PRINCÍPIOS DA LUZ ESTRUTURADA E TRIANGULAÇÃO

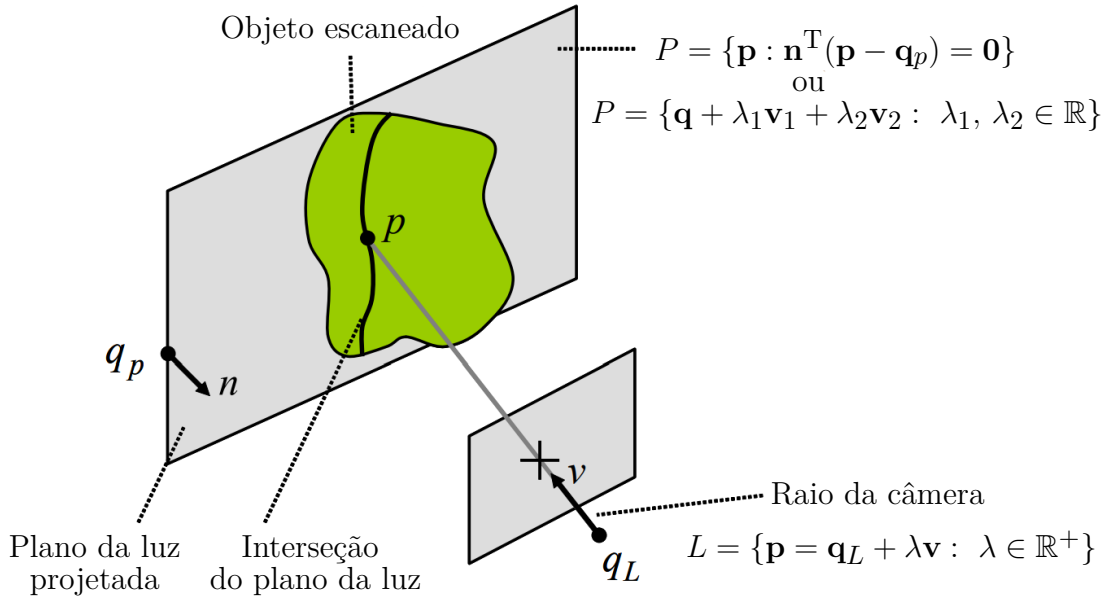
Luz estruturada é uma técnica ativa de digitalização 3D na qual um emissor (projetor ou laser) gera padrões luminosos conhecidos sobre a superfície do objeto, permitindo resolver o problema de correspondência e, por triangulação, estimar coordenadas tridimensionais (Geng, 2011; Rocchini et al., 2001). Diferentemente de abordagens passivas (por exemplo, estéreo ou fotogrametria), a fonte de iluminação controlada fornece identidade espacial aos pontos observados, aumentando robustez e velocidade de aquisição (Sansoni, 2009; Liu et al., 2019).

2.2.1 Reconstrução por Triangulação

A reconstrução por triangulação é uma técnica de determinação de pontos no espaço tridimensional a partir de uma imagem bidimensional. Nesse âmbito, a luz estruturada, associada a um plano geométrico conhecido, ao ser projetada sobre o objeto e refletida pela sua superfície, gera um padrão de deformação diretamente relacionado à geometria do objeto (Taubin, 2014). Para que a triangulação seja válida, é necessário que tanto o plano da luz quanto os raios projetivos da câmera estejam corretamente calibrados, de modo que os pontos pertencentes à linha de deformação possam ser obtidos a partir

da interseção entre o plano da luz e os raios da câmera, permitindo a estimação de suas coordenadas no espaço tridimensional, conforme ilustrado na Figura 3. Entretanto, ressalta-se a importância da consistência geométrica do modelo de calibração, sendo necessária a análise da estabilidade dos parâmetros intrínsecos para assegurar a confiabilidade da reconstrução tridimensional por triangulação (Chen et al., 2025; Ricolfe-Viala, 2012).

Figura 3 – Triangulação por interseção de linha no plano.



Fonte: Adaptado de (Taubin, 2014).

Neste momento é importante abordar a representação paramétrica de linhas, raios e planos. Dessa forma uma linha L pode ser descrita a partir da definição de um ponto q no espaço, como origem e um vetor \mathbf{v} que determina a direção, conforme Figura 4. Qualquer outro ponto \mathbf{p} na linha L pode ser descrito como a adição do ponto \mathbf{q} ao produto do vetor v pelo escalar λ , ou seja, $\mathbf{p} = \mathbf{q} + \lambda \cdot v$. Matematicamente para linhas, considera-se $\lambda \in \mathbb{R}$ (pode assumir valores positivos ou negativos). Assim, uma linha L pode ser descrita como

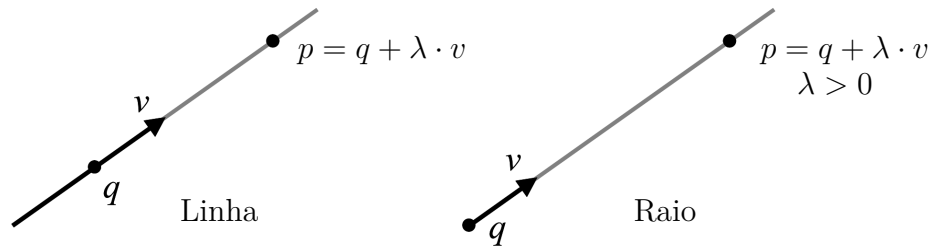
$$L = \{\mathbf{p} = \mathbf{q} + \lambda \cdot v : \lambda \in \mathbb{R}\} \quad (1)$$

Já para raios que são descritos como a metade de uma linha, considera-se $\lambda \in \mathbb{R}^+$, ou seja,

$$L = \{\mathbf{p} = \mathbf{q} + \lambda \cdot v : \lambda \in \mathbb{R}^+\} \quad (2)$$

Com a equação paramétrica do raio, Equação (2), é possível encontrar a representação paramétrica e implícita do plano P . Basta apenas ser especificado um ponto \mathbf{q} e dois vetores linearmente independentes \mathbf{v}_1 e \mathbf{v}_2 , conforme Figura 5(a). Logo qualquer ponto \mathbf{p}

Figura 4 – Representação Paramétrica de linhas e raios.



Fonte: Adaptado de (Taubin, 2014).

no plano P pode ser representado pelo resultado da adição do ponto \mathbf{q} com os produtos $\lambda_1 \cdot \mathbf{v}_1$ e $\lambda_2 \cdot \mathbf{v}_2$. Assim, o plano P pode ser descrito como

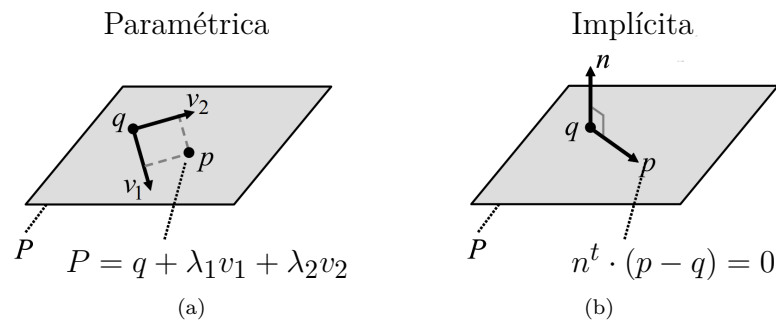
$$P = \{\mathbf{q} + \lambda_1 \cdot \mathbf{v}_1 + \lambda_2 \cdot \mathbf{v}_2 : \lambda_1, \lambda_2 \in \mathbb{R}\}. \quad (3)$$

Por fim, um plano P também pode ser descrito na forma implícita, Figura 5(b), através do ponto \mathbf{q} e do vetor normal \mathbf{n} , resultante do produto vetorial de \mathbf{v}_1 e \mathbf{v}_2 ($\mathbf{n} = \mathbf{v}_1 \times \mathbf{v}_2$). Tendo em vista que um ponto \mathbf{p} pertence ao plano P se e somente se os vetores $\mathbf{p} - \mathbf{q}$ e \mathbf{n} são ortogonais, em que o plano pode ser descrito como

$$P = \{p : \mathbf{n}^t \cdot (p - q) = 0\}, \quad (4)$$

onde $(\cdot)^t$ denota a operação de transposto.

Figura 5 – (a) Representação paramétrica do plano. (b) Representação implícita do plano.



Fonte: Adpatado de (Taubin, 2014).

Neste trabalho, a representação paramétrica da linha e implícita do plano é adotada por conveniência geométrica e clareza física no processo de triangulação. A linha, descrita de forma paramétrica, expressa o conjunto de pontos gerados a partir do centro óptico da câmera em uma direção específica, permitindo calcular diretamente a posição espacial de um *pixel* observado. Já o plano, descrito em forma implícita, define o conjunto de pontos

que satisfazem a condição geométrica do feixe de luz projetado, sendo mais adequada para representar superfícies fixas no espaço. Essa combinação é especialmente útil na triangulação, pois a interseção entre uma linha paramétrica e um plano implícito resulta em um ponto único no espaço tridimensional, correspondendo à posição real do ponto iluminado pela luz estruturada.

2.2.1.1 Interseção de Linha no Plano e Entre Linhas

É notável que não haverá interseção entre uma linha e um plano no caso desses elementos serem paralelos. Esse caso ocorre no caso dos vetores \mathbf{v} e \mathbf{n} serem ortogonais ($\mathbf{n}^t \cdot \mathbf{v} = 0$). Outro ponto importante é que os vetores \mathbf{v} e \mathbf{n} também serão sempre ortogonais quando a linha L está contida no plano P . Por outro lado se os vetores \mathbf{v} e \mathbf{n} não são ortogonais ($\mathbf{n}^t \cdot \mathbf{v} \neq 0$), há um indicativo que a linha e o plano contém um ponto \mathbf{p} em comum de interseção. Como esse ponto pertence a linha, pode ser escrito como $\mathbf{p} = \mathbf{q}_L + \lambda \cdot \mathbf{v}$, com um valor de λ a ser determinado. Uma vez que o ponto pertence ao plano também, é possível escrever a seguinte equação que satisfaz essa condição:

$$\mathbf{n}^t \cdot (\mathbf{p} - \mathbf{q}_p) = \mathbf{n}^t \cdot (\lambda \cdot \mathbf{v} + \mathbf{q}_L - \mathbf{q}_p) = 0$$

podendo ser reescrita de forma a determinar o valor de λ , como

$$\lambda = \frac{\mathbf{n}^t \cdot (\mathbf{q}_p - \mathbf{q}_L)}{\mathbf{n}^t \cdot \mathbf{v}}. \quad (5)$$

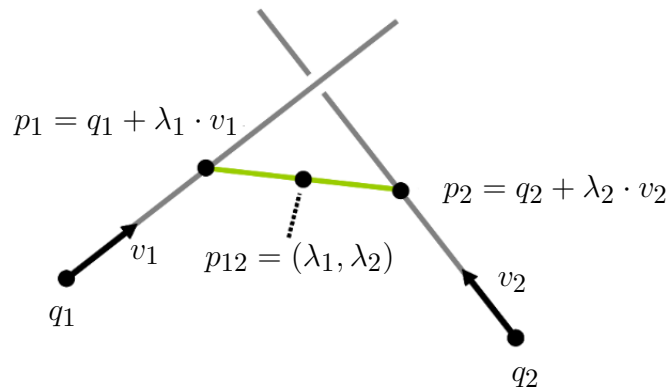
Por fim, uma abordagem alternativa para determinar o ponto de interseção entre duas linhas consiste em minimizar o vetor $\mathbf{p}_{12}(\lambda_1, \lambda_2)$, definido como a diferença entre os pontos parametrizados de cada linha,

$$\mathbf{p}_{12}(\lambda_1, \lambda_2) = (\mathbf{q}_1 + \lambda_1 \mathbf{v}_1) - (\mathbf{q}_2 + \lambda_2 \mathbf{v}_2),$$

de modo que, no ponto ótimo, esse vetor se anule. Essa condição é satisfeita quando existe um ponto de cruzamento entre as linhas, assumindo que elas não são paralelas, conforme é possível ver na Figura 6.

Além da formulação geométrica da interseção, é importante considerar a propagação das incertezas associadas à calibração da câmera e à estimação do plano do laser no erro final do ponto tridimensional reconstruído. Pequenas perturbações nos parâmetros intrínsecos e extrínsecos da câmera, assim como na normal e na posição do plano da luz, afetam diretamente a direção dos raios projetivos e a posição do plano de interseção, resultando em desvios na estimativa do ponto 3D. Em sistemas de luz estruturada por linha, o erro de reprojeção, expresso em *pixels*, pode ser interpretado como uma incerteza angular nos raios da câmera que, ao ser propagada pela triangulação, gera erros espaciais dependentes da distância ao sensor e da geometria entre o plano do laser e o eixo óptico. Estudos experimentais indicam que, mesmo para baixos valores de erro de reprojeção, a

Figura 6 – Minimização para determinação da interseção.



Fonte: Adaptado de (Taubin, 2014).

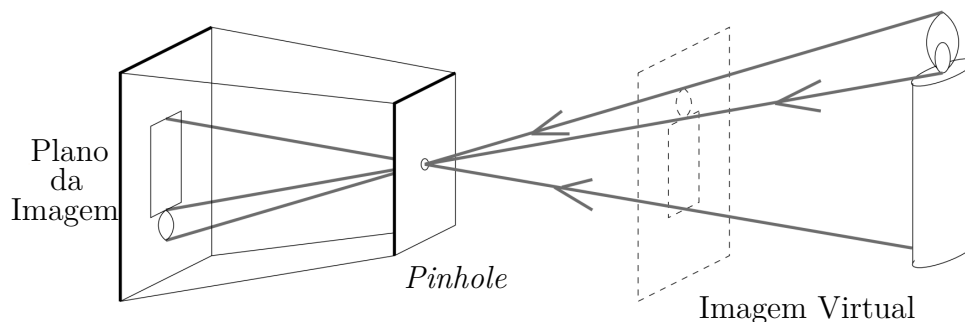
instabilidade dos parâmetros de calibração pode resultar em erros tridimensionais significativos, reforçando a necessidade de calibrações geometricamente consistentes e estáveis (Shao; Dong; Madessa, 2019).

2.3 GEOMETRIA DA CÂMERA

2.3.1 Câmera Estenopeica (*Pinhole*)

O processo de formação de imagem em uma câmera pode ser descrito matematicamente a partir do modelo *pinhole*, também conhecido como *modelo de câmera estenopeica*, conforme Figura 7. Esse modelo constitui a base da maior parte dos algoritmos de visão computacional e reconstrução tridimensional, por fornecer uma representação geométrica simples e precisa da projeção de pontos do espaço tridimensional sobre o plano da imagem. A partir dele, é possível compreender e modelar a relação entre o espaço físico e o espaço da imagem (Hartley; Zisserman, 2003).

Figura 7 – Representação de câmera estenopeica.



Fonte: Adaptado de (Forsyth; Ponce, 2011).

O modelo *pinhole* considera que os raios de luz provenientes de uma fonte de luz e refletidos por um objeto convergem em um único ponto — o centro óptico da câmera — e são projetados linearmente sobre o plano da imagem (localizada na parede oposta da câmera) (Hartley; Zisserman, 2003). Assim, a imagem formada corresponde a uma transformação projetiva (ou perspectiva) do espaço tridimensional para o plano bidimensional da imagem. Essa relação pode ser expressa de forma compacta pela equação:

$$\mathbf{x} \sim \mathbf{K}[\mathbf{R} \mid \mathbf{t}] \mathbf{X}, \quad (6)$$

onde $\mathbf{X} = [x_c \ y_c \ z_c \ 1]^\top$ representa as coordenadas homogêneas de um ponto no espaço, $\mathbf{x} = [\tilde{u} \ \tilde{v} \ \tilde{w}]^\top$ representa o ponto correspondente no plano da imagem, \mathbf{R} e \mathbf{t} são, respectivamente, a matriz de rotação e o vetor de translação que relacionam o sistema de coordenadas do mundo ao sistema da câmera, e \mathbf{K} é a matriz de parâmetros intrínsecos que descreve as propriedades internas da câmera, relacionadas com o sensor e a lente. Contudo esse modelo ainda não é completo o suficiente.

Em termos físicos, o modelo *pinhole* idealiza o processo de captura desconsiderando as imperfeições ópticas da lente (isto é, considera a “lente ideal”). Na prática, porém, as lentes introduzem distorções — principalmente radial e tangencial — que alteram a posição aparente dos *pixels* projetados e será abordado à frente. Essas distorções devem ser modeladas e compensadas por meio do processo de calibração, etapa fundamental para aplicações que exigem precisão geométrica, como a reconstrução 3D por luz estruturada. As correções associadas são implementadas em bibliotecas como o OpenCV, por meio da função `cv::undistort`, que utiliza os coeficientes de distorção obtidos durante a calibração para restaurar a projeção do modelo.

2.3.2 Formação da Imagem

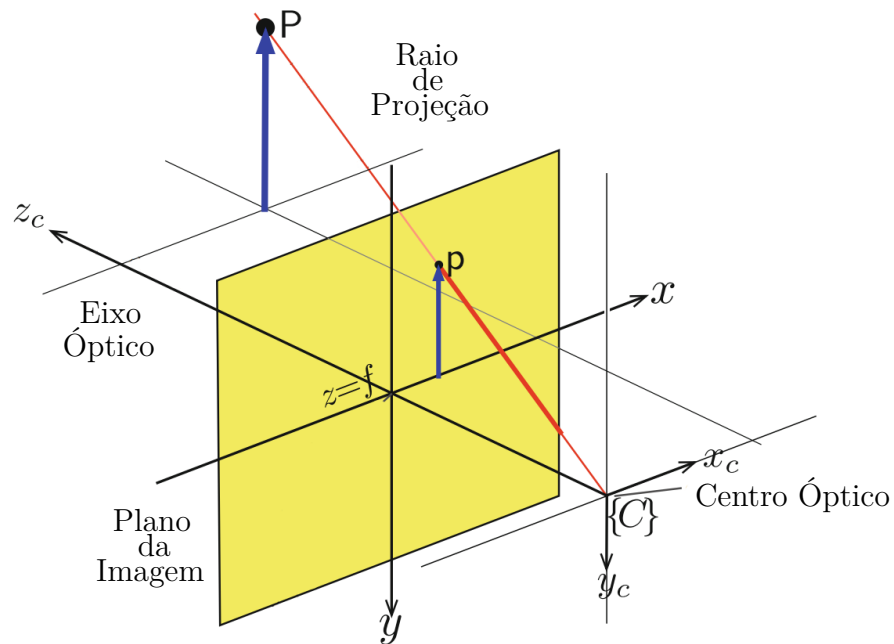
Dessa forma é necessário compreender a formação da imagem na câmera. Para isso considera-se a projeção de um ponto \mathbf{P} do espaço físico para o plano bidimensional da imagem de uma câmera, sendo preciso compreender também que um raio do ponto \mathbf{P} indo ao o ponto de origem da câmera \mathbf{C} , intersectará o plano da imagem em $z = f$ no ponto de projeção \mathbf{p} , conforme Figura 8 (Peter Corke Witold Jachimczyk, 2023).

Através do uso de semelhança de triângulos, é possível determinar que o vetor de coordenadas $\mathbf{P} = [x_c \ y_c \ z_c]^\top$ é projetado para o ponto no plano da imagem com uma coordenada $\mathbf{p} = [x \ y]^\top$ no plano imagem, como mostra a Figura 9. E portanto é possível inferir a relação:

$$x = f \cdot \frac{x_c}{z_c}, \quad y = f \cdot \frac{y_c}{z_c}, \quad (7)$$

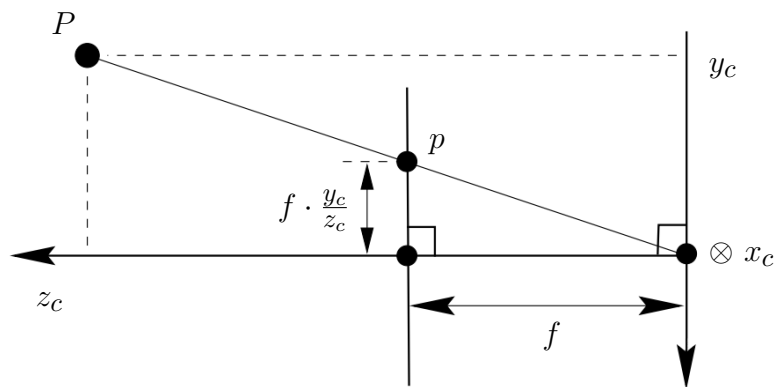
a qual descreve o mapeamento de um ponto \mathbf{P} do espaço tridimensional para o plano da imagem bidimensional ($P : \mathbb{R}^3 \mapsto \mathbb{R}^2$).

Figura 8 – Modelo de Projeção.



Fonte: Adpatado de (Peter Corke Witold Jachimczyk, 2023).

Figura 9 – Semelhança de Triângulos.



Fonte: Adpatado de (Hartley; Zisserman, 2003).

2.3.3 Modelo da Câmera

A partir das relações estabelecidas na Equação (7), é conveniente utilizar coordenadas homogêneas a fim de obter linearidade na conversão o ponto tridimensional para o ponto bidimensional. Desse modo, infere-se que o ponto \mathbf{P} e \mathbf{p} podem ser convertidos de coordenadas cartesianas para homogêneas conforme Equação (8) e Equação (9).

$$\mathbf{P} = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \longrightarrow \tilde{\mathbf{P}} = \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} \quad (8)$$

$$\mathbf{p} = \begin{bmatrix} x \\ y \end{bmatrix} \longrightarrow \tilde{\mathbf{p}} = \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{bmatrix} \quad (9)$$

E portanto é possível reescrevê-las como:

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} \quad (10)$$

Observe que o vetor $\tilde{\mathbf{p}} = [\tilde{x} \ \tilde{y} \ \tilde{z}]^T$ que representa o ponto projetado no plano imagem em coordenadas homogêneas pode ser convertido para coordenadas cartesianas (x, y) , dividindo o primeiro e o segundo elementos de $\tilde{\mathbf{p}}$ (isto é, \tilde{x} e \tilde{y}) pelo último (ou seja, \tilde{z}).

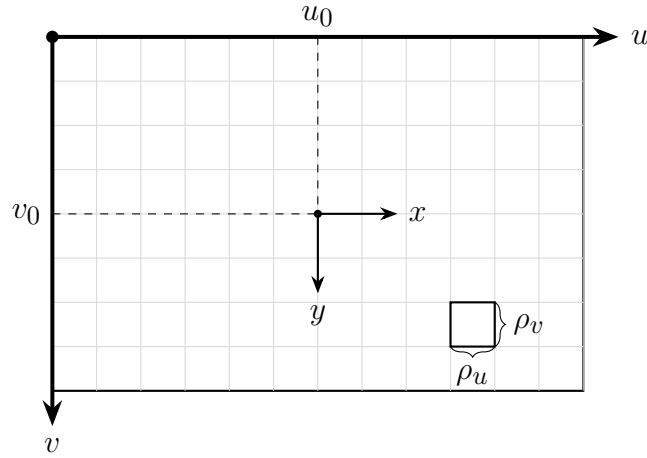
A matriz 3×4 Equação (10) é conhecida como matriz da câmera, que relaciona os pontos no mundo real com a imagem bidimensional. Agora, considerando uma câmera digital, com sensores sensíveis à luz que capturam a imagem e são formados por uma grade $W \times H$ que representa os elementos da imagem ou os *pixels*, conforme mostra Figura 10, as coordenadas finais do ponto projetado devem ser expressas como índices do *pixel* (u, v) ao invés das coordenadas x e y do plano imagem. Dessa forma deve-se realizar o mapeamento $(x, y) \mapsto (u, v)$, o qual é dado pela Equação (11), onde ρ_w e ρ_h são a largura e altura de cada conjunto de fotosensor que compõe o sensor da câmera e (u_0, v_0) é a localização da origem do plano da imagem em relação ao novo sistema de coordenadas u e v .

$$u = \frac{x}{\rho_w} + u_0, \quad v = \frac{y}{\rho_h} + v_0 \quad (11)$$

Dessa forma, é possível calcular as coordenadas do ponto projetado (em sua forma homogênea) em relação ao sistema de coordenadas u e v conforme a Equação (12).

$$\tilde{\mathbf{p}} = \begin{bmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{bmatrix} = \begin{bmatrix} \frac{1}{\rho_w} & s & u_0 \\ 0 & \frac{1}{\rho_h} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{bmatrix} \quad (12)$$

Substituindo (10) em (12), obtemos a equação que permite calcular as coordenadas do ponto projetado (na forma homogênea e em relação ao sistema de coordenadas u e v) a partir das coordenadas do ponto no espaço tridimensional, dado pelo vetor $\tilde{\mathbf{P}}$. Ou seja,

Figura 10 – Plano da Imagem com Grade de *Pixels*.

Fonte: Desenvolvido pelo autor.

$$\begin{bmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{1}{\rho_w} & s & u_0 \\ 0 & \frac{1}{\rho_h} & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{K}} \cdot \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix}, \quad (13)$$

onde a matriz \mathbf{K} , de dimensão 3×4 , é chamada de matriz de parâmetros intrínsecos da câmera.

E por fim, nas aplicações reais usando câmeras, geralmente os pontos no espaço tridimensional são representados a partir de um sistema de coordenadas externo que pode estar transladado e rotacionado em relação ao sistema de coordenadas da câmera. Nesses casos, em primeiro lugar deve-se mapear o ponto $\mathbf{P} = [X, Y, Z]^T$ representado em relação ao sistema de coordenadas externo para o sistema de coordenadas da câmera. Tal mapeamento pode ser realizado como

$$\begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{3 \times 3} & -\mathbf{t}_{3 \times 1} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \quad (14)$$

onde $\mathbf{t}_{3 \times 1}$ e $\mathbf{R}_{3 \times 3}$ são o vetor de translação e a matriz de rotação que modelam, respectivamente, a translação e a rotação do sistema de coordenadas externo em relação ao sistema de coordenadas da câmera. A matriz de rotação $\mathbf{R}_{3 \times 3}$ é dada pelo produto matricial das matrizes de rotação individuais em torno dos eixos x , y e z . Por fim, $\mathbf{0}$ é um vetor linha nulo.

Em particular, o vetor de translação $\mathbf{t}_{3 \times 1}$ corresponde ao vetor que contém as coordenadas do origem do sistema de coordenadas da câmera em relação ao sistema de coordenadas externo.

Do ponto de vista da rotação, é importante estabelecer previamente a ordem do produto matricial entre as rotações ao redor dos eixos. Ou seja, se primeiramente for realizada a rotação em relação ao eixo x , na sequência em relação ao eixo y e, por fim, em relação ao eixo z , a matriz de rotação $\mathbf{R}_{3 \times 3}$ é calculada como

$$\mathbf{R}_{3 \times 3} = \mathbf{R}_z(\alpha) \cdot \mathbf{R}_y(\beta) \cdot \mathbf{R}_x(\gamma), \quad (15)$$

onde a rotação com ângulo α em torno do eixo z é modelada pela matriz

$$\mathbf{R}_z = \begin{bmatrix} \cos(\alpha) & -\text{sen}(\alpha) & 0 \\ \text{sen}(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (16)$$

De forma similar, a rotação com ângulo β em torno do eixo y é representada pela matriz

$$\mathbf{R}_y = \begin{bmatrix} \cos(\beta) & 0 & \text{sen}(\beta) \\ 0 & 1 & 0 \\ -\text{sen}(\beta) & 0 & \cos(\beta) \end{bmatrix}, \quad (17)$$

Por fim, a rotação com ângulo γ em torno do eixo x é modelada por

$$\mathbf{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\beta) & -\text{sen}(\beta) \\ 0 & \text{sen}(\beta) & \cos(\beta) \end{bmatrix}. \quad (18)$$

Portanto, o modelo final e completo da câmera *pinhole* é dado pela Equação (19), onde \mathbf{K} é a matriz de parâmetros intrínsecos da câmera (que contém informações internas da câmera) e \mathbf{M} é a matriz de parâmetros extrínsecos da câmera (que modela a posição da câmera em relação ao sistema de coordenadas externo).

$$\begin{bmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{1}{\rho_w} & s & u_0 \\ 0 & \frac{1}{\rho_h} & v_0 \\ 0 & & 1 \end{bmatrix}}_{\mathbf{K}} \cdot \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \underbrace{\begin{bmatrix} \mathbf{R}_{3 \times 3} & -\mathbf{t}_{3 \times 1} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}}_{\mathbf{M}} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (19)$$

2.4 CALIBRAÇÃO DE CÂMERA

A calibração de câmera é um procedimento fundamental em diversas aplicações de visão computacional, como reconstrução tridimensional, metrologia óptica e sistemas de

luz estruturada para medição 3D. O objetivo da calibração é determinar, através de um procedimento experimental, os parâmetros intrínsecos e extrínsecos da câmera, bem como os coeficientes de distorção da lente (Hartley; Zisserman, 2003).

No modelo descrito na Seção 2.3.3, a relação entre um ponto no espaço tridimensional e a sua projeção no plano da imagem é descrita pela matriz de parâmetros intrínsecos \mathbf{K} e pelas matrizes de rotação e translação $[\mathbf{R} \mid \mathbf{t}]$. Entretanto, câmeras reais apresentam imperfeições ópticas e defeitos relacionadas ao processo de fabricação — como distorção radial, distorção tangencial, desalinhamento do sensor e variações da distância focal — que tornam necessário estimar esses parâmetros de forma experimental visando, por exemplo, correção dos efeitos dessas imperfeições. Especificamente, o processo de calibração consiste em resolver um problema de otimização que busca minimizar o erro de reprojeção entre pontos conhecidos do mundo e suas observações na imagem (Zhang, 2000).

Um procedimento de calibração robusto pode ser realizado a partir do processamento de várias fotografias contendo um padrão planar conhecido, posicionadas em orientações distintas. A partir da correspondência entre pontos detectados no padrão e suas coordenadas reais no referencial do tabuleiro, estimam-se inicialmente as matrizes de homografia e, em seguida, os parâmetros intrínsecos da câmera. Esse processo é posteriormente refinado por um ajuste de mínimos quadrados não lineares (*bundle adjustment*), resultando em valores precisos para a matriz intrínseca, para os vetores extrínsecos de cada imagem e para os coeficientes de distorção (Hartley; Zisserman, 2003; Zhang, 2000).

Em sistemas de digitalização 3D baseados em luz estruturada, a calibração torna-se ainda mais crítica, pois pequenos erros nos parâmetros da câmera propagam-se diretamente para os cálculos de profundidade obtidos pela triangulação. A qualidade da reconstrução depende da correta estimativa do centro óptico, da distância focal, da distorção da lente e da relação geométrica entre a câmera e o emissor de luz. A etapa de calibração é, portanto, indispensável para garantir que os pontos reconstruídos estejam no seu devido posicionamento tridimensional (Shao; Dong; Madessa, 2019).

Embora métodos clássicos de calibração, como o proposto por Zhang, permaneçam amplamente utilizados devido à sua eficiência e simplicidade, abordagens mais recentes têm buscado aumentar a robustez do processo por meio de estratégias de otimização adaptativa. Técnicas modernas formulam a calibração como um problema de otimização global, visando reduzir a sensibilidade a ruído, condições de iluminação variáveis e imperfeições na detecção dos pontos de referência. Nesse contexto, métodos baseados em algoritmos de otimização adaptativa demonstram maior estabilidade dos parâmetros intrínsecos, mesmo quando o erro médio de reprojeção apresenta valores semelhantes aos obtidos por métodos tradicionais, evidenciando que métricas adicionais de consistência devem ser consideradas na avaliação da calibração (Li; Xiao; Hu, 2025).

As subseções seguintes apresentam os componentes utilizados no processo de calibração adotado neste trabalho, incluindo o método de captura das imagens via HTTP, o

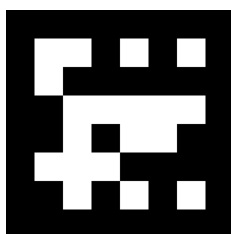
padrão ChArUco utilizado para a estimação dos parâmetros da câmera e uma discussão conceitual dos resultados da calibração.

2.4.1 Tabuleiro de Calibração (ChArUco)

A estimação precisa dos parâmetros intrínsecos e extrínsecos de uma câmera depende diretamente da qualidade e da robustez dos pontos de referência extraídos das imagens utilizadas na calibração. Tradicionalmente, padrões do tipo *chessboard* são amplamente empregados nesse processo, uma vez que seus cantos internos apresentam regularidade geométrica e são facilmente detectáveis em diferentes orientações do plano (Zhang, 2000). No entanto, tais padrões são sensíveis a oclusões parciais, desfoque e mudanças abruptas de iluminação, o que pode dificultar a detecção consistente dos cantos.

Com o objetivo de tornar a detecção mais robusta, o OpenCV introduziu os marcadores ArUco (mostrado na Figura 11), que consistem em padrões binários quadrados capazes de codificar identificadores únicos em cada marcador (Lin et al., 2022). Esses marcadores podem ser detectados de forma estável mesmo em condições de iluminação adversas, e sua identificação individual permite estabelecer correspondências confiáveis entre diferentes imagens. Apesar disso, a precisão espacial dos vértices dos marcadores é inferior àquela obtida nos cantos de *chessboards*, devido à segmentação binária e às bordas mais espessas dos marcadores.

Figura 11 – Exemplo de Aruco.



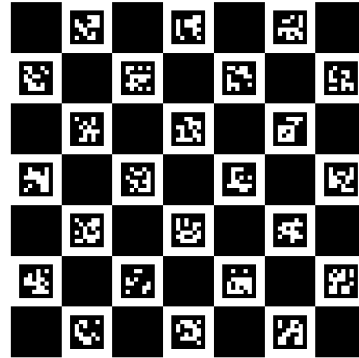
Fonte: Desenvolvido pelo autor.

O padrão ChArUco (apresentado na Figura 12) combina os dois conceitos ao integrar marcadores ArUco dentro de um tabuleiro do tipo *chessboard*. A partir da detecção inicial dos marcadores individuais, suas poses aproximadas são estimadas e, em seguida, os cantos subjacentes do tabuleiro são interpolados com alta precisão, mesmo quando partes do padrão estão parcialmente ocluídas ou fora de foco. Esse processo, denominado *corner interpolation*, resulta em um conjunto de pontos 2D bem distribuídos e com localização sub-pixel, ideal para calibração de câmera (Garrido-Jurado et al., 2014).

Uma vantagem adicional do ChArUco é sua capacidade de produzir correspondências válidas mesmo com poucos marcadores visíveis. Isso torna possível calibrar a câmera utilizando imagens capturadas em ângulos extremos ou em situações em que apenas uma

fração do tabuleiro está no campo de visão. Em conjunto com o algoritmo de calibração de Zhang (2000), a abordagem ChArUco proporciona um método eficiente, robusto e amplamente utilizado para a obtenção dos parâmetros intrínsecos, extrínsecos e dos coeficientes de distorção da lente.

Figura 12 – Tabuleiro ChArUco.



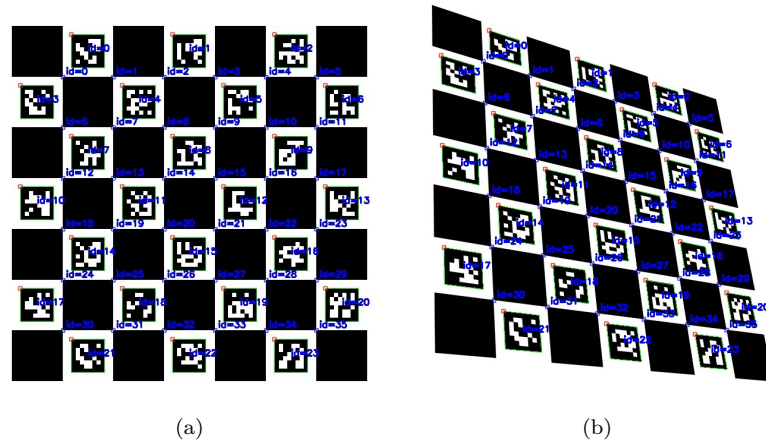
Fonte: Desenvolvido pelo autor.

Os marcadores ArUco são padrões fiduciais binários projetados para permitir detecção robusta, identificação única e estimação de pose a partir de imagens adquiridas por uma câmera (Garrido-Jurado et al., 2014). Cada marcador consiste em um contorno quadrado preto que circunscreve uma matriz interna de bits (geralmente 4×4 , 5×5 ou 6×6), cujas combinações formam um código capaz de representar um identificador (*ID*) único. Os marcadores de uma mesma família compõem um dicionário, no qual cada padrão interno é construído para maximizar a distância de *Hamming* entre entradas, reduzindo ambiguidades durante a detecção.

Do ponto de vista computacional, a detecção de um marcador ArUco no OpenCV ocorre em três etapas principais. Primeiro, um processo de segmentação identifica regiões quadradas na imagem que possuam bordas bem definidas e contraste adequado. Em seguida, cada candidato é normalizado por meio de uma transformação de perspectiva e convertido para uma representação binária. Finalmente, o padrão interno é comparado com todos os elementos do dicionário selecionado, sendo escolhido aquele cuja distância de *Hamming* é mínima. Se a distância estiver dentro do limite permitido, o ID correspondente é atribuído ao marcador detectado, conforme Figura 13.

A presença de um ID único em cada marcador é fundamental para aplicações de calibração e rastreamento, pois permite estabelecer correspondências confiáveis entre múltiplas imagens de um mesmo padrão. No caso dos tabuleiros ChArUco, os marcadores ArUco fornecem as referências iniciais para estimar a orientação aproximada do padrão e, a partir dela, interpolar os cantos do tabuleiro com ótima precisão. Dessa forma, a estabilidade da detecção ArUco, aliada à precisão da localização dos cantos provenientes

Figura 13 – (a) ChArUco visto de frente com chaves identificadas. (b) ChArUco visto em perspectiva com chaves identificadas.



Fonte: Desenvolvido pelo autor.

da estrutura tipo *chessboard*, resulta em um método de calibração robusto e adequado para ambientes com variação de iluminação, oclusões parciais e diferentes distâncias de captura.

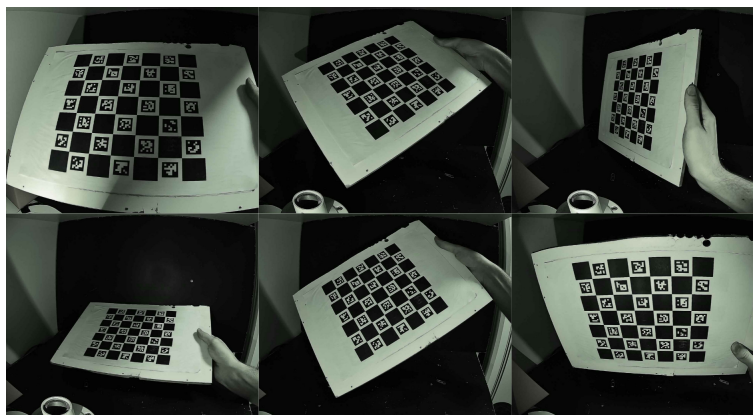
2.4.2 Resultados da Calibração

O processo de calibração adotado neste trabalho segue a lógica clássica dos métodos baseados em padrões planares, conforme proposto inicialmente por Zhang (Zhang, 2000) e amplamente difundido em bibliotecas modernas como o OpenCV. A partir de imagens adquiridas do tabuleiro ChArUco em diferentes poses (ver Figura 14 em ambiente de aplicação), o procedimento realiza três etapas principais: detecção dos marcadores, interpolação dos cantos do tabuleiro e estimação dos parâmetros intrínsecos e extrínsecos da câmera.

Na primeira etapa, para cada imagem capturada, são detectados os marcadores ArUco presentes no tabuleiro. A detecção envolve a segmentação das regiões quadradas, normalização por homografia e identificação do marcador com base na comparação do padrão binário interno com as entradas do dicionário utilizado (Garrido-Jurado et al., 2014). Cada marcador detectado fornece uma estimativa aproximada da pose local do tabuleiro.

Em seguida, o método realiza a interpolação dos cantos internos do tabuleiro ChArUco por meio da função `interpolateCornersCharuco`. Essa etapa aproveita a identificação dos marcadores para estimar a orientação do tabuleiro e, com isso, calcular a posição dos vértices do padrão de *chessboard* em alta precisão. Como consequência, obtêm-se, para cada imagem, um conjunto de pontos 2D altamente precisos associados às respectivas chaves das células do tabuleiro.

Figura 14 – Exemplos de imagens do ChArUco em diferentes poses para calibração de câmera.



Fonte: Desenvolvido pelo autor.

Com todos os conjuntos de cantos detectados, realiza-se a etapa de calibração propriamente dita por meio da função `calibrateCameraCharuco`. O algoritmo resolve um problema de otimização que busca minimizar o erro de reprojeção entre os pontos observados na imagem e os pontos correspondentes projetados pelo modelo *pinhole*. A solução do problema resulta na estimação da matriz intrínseca \mathbf{K} , dos coeficientes de distorção da lente, que são detalhados adiante, e dos vetores de rotação e translação associados a posição relativa da câmera em relação ao tabuleiro ChArUco de cada imagem utilizada na calibração. O valor retornado pela função corresponde ao erro quadrático médio de reprojeção (RMS), métrica amplamente utilizada para quantificar a qualidade da calibração (Hartley; Zisserman, 2003).

Por fim, o conjunto de parâmetros obtidos é armazenado para uso posterior nas etapas de reconstrução tridimensional. A matriz de parâmetros intrínsecos e os coeficientes de distorção permitem aplicar a transformação de *undistort* às imagens, garantindo que a projeção da câmera seja corrigida de acordo com o modelo pinhole ideal. Já os parâmetros extrínsecos estimados para cada pose do tabuleiro servem como base para o alinhamento geométrico do sistema, assegurando consistência espacial ao longo de todo o processo de triangulação.

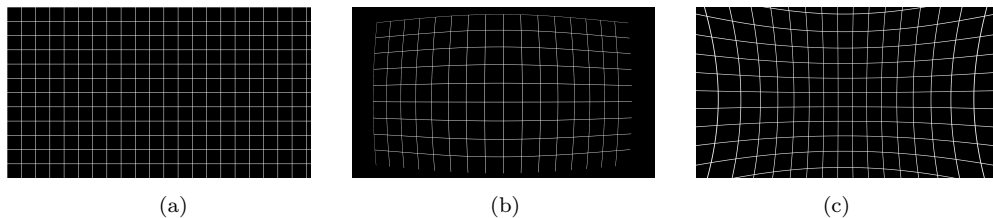
2.4.2.1 Lentes e Matriz de Distorção

Muitas vezes ao teorizar um modelo de câmera, de forma ideal, não se leva em consideração a lente usada pela câmera. No entanto, de forma prática, nenhuma lente é perfeita e a depender desta, podem ocorrer imperfeições que podem acentuar distorções na imagem. Dentre as distorções pode-se citar as que geram alteração de cor, como a aberração cromática e aberração esférica, e também aquelas que geram distorções geométricas, onde

pontos da imagem são deslocados seguindo o padrão de deformação da lente (Bradski; Kaehler, 2008).

As distorções geométricas são tradicionalmente classificadas em dois grupos: distorções radiais e distorções tangenciais. As distorções radiais resultam da curvatura da lente e manifestam-se como deformações simétricas em torno do centro óptico, produzindo efeitos como *barril* ou *almofada*. Já as distorções tangenciais decorrem do desalinhamento entre as superfícies ópticas da lente e o sensor, fazendo com que os raios sejam desviados de forma não radial, ocasionando alongamentos ou cisalhamentos nos padrões reticulados (Hartley; Zisserman, 2003). A Figura 15 ilustra os principais tipos de distorção radial, destacando os padrões característicos de distorção do tipo barril e do tipo almofada.

Figura 15 – (a) Grade sem distorção de lente. (b) Grade com distorção do tipo Barril. (c) Grade com distorção do tipo Almofada.



Fonte: Desenvolvido pelo Autor.

O OpenCV utiliza o modelo de distorção conhecido como modelo de Brown–Conrady, amplamente adotado em sistemas de calibração. Nesse modelo, os coeficientes de distorção radial são representados por k_1 , k_2 , k_3 , enquanto as distorções tangenciais são representadas por p_1 e p_2 . A correção geométrica é aplicada durante o processo de calibração, de forma que a imagem corrigida (*undistorted*) se aproxime da projeção ideal descrita pelo modelo *pinhole* (Bradski; Kaehler, 2008). Esses efeitos podem ser observados na Figura 16, onde (a) apresenta a imagem distorcida e (b) sua versão corrigida após a calibração.

Figura 16 – (a) Vista da câmera com distorção da lente. (b) Vista da câmera sem distorção da lente.



Fonte: Desenvolvido pelo autor.

Durante o processo de calibração utilizando o tabuleiro ChArUco, os coeficientes de distorção são estimados simultaneamente aos parâmetros intrínsecos. Uma vez determinados, esses coeficientes permitem corrigir a imagem por meio da função `cv::undistort`, reestabelecendo a relação projetiva ideal. A remoção da distorção é essencial para aplicações de reconstrução tridimensional baseadas em triangulação, pois mesmo pequenas deformações introduzidas pela lente propagam-se diretamente para o cálculo da profundidade, afetando a precisão da nuvem de pontos resultante.

3 DESENVOLVIMENTO

3.1 CONCEPÇÃO DO SISTEMA

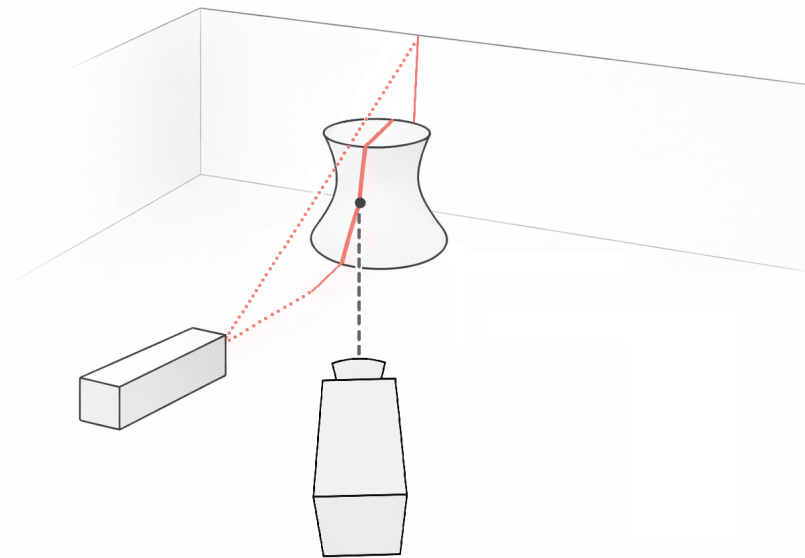
O sistema desenvolvido neste trabalho consiste em um *scanner 3D* baseado na técnica de luz estruturada por linha a laser que projeta um feixe linear sobre a superfície do objeto, combinado com uma plataforma rotativa para aquisição sequencial de sua geometria. A concepção do projeto foi guiada pelo objetivo de obter um sistema de digitalização tridimensional de baixo custo, modular e suficientemente preciso para a reconstrução de objetos de pequenas e médias dimensões, preservando detalhes geométricos relevantes.

A arquitetura adotada fundamenta-se no princípio da triangulação óptica, no qual a profundidade de um ponto iluminado é determinada a partir da interseção entre o raio de visão da câmera e o plano formado pelo feixe de laser. Diferentemente de sistemas baseados em projeção de padrões complexos, optou-se pela utilização de uma linha a laser fixa, cuja simplicidade estrutural reduz a complexidade computacional e facilita a implementação, ao mesmo tempo em que mantém adequada precisão para aplicações de engenharia reversa e análise geométrica.

Para permitir a reconstrução completa do objeto, uma mesa rotativa é utilizada para realizar a varredura angular da superfície. A cada incremento de rotação, uma nova imagem é capturada, resultando em um conjunto de seções tridimensionais que, quando combinadas, formam a nuvem de pontos final. Essa abordagem elimina a necessidade de mecanismos de varredura do laser, transferindo a complexidade do movimento para a plataforma, o que simplifica o controle do sistema e aumenta sua repetibilidade.

Do ponto de vista funcional, o sistema é composto por três subsistemas principais: aquisição de dados, calibração geométrica e reconstrução tridimensional. O subsistema de aquisição engloba a câmera IP, o emissor laser e o mecanismo de rotação, sendo responsável pela captura sincronizada das imagens. O subsistema de calibração inclui a estimação dos parâmetros intrínsecos da câmera, da orientação da mesa rotativa e do plano do laser, assegurando a coerência geométrica entre todos os elementos do sistema. Por fim, o subsistema de reconstrução utiliza os parâmetros estimados para realizar a triangulação e gerar a nuvem de pontos e as malhas tridimensionais correspondentes.

A Figura 17 ilustra a concepção geral do sistema, destacando os principais elementos como a câmera, o gerador do feixe de laser e o objeto escaneado. O sistema completo, pode ser visto na Seção 3.2.2, na Figura 23, que abordará a integração de todos os componentes e ilustra como estão fisicamente dispostos. A partir dessa concepção geral, as seções seguintes também detalham a implementação física e computacional do sistema, bem como os procedimentos empregados para sua calibração e operação.

Figura 17 – Esquemático da cena do *scanner*.

Fonte: Desenvolvido pelo autor.

3.2 ARQUITETURA DE *HARDWARE*

A arquitetura de *hardware* do sistema de digitalização tridimensional foi projetada de modo a garantir estabilidade mecânica, sincronização entre aquisição e movimento, e integração eficiente com o sistema de processamento computacional. O conjunto do *hardware* é responsável por posicionar o objeto, projetar a linha de luz estruturada, capturar as imagens necessárias e transmitir os dados ao computador, onde ocorre o processamento e a reconstrução tridimensional.

O sistema foi concebido de forma modular, permitindo que cada componente desempenhe uma função bem definida e possa ser substituído ou ajustado sem comprometer o funcionamento global. Essa abordagem facilita tanto a manutenção quanto futuras expansões do projeto, além de reduzir o acoplamento entre as partes mecânica, eletrônica e computacional.

De maneira geral, o *hardware* é composto por quatro elementos principais: uma câmera IP responsável pela captura das imagens, um emissor de luz retilínea ajustado para projetar um plano de luz que intersecta o objeto, uma mesa rotativa acionada por motor de passo para realizar a varredura angular, e um microcontrolador responsável pelo controle do movimento. Esses elementos são integrados a um computador central, que coordena a aquisição dos dados, executa os algoritmos de calibração e estimação geométrica e realiza a reconstrução tridimensional.

3.2.1 Componentes Utilizados

3.2.1.1 Câmera IP

A captura das imagens utilizadas tanto na calibração quanto na aquisição dos dados de reconstrução tridimensional é realizada por meio de uma câmera IP da fabricante Hikvision, conforme Figura 18. A escolha por uma câmera IP, em detrimento de câmeras convencionais com interface USB, deve-se principalmente à disponibilidade. Apesar de não se tratar de uma câmera industrial própria para visão computacional, que usualmente apresentam custos de entrada elevados, trata-se de uma câmera de monitoramento de boa qualidade que permite o ajuste de vários de parâmetros e configurações através de sua interface gráfica.

Figura 18 – Câmera IP Hikvision.



Fonte: (Hikdistribution, 2025)

A comunicação entre a câmera e o computador é realizada por meio do protocolo RTSP para *stream* de vídeo e HTTP para as capturas de imagens estáticas, utilizando conexão Ethernet. Essa abordagem permite a aquisição de quadros em tempo real, sem a necessidade de armazenamento intermediário. A alimentação elétrica da câmera é fornecida por meio de um injetor PoE (*Power over Ethernet*), que possibilita a transmissão simultânea de dados e energia através de um único cabo Ethernet. Essa solução reduz a complexidade da fiação do sistema e contribui para uma montagem mais organizada e confiável.

Do ponto de vista de visão computacional, a câmera IP é tratada como um sensor monocular convencional que pode ser posteriormente calibrada visando a estimação de seus parâmetros intrínsecos e coeficientes de distorção de lente. A qualidade da reconstrução tridimensional depende diretamente da resolução do sensor, da estabilidade da captura e da precisão da calibração, fatores que tornam a escolha da câmera um elemento crítico no desempenho global do sistema.

3.2.1.2 Emissor de Luz Estruturada

O emissor de luz é o elemento responsável pela projeção de uma linha estruturada utilizada no processo de digitalização tridimensional. No sistema proposto, emprega-se um laser configurado para gerar uma linha luminosa, obtida por meio da expansão do feixe pontual em um plano de luz. Essa configuração permite que a interseção entre o plano do laser e a superfície do objeto resulte em uma curva visível, cuja posição varia de acordo com a geometria tridimensional do objeto.

A escolha por um laser de linha (como mostra Figura 19) deve-se à simplicidade geométrica do método e à sua ampla utilização em sistemas de triangulação óptica, além do dispositivo escolhido apresentar custo reduzido. Diferentemente de projetores de padrões complexos, a projeção de uma única linha reduz significativamente a complexidade do sistema, tanto do ponto de vista de hardware quanto de processamento computacional. Além disso, essa abordagem apresenta elevada robustez frente a variações de iluminação ambiente, uma vez que a linha de laser possui alto contraste em relação ao fundo.

Figura 19 – Módulo laser 5mW com alinhamento em linha.



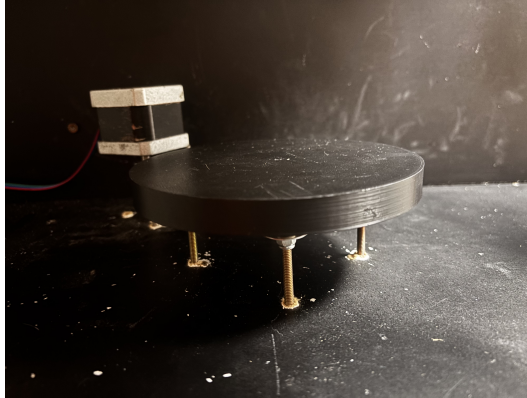
Fonte: (Proesi, 2025)

3.2.1.3 Mesa Rotativa e Motor de Passo

A mesa rotativa é o componente responsável pela varredura angular do objeto durante o processo de digitalização tridimensional. Em vez de movimentar o emissor de laser ou a câmera, optou-se por rotacionar o objeto em torno de um eixo vertical fixo, permitindo que sucessivas seções da superfície sejam iluminadas e capturadas de forma controlada. Essa estratégia simplifica o modelo geométrico do sistema e garante maior estabilidade durante a aquisição dos dados. A mesa é um disco feito com termoplástico ABS (Acrilonitrila Butadieno Estireno), devido à maior resistência física, com 130 milímetros de diâmetro ($\emptyset 130\text{ mm}$) parafusado a um rolamento e com uma polia dentada fixa e centralizada, conforme mostrado na Figura 20.

O acionamento da mesa é realizado por meio de um motor de passo, escolhido por sua capacidade de fornecer controle preciso e repetível do movimento angular. Motores de passo são amplamente utilizados em sistemas de automação e posicionamento devido

Figura 20 – Mesa rotativa de ABS com polia dentada acoplada a rolamento.



Fonte: Desenvolvido pelo autor.

à sua alta resolução, facilidade de controle em malha aberta e confiabilidade. Cada pulso aplicado ao motor resulta em um deslocamento angular conhecido, permitindo definir com precisão o número de passos por revolução e, conseqüentemente, a resolução angular do escaneamento.

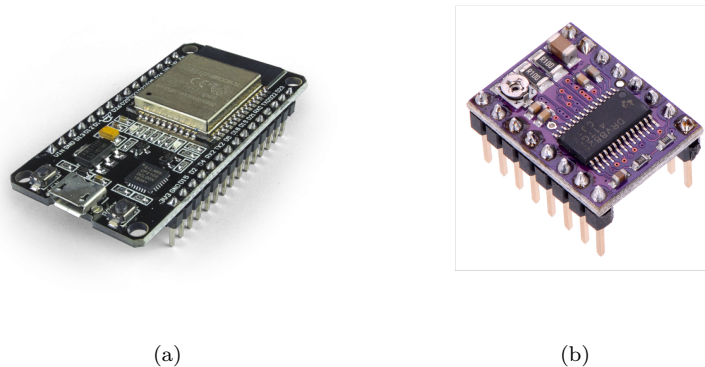
No contexto deste projeto, a rotação incremental da mesa permite associar cada imagem capturada a um ângulo específico em relação ao eixo de rotação. Essa associação é fundamental para a reconstrução da nuvem de pontos global, pois cada curva tridimensional obtida pela interseção da linha de laser com o objeto deve ser corretamente posicionada no espaço tridimensional. A precisão do eixo de rotação e a repetibilidade do movimento influenciam diretamente a coerência geométrica entre as diferentes seções reconstruídas.

3.2.1.4 Microcontrolador (ESP32) e Driver DRV8825

O controle do movimento da mesa rotativa é realizado por meio de um microcontrolador ESP32 em conjunto com um *driver* de motor de passo DRV8825, mostrados na Figura 21. Essa combinação é responsável por gerar os sinais de controle adequados para o acionamento preciso do motor, garantindo repetibilidade e resolução angular compatíveis com as exigências do processo de digitalização tridimensional.

O ESP32 atua como unidade de controle principal, recebendo comandos enviados pelo computador central por meio de uma conexão USB com comunicação serial. Esses comandos definem parâmetros como número de passos, sentido de rotação e velocidade do movimento. A utilização de um microcontrolador dedicado permite separar as tarefas de controle de baixo nível do processamento computacional mais complexo, que permanece concentrado no computador responsável pela aquisição de imagens e reconstrução 3D. Já a escolha do ESP32 se deu pela disponibilidade, ótimo desempenho e compatibilidade lógica com outros microcontroladores, como o Arduino, que esteve presente na primeira

Figura 21 – (a) Microcontrolador ESP32. (b) Driver para motor de passo DRV8825.



Fonte: (RoboCore, 2025) e (AaranaCorp, 2025).

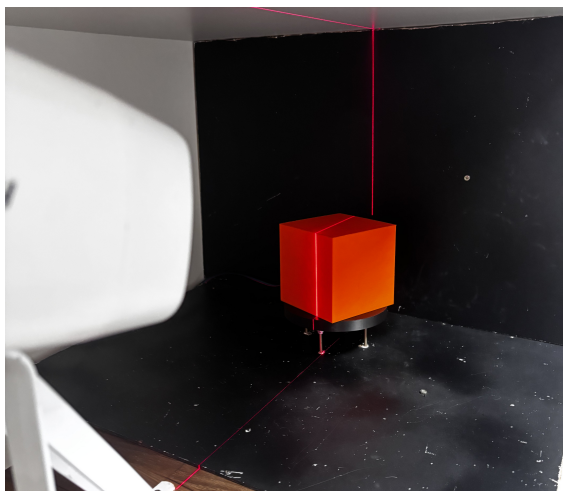
versão de protótipo deste projeto.

O *driver* DRV8825 é responsável por realizar o acionamento elétrico do motor de passo a partir dos sinais de controle fornecidos pelo ESP32. Esse *driver* permite operar o motor tanto em passo inteiro quanto em modos de *microstepping*, nos quais cada passo mecânico é subdividido por meio de comandos eletrônicos adequados. Considerando um motor de passo padrão com 200 passos por revolução, o uso do DRV8825 possibilita configurações equivalentes a 400, 800, 1600 ou até 3200 passos por revolução, dependendo da divisão selecionada. Esse *driver* foi escolhido pelo seu custo reduzido e excelente capacidade de controlar *microstepping* no motor de passo.

3.2.2 Integração dos Componentes

A integração dos componentes de *hardware* foi realizada de modo a garantir comunicação confiável, sincronização entre movimento e aquisição de imagens e simplicidade na montagem do sistema. Para isso, adotou-se uma arquitetura centralizada, na qual o computador atua como unidade de coordenação, enquanto os demais elementos desempenham funções específicas e bem definidas, como mostra Figura 22, que apresenta o cenário do *scanner*, com a câmera direcionada para o objeto em cima da mesa rotativa sendo intersectado pelo laser.

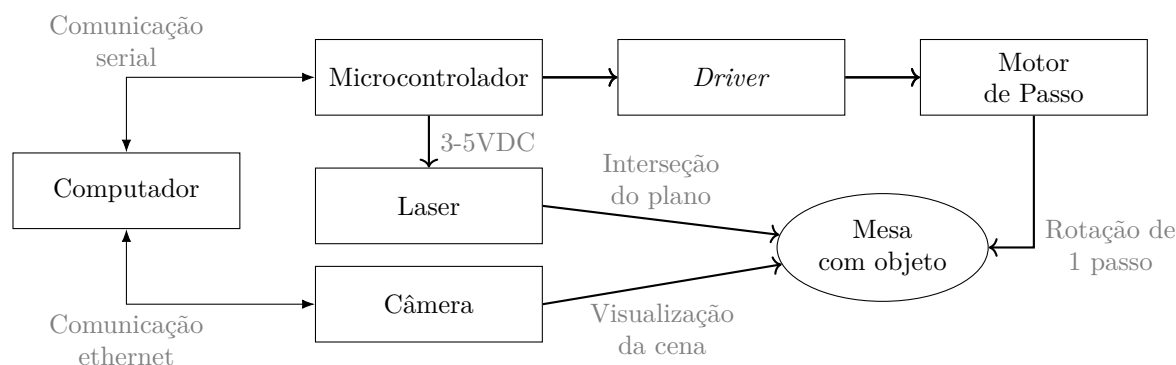
A câmera IP é conectada ao computador por meio de um cabo Ethernet, utilizando o protocolo RTSP (*Real-Time Streaming Protocol*) para transmissão contínua do fluxo de vídeo e capturas usando chamadas de API para o *SDK* nativo da câmera. A alimentação elétrica da câmera é realizada através de um injetor PoE (*Power over Ethernet*), o que possibilita o fornecimento simultâneo de energia e dados por um único cabo. Essa solução reduz a quantidade de conexões físicas necessárias e contribui para uma instalação mais organizada e robusta.

Figura 22 – Cena do *scanner* vista da lateral direita da câmera.

Fonte: Desenvolvido pelo autor.

O microcontrolador ESP32 é conectado ao computador por meio de uma interface USB, sendo utilizada comunicação serial para o envio de comandos de controle e sincronização. Através dessa interface, o computador envia instruções relacionadas à rotação da mesa, como número de passos e sentido do movimento, enquanto o ESP32 executa essas instruções de forma determinística, acionando o *driver* DRV8825 e, conseqüentemente, o motor de passo, como indicado no diagrama da Figura 23.

Figura 23 – Diagrama da disposição do componentes no sistema.



Fonte: Desenvolvido pelo Autor.

A sincronização entre a rotação da mesa e a aquisição das imagens é realizada pelo *software* de controle executado no computador. Para cada incremento angular da mesa rotativa, é realizada a captura de um conjunto de quadros provenientes da chamada de API da câmera. Essa estratégia garante que cada imagem ou segmento do vídeo esteja associado a um ângulo conhecido em relação ao eixo de rotação, permitindo o correto posicionamento das seções tridimensionais durante a reconstrução. Ou seja, o computador envia para o microcontrolador um comando para acionar o *driver* e rotacionar um passo e

este por sua vez, retorna que a operação foi realizada. Na sequência, o computador realiza uma captura de imagem da cena e, ao obtê-la, repete o comando ao microcontrolador para realizar mais um passo de rotação e assim sucessivamente até completar uma revolução.

Observe que a separação clara entre as funções de aquisição, controle de movimento e processamento computacional com um fluxo de informação de ponta a ponta, confere maior robustez ao sistema. Eventuais atrasos na captura de imagens ou no processamento não afetam diretamente o controle do motor, enquanto ajustes nos parâmetros de rotação podem ser realizados sem impactar o modelo de calibração da câmera ou do laser. Essa abordagem modular facilita a depuração, a manutenção e a expansão futura do sistema, além de garantir maior confiabilidade durante o processo de escaneamento tridimensional.

3.3 ARQUITETURA DE *SOFTWARE*

A arquitetura de *software* do sistema de digitalização tridimensional foi projetada para organizar de forma clara e modular as diferentes etapas do processo de reconstrução 3D, desde a aquisição dos dados até a geração da nuvem de pontos e das malhas finais. O *software* atua como elemento central de coordenação do sistema, sendo responsável pela comunicação com os dispositivos de *hardware*, pela execução dos procedimentos de calibração e pela aplicação dos algoritmos de reconstrução geométrica.

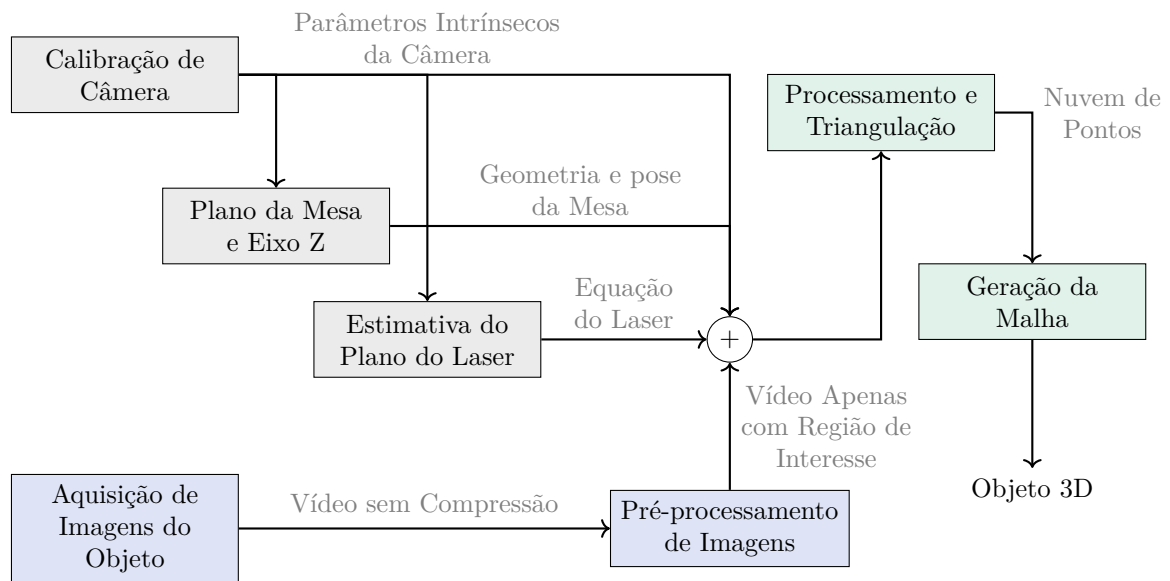
O desenvolvimento foi realizado majoritariamente na linguagem *Python*, escolhida por sua ampla adoção em aplicações de visão computacional, pela disponibilidade de bibliotecas consolidadas e pela facilidade de prototipagem. A utilização de bibliotecas como OpenCV, NumPy e Open3D permite implementar algoritmos eficientes para processamento de imagens, operações geométricas e manipulação de nuvens de pontos, além de facilitar a visualização e o pós-processamento dos resultados. A organização dos módulos de *software* pode ser vista na Figura 24

O módulo de calibração reúne os procedimentos necessários para a estimação dos parâmetros intrínsecos da câmera, dos coeficientes de distorção, dos parâmetros dos planos da mesa rotativa e do laser. Na Figura 24 é possível identificar os blocos da cor cinza (●) como pertencentes a esse módulo. Os parâmetros estimados nessa etapa são persistidos e reutilizados nas etapas subsequentes.

O módulo de aquisição e pré-processamento de dados é responsável pelo controle da mesa rotativa, pela captura das imagens a partir do *SDK* da câmera e pela organização temporal dos dados adquiridos. Na Figura 24 é possível identificar os blocos da cor azul (●) como pertencentes a esse módulo. Esse módulo garante a sincronização entre movimento e aquisição, associando cada imagem ou conjunto de quadros a um ângulo conhecido da mesa rotativa.

O módulo de reconstrução tridimensional implementa os algoritmos responsáveis pela extração da linha do laser nas imagens, pela triangulação dos pontos por meio da interseção raio-plano e pela composição das diferentes seções reconstruídas em uma nuvem

Figura 24 – Diagrama do funcionamento do sistema.



Fonte: Desenvolvido pelo Autor.

de pontos global. Na Figura 24 é possível identificar os blocos da cor verde (●) como pertencentes a esse módulo. Por fim, o módulo de pós-processamento aplica técnicas de filtragem, suavização e geração de malhas triangulares, permitindo a obtenção de modelos tridimensionais adequados para visualização, análise e exportação.

Nas seções seguintes, cada um desses módulos é detalhado individualmente, apresentando os procedimentos adotados, as principais decisões de projeto e as estratégias utilizadas para garantir a robustez e a precisão do sistema.

3.4 CALIBRAÇÃO DA CÂMERA

A calibração da câmera constitui uma etapa fundamental do sistema, pois todos os cálculos geométricos subsequentes dependem diretamente da precisão dos parâmetros intrínsecos estimados. Nesta seção descrevem-se os procedimentos práticos adotados para a aquisição das imagens utilizadas na calibração, bem como os cuidados experimentais considerados durante esse processo.

Para a realização do projeto fixou-se alguns parâmetros da câmera que serão completamente justificados ao decorrer do documento. Dentre eles ressaltam-se:

- Contraste ajustado para valor máximo, permitindo que os detalhes, principalmente em variações abruptas de cor fossem bem definidos;
- Saturação foi ajustada para valor mínimo, tendenciando a imagem para escala de cinza;
- Tempo de exposição ajustado para $\frac{1}{75}$, ou seja, fazendo com que o obturador permanecesse aberto por $\frac{1}{75}$ de segundo, permitindo que o sensor receba luz por

um tempo bastante curto, porém sem escurecer demais a imagem, equilibrando o brilho;

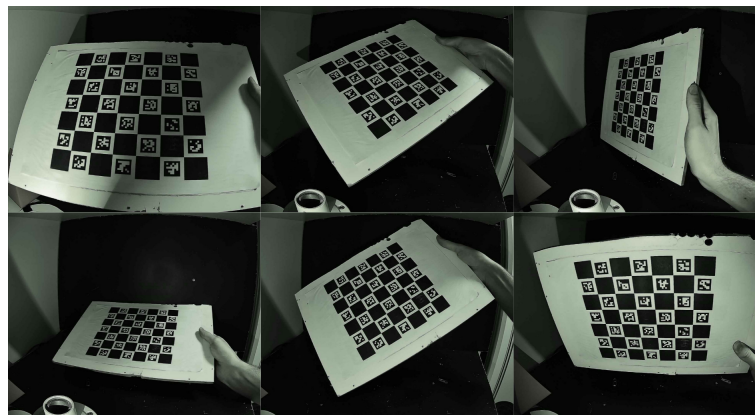
- Ganho ajustado para 30, permitindo equilíbrio juntamente com o tempo de exposição de modo que um ponto na imagem com alta reflexão de luz não cause saturação de branco ou “estouro de luz”.

3.4.0.1 Aquisição das Imagens de Calibração

As imagens de calibração foram adquiridas utilizando a câmera IP integrada ao sistema. Dois fluxos de vídeo da câmera são utilizados: (i) fluxo RTSP para visualização da cena através da câmera pelo operador do *scanner* e (ii) fluxo para captura da imagem através de requisição HTTP. A captura dos quadros é realizada diretamente no computador, sem compressão adicional, de forma a preservar a qualidade das imagens utilizadas no processo de calibração.

Para a calibração, foi utilizado um tabuleiro *ChArUco*, posicionado manualmente em diferentes orientações e distâncias em relação à câmera como mostra a Figura 25, que compila algumas das várias imagens utilizadas na calibração. O tabuleiro é movimentado de forma a cobrir uma ampla região do campo de visão, incluindo variações significativas de inclinação, rotação e profundidade. Essa diversidade de poses é essencial para garantir a observabilidade adequada dos parâmetros intrínsecos da câmera e reduzir ambiguidades no processo de estimação.

Figura 25 – Exemplos de imagens do ChArUco para calibração.



Fonte: Desenvolvido pelo autor.

3.4.1 Detecção do Padrão ChArUco

A detecção do padrão ChArUco constitui a etapa central do processo de calibração, pois é a partir dela que são obtidas as correspondências entre pontos tridimensionais

conhecidos no espaço do tabuleiro e suas projeções bidimensionais na imagem. O padrão ChArUco combina marcadores ArUco com um tabuleiro do tipo *chessboard*, explorando as vantagens de ambos: a identificação robusta por meio de marcadores codificados e a alta precisão geométrica dos cantos internos do tabuleiro.

Inicialmente, para cada imagem capturada, é realizada a detecção dos marcadores ArUco presentes no tabuleiro. Esse processo envolve a identificação de regiões quadradas na imagem, a normalização geométrica dessas regiões e a comparação do padrão binário interno com um dicionário pré-definido. Cada marcador detectado é associado a um identificador único (*ID*), o que permite reconhecer sua posição relativa no tabuleiro mesmo em situações de oclusão parcial ou quando apenas uma fração do padrão está visível.

A partir da detecção dos marcadores ArUco, procede-se à interpolação dos cantos do tabuleiro ChArUco. Essa etapa utiliza a geometria conhecida do tabuleiro e os identificadores dos marcadores detectados para estimar com precisão subpixel a posição dos vértices internos do padrão de *chessboard*, usando as funções da biblioteca OpenCV `cv2.aruco.detectMarkers()` e `cv2.aruco.interpolateCornersCharuco()`. A interpolação dos cantos permite obter um conjunto mais denso e geometricamente consistente de pontos, aumentando significativamente a precisão da calibração em comparação ao uso exclusivo de marcadores ArUco.

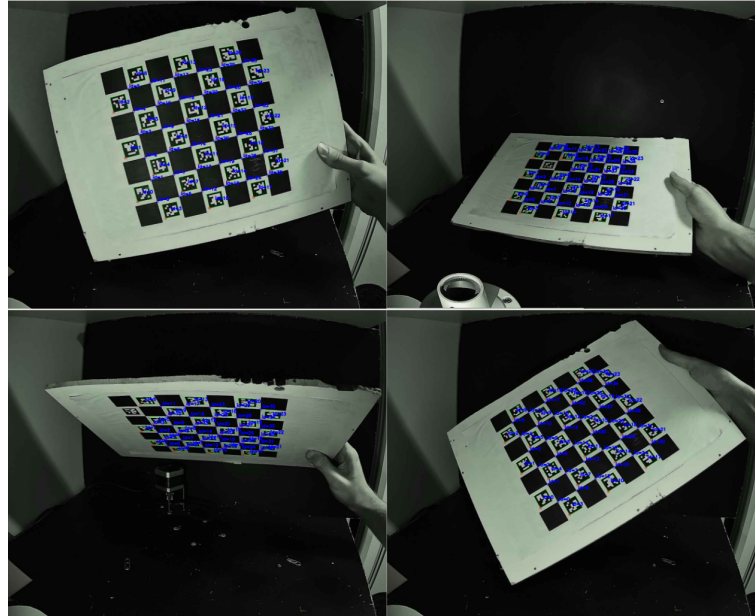
As imagens capturadas são selecionadas automaticamente com base na detecção bem sucedida do padrão ChArUco. Apenas quadros nos quais um número mínimo de marcadores é identificado são considerados válidos para a calibração, retornando para o operador os resultados individuais de identificação dos ChArUcos de cada imagem, como mostra a Figura 26, ressaltando a identificação da chave única de cada ArUco presente do tabuleiro. Essa estratégia contribui para aumentar a robustez do processo, descartando imagens com baixa qualidade de detecção ou com o tabuleiro parcialmente fora do campo de visão.

3.4.2 Estimação dos Parâmetros Intrínsecos da Câmera

A estimação dos parâmetros intrínsecos da câmera é realizada a partir das correspondências entre os pontos bidimensionais detectados nas imagens e as coordenadas tridimensionais conhecidas do tabuleiro ChArUco. Esse processo tem como objetivo determinar os parâmetros que descrevem o modelo interno da câmera e é realizado através da função `cv2.aruco.calibrateCameraCharuco()` também do OpenCV, que retorna a matriz de parâmetros intrínsecos da câmera, coeficientes de distorção e uma lista dos vetores de rotação e translação de todos tabuleiros identificados.

A partir do conjunto de imagens válidas, contendo cantos ChArUco detectados, o problema de calibração é formulado como um problema de otimização não linear, que é resolvido usando o Algoritmo de Levenberg-Marquardt (LMA) (Bradski; Kaehler, 2008; OpenCV, 2025). O algoritmo busca minimizar o erro de reprojeção, definido como a

Figura 26 – Exemplos de retornos da identificação de ChArUcos para calibração.



Fonte: Desenvolvido pelo autor.

distância média entre os pontos observados na imagem e os pontos projetados pelo modelo da câmera a partir das coordenadas conhecidas do tabuleiro (Zhang, 2000).

Como resultado do processo de calibração, obtém-se a matriz intrínseca da câmera, dada por

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \quad (20)$$

onde f_x e f_y representam as distâncias focais e (c_x, c_y) correspondem às coordenadas do ponto principal da imagem. Além disso, são estimados os coeficientes de distorção radial e tangencial da lente, responsáveis por modelar as deformações geométricas introduzidas pelo sistema óptico.

O algoritmo de calibração também estima, para cada imagem utilizada, os parâmetros extrínsecos correspondentes à pose do tabuleiro em relação à câmera. Esses parâmetros são expressos por vetores de rotação e translação, os quais permitem relacionar o sistema de coordenadas do tabuleiro ao sistema da câmera. Embora esses parâmetros não sejam utilizados diretamente na reconstrução final do objeto, eles desempenham papel fundamental na estimação do plano do laser e na validação da coerência geométrica do sistema.

Os parâmetros intrínsecos e os coeficientes de distorção estimados nesta etapa são armazenados em um arquivo `.npz` da biblioteca Numpy, o qual é bastante útil e

versátil para armazenamento e leitura de vetores que serão utilizados em todas as etapas subsequentes do sistema, incluindo a correção das imagens, a estimação do plano da mesa, a calibração do plano do laser e o processo de triangulação. Dessa forma, a calibração da câmera estabelece a base geométrica sobre a qual todo o sistema de reconstrução tridimensional é construído.

3.5 ESTIMAÇÃO DO SISTEMA DE REFERÊNCIA DA MESA ROTATIVA

A reconstrução tridimensional baseada na rotação do objeto requer que todas as seções reconstruídas sejam expressas em um sistema de referência comum. Para isso, é necessário estimar com precisão a geometria da mesa rotativa em relação à câmera, incluindo a posição do plano da mesa e a orientação do eixo de rotação, que sai do centro da mesa e perpendicularmente aponta para cima. Esses elementos permitem alinhar corretamente as diferentes curvas tridimensionais obtidas ao longo do processo de varredura, garantindo coerência espacial na nuvem de pontos final.

A estimação do sistema de referência da mesa é realizada em duas etapas principais. Inicialmente, estima-se o plano da mesa rotativa, o qual define uma superfície de referência sobre a qual o objeto é apoiado. Em seguida, determina-se o eixo de rotação associado ao movimento da mesa, assumido como o eixo Z do sistema de coordenadas do objeto. Essas duas informações permitem definir transformações rígidas que alinham as nuvens parciais reconstruídas em um referencial comum.

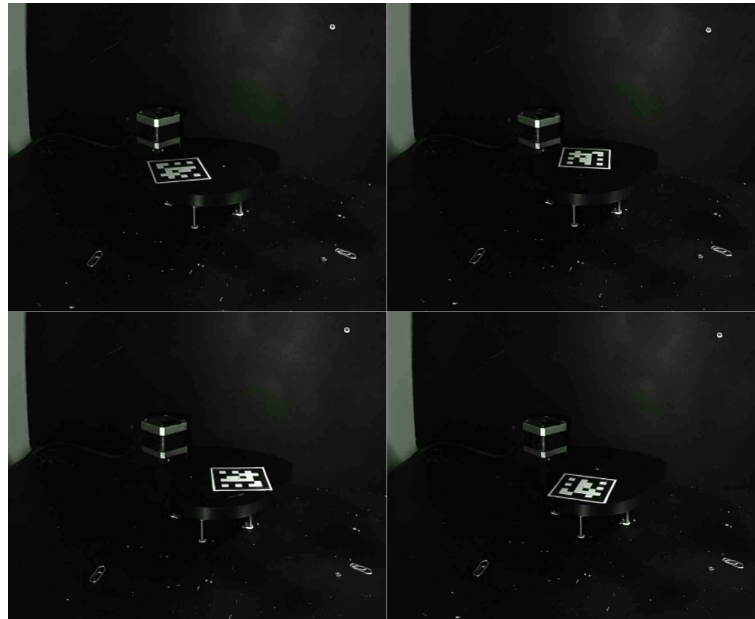
A correta determinação do plano da mesa e do eixo de rotação é fundamental para evitar artefatos geométricos indesejados na reconstrução, como inclinações indesejadas, deslocamentos verticais inconsistentes ou deformações helicoidais, alongamentos e achatamentos na nuvem de pontos. Além disso, essa etapa possibilita separar de forma clara os efeitos do movimento controlado da mesa das variações geométricas reais do objeto escaneado. Na sequência, apresenta-se o método adotado para a estimação do plano da mesa e do eixo de rotação associado, cuja implementação foi realizada por meio de um algoritmo desenvolvido em Python¹.

3.5.1 Método de Estimação do Plano e Eixo Z da Mesa

A estimação do plano da mesa e do eixo de rotação (Z) é realizada a partir de princípios geométricos semelhantes aos empregados nas etapas de calibração baseadas em marcadores fiduciais do tipo ArUco. Para esse procedimento, um único marcador é rigidamente posicionado sobre a superfície da mesa rotativa, propositalmente o mais afastado do seu centro geométrico possível, e então submetido a uma sequência de aquisições de imagens ao longo de uma rotação completa da mesa, como mostra Figura 27.

¹ O código-fonte correspondente a esta implementação encontra-se disponível em: https://github.com/BrunoBronzeri/Scanner3D_TCC/blob/main/Laser/GenZAxis1.py.

Figura 27 – Exemplos de imagens do ArUco sobre a mesa rotativa.



Fonte: Desenvolvido pelo autor.

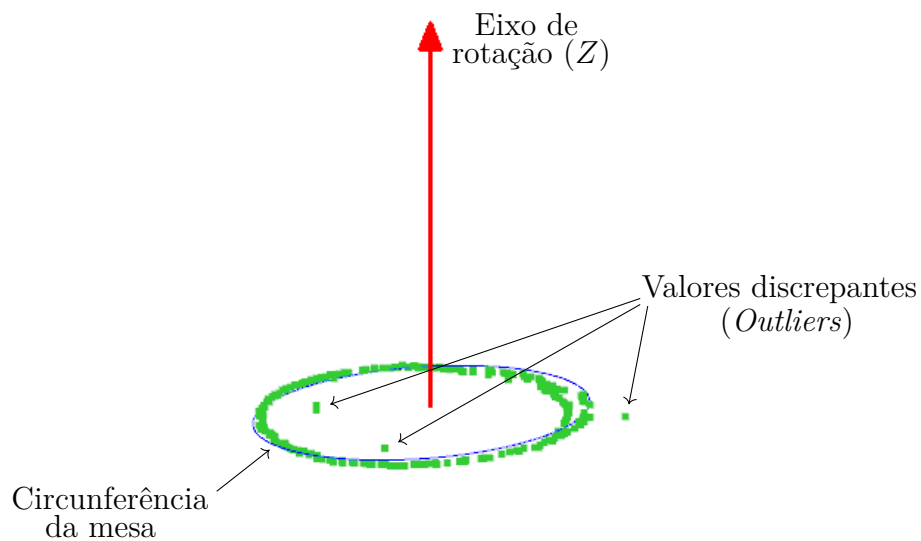
Durante a rotação, a pose tridimensional do marcador é estimada em cada imagem, permitindo obter sua posição espacial em relação ao sistema de coordenadas da câmera. Como a distância entre o marcador e o centro da mesa permanece constante ao longo do movimento, o conjunto de posições tridimensionais estimadas descreve, idealmente, uma circunferência no espaço. Essa circunferência corresponde à trajetória do marcador durante a rotação e encontra-se contida no plano da mesa.

A partir desse conjunto de pontos tridimensionais, é possível ajustar geometricamente um plano que melhor representa a superfície da mesa, bem como estimar a circunferência associada à órbita do marcador, que pertence ao plano da mesa. O centro dessa circunferência fornece uma estimativa direta do centro de rotação da mesa, enquanto o vetor normal ao plano ajustado define o eixo de rotação da mesma, assumido como o eixo Z do sistema da mesa. Para isso ser possível inicialmente, calcula-se o centróide do conjunto de pontos, o qual é utilizado para centralizar os dados e eliminar o efeito de translação. Em seguida, aplica-se a decomposição em valores singulares (SVD) à matriz formada pelos pontos centralizados (função `np.linalg.svd(pontos - centroide)` da biblioteca Numpy), ou seja a matriz de entrada do método é dada pela diferença de cada ponto com o centróide, resultando em uma matriz que contém o vetor ortogonal ao plano localizado no centróide, portanto representando o eixo de rotação Z da mesa.

É possível visualizar na Figura 28 uma representação tridimensional feita com a biblioteca Open3D para validação da estimação. Nessa figura, destacam-se a circunferência em azul que está contida no plano da mesa, o eixo vertical, perpendicular em vermelho

localizado no centróide, que trata-se do eixo de rotação da mesa (eixo Z), e os pontos em verde que representam a pose de cada ArUco em diferentes imagens capturadas ao longo da revolução. Fica evidente a necessidade da utilização de um número grande de imagens devido a ruídos (*outliers*) que contribuiriam para erros de estimação com poucas amostras.

Figura 28 – Representação 3D da estimação da circunferência que descreve a mesa e do eixo de rotação.



Fonte: Desenvolvido pelo autor.

Esse método apresenta elevada robustez, uma vez que a estimação do eixo e do plano é realizada a partir de múltiplas observações distribuídas ao longo de toda a revolução da mesa, reduzindo significativamente o impacto de ruídos de medição e erros pontuais na estimação da pose do marcador. Além disso, a suposição de circularidade da trajetória do ArUco é fisicamente coerente com o movimento imposto pelo sistema mecânico, o que contribui para a precisão do modelo geométrico obtido.

3.5.2 Estimação do Plano do Laser

A estimação do plano do laser constitui uma etapa fundamental no sistema de digitalização tridimensional proposto, uma vez que a reconstrução dos pontos 3D baseia-se diretamente na interseção entre o raio de visão da câmera e o plano definido pelo feixe luminoso do laser. Assim, a determinação precisa da posição e da orientação desse plano no sistema de coordenadas da câmera é essencial para garantir a exatidão do processo de triangulação e, conseqüentemente, da nuvem de pontos reconstruída.

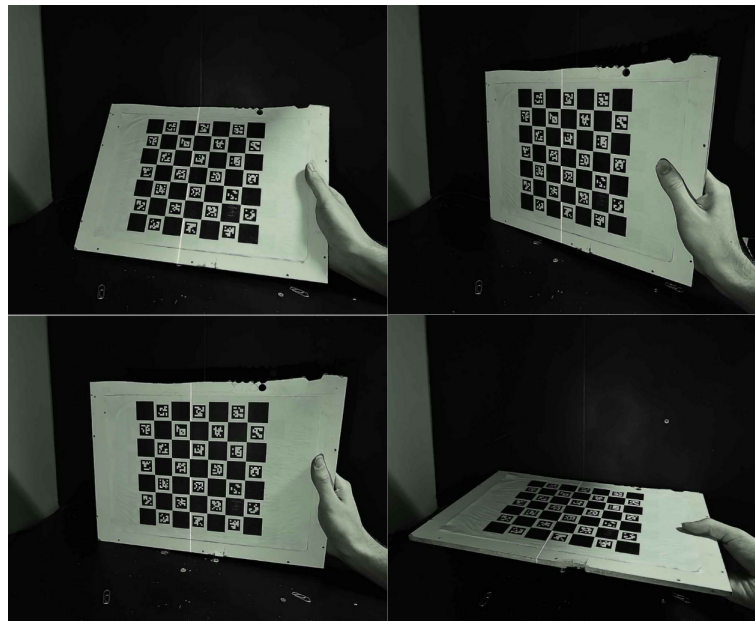
No sistema desenvolvido e implementado em Python², o emissor de laser projeta

² O código-fonte correspondente a esta implementação encontra-se disponível em: https://github.com/BrunoBronzeri/Scanner3D_TCC/blob/main/Laser/EstimatePlane.py.

uma linha luminosa fixa no espaço, a qual pode ser modelada geometricamente como um plano tridimensional. Esse plano permanece invariante ao longo de todo o processo de aquisição, visto que o laser encontra-se rigidamente fixado em relação à câmera. Entretanto, devido a tolerâncias mecânicas, desalinhamentos e imprecisões inerentes ao processo de montagem, não é possível assumir previamente a equação desse plano, tornando necessária sua estimação experimental.

Para esse fim, utiliza-se um tabuleiro ChArUco como referência geométrica conhecida. Ao projetar a linha de laser sobre o tabuleiro posicionado em diferentes orientações e posições dentro do campo de visão da câmera, obtém-se, para cada configuração, um conjunto de pontos pertencentes à interseção entre o plano do laser e o plano do tabuleiro como mostra a Figura 29. Uma vez que a pose do tabuleiro é estimada com precisão a partir dos marcadores, torna-se possível reconstruir tridimensionalmente esses pontos no referencial da câmera.

Figura 29 – Exemplos de imagens do ChArUco para a estimação do plano do laser.



Fonte: Desenvolvido pelo autor.

Para cada imagem do tabuleiro ChArUco interceptado pelo feixe de laser, o processo de extração dos pontos tridimensionais inicia-se com a estimação da pose do tabuleiro no sistema de coordenadas da câmera, a partir dos marcadores detectados. Conhecida essa pose, o plano do ChArUco passa a ser descrito explicitamente no espaço tridimensional, permitindo que pontos observados na imagem sejam reconstruídos no referencial da câmera por meio de interseção geométrica.

Em cada imagem, o operador seleciona manualmente dois pontos em coordenadas de *pixel* (u, v) pertencentes à linha luminosa projetada pelo laser sobre o tabuleiro. Esses

dois pontos definem uma reta no plano da imagem, a partir da qual são gerados múltiplos pontos intermediários por interpolação. Essa estratégia permite aumentar a densidade de amostragem do feixe luminoso e reduzir a sensibilidade a imprecisões pontuais na seleção manual.

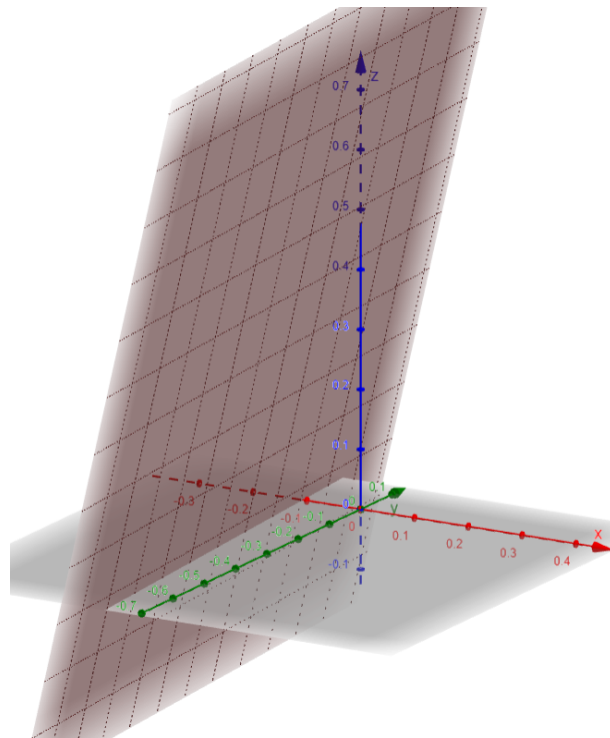
Cada ponto interpolado na imagem define um raio de visão no sistema de coordenadas da câmera, obtido a partir da calibração intrínseca e da correção de distorções da lente. Assumindo que esses pontos pertencem à superfície plana do tabuleiro ChArUco, calcula-se a interseção de cada raio com o plano do tabuleiro, resultando em um conjunto de pontos tridimensionais expressos diretamente no referencial da câmera. Dessa forma, para cada imagem, obtém-se uma amostra tridimensional da interseção entre o plano do laser e o plano do ChArUco.

O conjunto global de pontos tridimensionais obtido a partir de múltiplas imagens, capturadas com diferentes posições e orientações do tabuleiro, pertence ao mesmo plano físico correspondente ao feixe luminoso do laser. A partir desse conjunto, ajusta-se um modelo de plano tridimensional por meio de mínimos quadrados ortogonais, utilizando novamente a decomposição em valores singulares (SVD). Esse procedimento fornece uma estimativa robusta do vetor normal do plano do laser e do termo de deslocamento associado, resultando na equação do plano na forma geral $\mathbf{n}^\top \mathbf{X} + d = 0$, expressa no sistema de coordenadas da câmera conforme é possível visualizar na Figura 30. Nessa figura é notável a coerência com a geometria real do projeto, já que o centro óptico da câmera e o emissor de luz distam-se em aproximadamente 100 mm , sendo possível observar que o plano do laser intersecta o eixo X em aproximadamente $-0,1\text{ m}$ e que o raio da câmera que se encontra com o raio do laser pouco acima da superfície da mesa possui comprimento de aproximadamente 500 mm , sendo possível observar também que o plano do laser intersecta o eixo Z em aproximadamente $0,5\text{ m}$.

A orientação da normal estimada é ajustada de modo a manter consistência com o sistema de coordenadas da câmera, garantindo que o plano do laser esteja corretamente orientado em relação ao eixo óptico. Uma vez estimado, esse plano permanece fixo ao longo de todo o processo de aquisição, sendo utilizado diretamente nas etapas subsequentes de triangulação para a reconstrução tridimensional da nuvem de pontos.

A utilização de referências geométricas conhecidas, como padrões fiduciais, para a estimação do plano do laser é amplamente discutida na literatura como uma estratégia robusta para reduzir a sensibilidade do processo a ruídos de medição e imprecisões na detecção da linha luminosa. Trabalhos recentes demonstram que métodos de calibração do plano do laser baseados em entidades geométricas bem definidas, como linhas e planos estimados a partir de múltiplas observações, apresentam maior estabilidade e precisão quando comparados a abordagens baseadas exclusivamente em pontos isolados. Em particular, Yu et al. (Yu et al., 2021) propõem um método de calibração do plano do laser fundamentado na representação por linhas de *Plücker*, evidenciando ganhos de robustez

Figura 30 – Representação do plano estimado a partir do sistema de coordenadas da câmera.



Fonte: Desenvolvido pelo autor.

e consistência geométrica na estimação do plano, aspectos diretamente relacionados à precisão da reconstrução tridimensional.

3.6 PROCESSO DE AQUISIÇÃO DE DADOS DE ESCANEAMENTO

O processo de aquisição de dados é responsável por estabelecer a ligação entre o sistema físico de escaneamento e os algoritmos de reconstrução tridimensional. Nessa etapa, o objeto é submetido a uma varredura angular controlada, enquanto imagens da linha de laser projetada sobre sua superfície são capturadas e organizadas de forma sistemática para posterior processamento.

Diferentemente de sistemas baseados em captura instantânea de múltiplas vistas, o método adotado neste trabalho realiza a aquisição de forma sequencial, associando cada conjunto de imagens a um ângulo específico da mesa rotativa. Essa estratégia permite construir a geometria tridimensional do objeto a partir de seções sucessivas. Para garantir o bom desempenho desse procedimento, é importante que o sistema esteja inserido em um ambiente com iluminação controlada, ou seja, o sistema possui no momento do escaneamento um ambiente sem fonte de luz fora o emissor do laser.

A aquisição de dados envolve três aspectos fundamentais: o controle preciso da rotação da mesa (varreduras de 1600 passos por revolução), a sincronização entre o movimento

mecânico e a captura das imagens e a organização estruturada dos dados adquiridos. Novamente para armazenamento massivo de imagens (1600 por objeto escaneado), optou-se por compilar as imagens com compressão FFV1 (*FF Video Codec 1*) em um vídeo no formato AVI (*Audio Video Interleave*), com extensão *.avi*, mantendo a organização dos diretórios e clareza nas operações de reconstrução.

Os parâmetros fixos de imagem configurados na câmera são definidos e justificados nesta etapa do projeto, uma vez que exercem influência direta na robustez da detecção da linha de laser. Em ambientes iluminados, a identificação do feixe luminoso costuma ser realizada por meio de segmentação por cor. Essa abordagem, entretanto, pode se mostrar pouco robusta em situações nas quais o objeto apresenta coloração semelhante à do laser ou quando a superfície possui alto grau de refletância. Nesses casos, é comum que a linha projetada apresente regiões periféricas avermelhadas, enquanto seus *pixels* centrais tendem a saturar o sensor, assumindo coloração próxima ao branco.

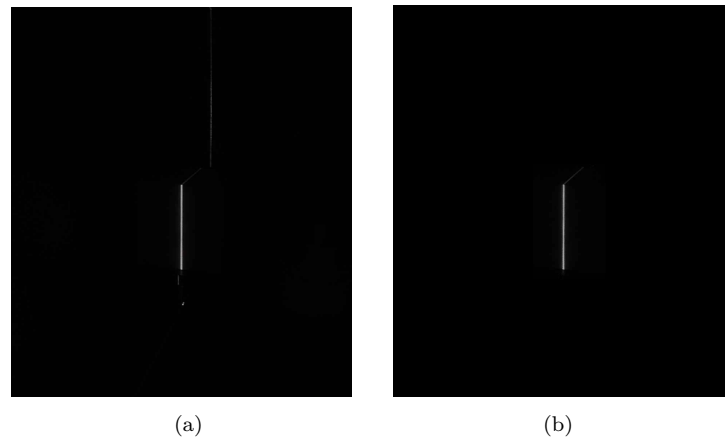
Por outro lado, em ambientes completamente escuros, o problema de confusão cromática é mitigado, uma vez que apenas a linha de laser permanece visível na cena. Contudo, essa configuração intensifica o efeito de saturação do sensor, conhecido como “estouro de luz”, o que pode resultar em imprecisões significativas na localização dos *pixels* pertencentes à linha e, conseqüentemente, na reconstrução tridimensional.

No modelo proposto, adota-se uma estratégia intermediária baseada na configuração manual e fixa dos parâmetros de imagem da câmera. A combinação de baixa saturação com alto contraste faz com que a linha do laser seja percebida como branca, independentemente de variações cromáticas do objeto. Além disso, o uso de baixa exposição aliado a um ganho reduzido minimiza o efeito de saturação, evitando o “estouro de luz” e preservando a definição espacial da linha projetada, conforme é possível ver na Figura 31 (a), que mostra um quadro do vídeo de escaneamento de um objeto cúbico nas condições descritas. No entanto ao final da aquisição das imagens, nota-se que a linha do laser atinge o anteparo e a mesa rotativa, que não pertencem à região de interesse. Por isso, outro algoritmo é utilizado, no qual o operador manualmente seleciona duas coordenadas na imagem: o ponto mais alto do objeto (u_h, v_h) e o ponto mais baixo (u_l, v_l). O algoritmo então percorre quadro a quadro do vídeo aplicando uma máscara que tornam nulos os valores dos *pixels* que ficam de fora da região de interesse, resultando na Figura 31 (b). Considerando a largura da imagem possuindo w *pixels*, a região de interesse por ser sempre representada conforme mostrado na Figura 32, onde são anulados *pixels* fora da região de interesse selecionada com valor menor a v_h , maior que v_l , menor que $\frac{w}{2} - 100$ *pixels* ou $\frac{w}{2} + 100$ *pixels*.

3.7 RECONSTRUÇÃO DA NUVEM DE PONTOS

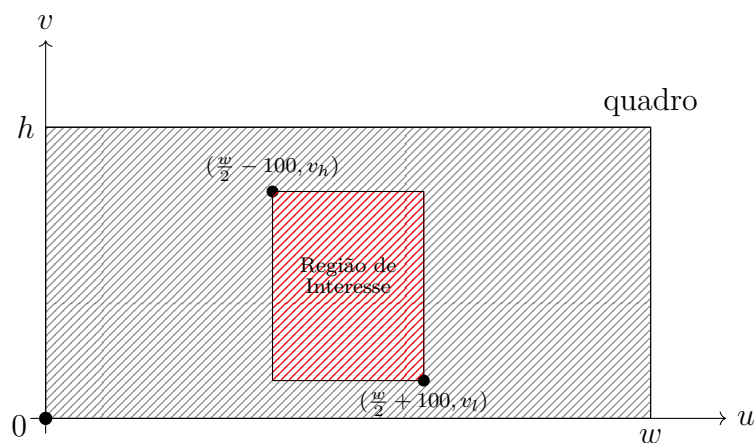
A reconstrução da nuvem de pontos é realizada a partir do um vídeo capturado durante a rotação do objeto, no qual a linha do laser é observada quadro a quadro,

Figura 31 – (a) Imagem para reconstrução sem processamento de região. (b) Imagem para reconstrução com processamento de região.



Fonte: Desenvolvido pelo autor.

Figura 32 – Definição da região de interesse.



Fonte: Desenvolvido pelo autor.

como descrito anteriormente. Para cada quadro do vídeo, extrai-se o perfil bidimensional correspondente à linha luminosa, estima-se sua posição com precisão de subpixel e realiza-se a triangulação por interseção raio-plano, obtendo-se as coordenadas de um conjunto de pontos no espaço tridimensional em relação ao referencial da câmera (Pauly; Gross; Kobbelt, 2002). Em seguida, esses pontos são mapeados para o referencial da mesa rotativa e rotacionados de acordo com o ângulo associado ao quadro, compondo progressivamente a nuvem de pontos global.

O algoritmo implementado organiza esse fluxo em três etapas principais: (i) segmentação e extração do perfil do laser na imagem; (ii) triangulação utilizando os parâmetros da câmera e do plano estimado do laser; e (iii) composição das seções reconstruídas em um referencial comum associado à mesa, produzindo a nuvem final, posteriormente exportada

em formato `.ply` para pós-processamento e geração de malha.

3.7.1 Pré-processamento e Extração da Linha do Laser

O primeiro passo do fluxo de processamento, consiste em destacar a linha do laser no quadro capturado. Cada quadro é convertido para escala de cinza, e em seguida aplica-se uma limiarização global para obter uma máscara binária dos *pixels* com alta intensidade luminosa. Seja \mathbf{I} a imagem em tons de cinza; a máscara \mathbf{M} é obtida por:

$$\mathbf{M} = \begin{cases} 255, & \mathbf{I} \geq \tau \\ 0, & \text{caso contrário} \end{cases}$$

onde τ é um limiar ajustado experimentalmente para isolar o feixe e no contexto de visão computacional geralmente é conhecido por *threshold*.

Após a limiarização, utiliza-se uma operação morfológica de fechamento com um elemento estruturante quadrado, de modo a preencher pequenas lacunas e reduzir ruídos localizados. O procedimento descrito é implementado conforme o trecho de código apresentado a seguir (Quadro 1), no qual se realiza a conversão para escala de cinza, a limiarização global e a operação morfológica de fechamento. Essa etapa melhora a continuidade da linha segmentada e aumenta a estabilidade da extração do perfil ao longo das linhas da imagem.

Quadro 1 – Pré-processamento e extração da máscara binária do laser

```

1 # Transforma imagem para escala de cinza
2 gray = cv2.cvtColor(frame_bgr, cv2.COLOR_BGR2GRAY)
3 # Aplica limiarizacao
4 _, mask = cv2.threshold(gray, THRESH, 255, cv2.THRESH_BINARY)
5 # Define elemento estruturante e aplica operacao morfologica
6 kernel = np.ones((KERNEL_SIZE, KERNEL_SIZE), np.uint8)
7 mask = cv2.morphologyEx(mask, cv2.MORPH_CLOSE, kernel)

```

Fonte: Desenvolvido pelo autor.

3.7.2 Extração Subpixel por Centroide

Para cada coordenada vertical v (isto é, para cada linha da imagem), determina-se o conjunto de colunas u para as quais $\mathbf{M}(v, u) = 255$. Se existirem mais de um *pixel* com valor 255 nessa linha, obtém-se as coordenadas $[u_0, u_1]$, correspondentes ao primeiro e ao último *pixel* com valor 255. Em seguida, calcula-se a coordenada horizontal subpixel $\hat{u}(v)$ por meio do centroide ponderado pela intensidade:

$$\hat{u}(v) = \frac{\sum_{u=u_0}^{u_1} u \mathbf{I}}{\sum_{u=u_0}^{u_1} \mathbf{I}}.$$

A seguir é possível visualizar o trecho de código que implementa essa etapa (Quadro 2). Essa estratégia reduz a sensibilidade a variações locais de espessura da linha, bem como a pequenas saturações centrais, produzindo uma estimativa contínua de posição ao longo do perfil.

Quadro 2 – Extração subpixel por centróide

```

1 # Identifica as colunas u onde o laser foi segmentado na linha v
2 xs = np.where(mask[v, :] == 255)[0]
3 if len(xs) < min_pixels:
4     return None
5 # Define o intervalo horizontal [u0, u1] ocupado pelo feixe na linha v
6 u0, u1 = xs[0], xs[-1]
7 # Extrai as intensidades de cinza correspondentes ao feixe segmentado
8 intens = gray[v, u0:u1+1].astype(np.float64)
9 s = intens.sum()
10 # Gera as coordenadas horizontais associadas a cada pixel do feixe
11 coords = np.arange(u0, u1+1, dtype=np.float64)
12 # Calcula a coordenada subpixel como centroide ponderado pela
    intensidade
13 u_sub = (coords * intens).sum() / s

```

Fonte: Desenvolvido pelo autor.

3.7.3 Suavização do Perfil ao Longo da Imagem

Após a extração de $\hat{u}(v)$ para múltiplas linhas, aplica-se um filtro de mediana unidimensional ao vetor resultante, com o objetivo de eliminar descontinuidades abruptas e *outliers* provenientes de reflexos, ruídos ou falhas de segmentação. O filtro de mediana pode ser implementado em Python pela função `cv2.medianBlur()` e tem como característica preservar a forma global do perfil enquanto reduz variações impulsivas (alterações abruptas de curta duração), contribuindo para maior estabilidade na triangulação subsequente.

3.7.4 Triangulação por Interseção Raio-Plano

Uma vez obtidos os pontos bidimensionais $(\hat{u}(v), v)$ pertencentes à linha do laser, realiza-se a triangulação para calcular suas coordenadas tridimensionais. Primeiramente, cada ponto é corrigido das distorções da lente e convertido para coordenadas normalizadas por meio da operação da função `undistort()` do OpenCV. O resultado são coordenadas (x_n, y_n) no plano normalizado da câmera, a partir das quais define-se o vetor diretor do raio:

$$\mathbf{d} = \begin{bmatrix} x_n \\ y_n \\ 1 \end{bmatrix}.$$

No sistema de coordenadas da câmera, o plano do laser é representado por

$$\mathbf{n}^\top \mathbf{X} + d = 0,$$

onde \mathbf{n} é o vetor normal e d é o termo de deslocamento previamente estimados. Considerando o centro óptico como origem, o ponto reconstruído pode ser escrito como $\mathbf{X}(t) = t \mathbf{d}$. Substituindo na equação do plano, obtém-se:

$$t = -\frac{d}{\mathbf{n}^\top \mathbf{d}}, \quad \mathbf{X} = t \mathbf{d}.$$

A condição $\mathbf{n}^\top \mathbf{d} \neq 0$ evita configurações em que o raio seja paralelo ao plano, e $t > 0$ garante que o ponto reconstruído esteja à frente da câmera. Esse procedimento é aplicado a todos os pontos do perfil, resultando em uma curva 3D correspondente à seção iluminada do objeto naquele quadro. Além disso, esse método pode ser implementado com mostra o trecho contido no Quadro 3.

Quadro 3 – Triangulação por interseção entre raio de visão e plano do laser

```

1 # Corrige distorcao e obtem coordenadas normalizadas
2 pt = np.array([[u_sub, v]], dtype=np.float64)
3 pt_norm = cv2.undistortPoints(
4     pt, K, distCoeffs
5 ) [0, 0]
6 # Vetor diretor do raio no sistema da camera
7 d = np.array([pt_norm[0], pt_norm[1], 1.0])
8 # Calcula parametro de intersecao raio-plano
9 den = np.dot(plane_normal, d)
10 if abs(den) < eps:
11     return None
12
13 t = -plane_offset / den
14 # Coordenada tridimensional reconstruida
15 X = t * d

```

Fonte: Desenvolvido pelo autor.

3.7.5 Transformações de Referencial e Composição Global da Nuvem

Os pontos tridimensionais obtidos por triangulação a partir de cada quadro do vídeo estão inicialmente expressos no sistema de coordenadas da câmera. Para que as diferentes seções reconstruídas possam ser combinadas em uma única nuvem de pontos coerente, é necessário transformá-las para um sistema de referência comum associado à mesa rotativa e aplicar a rotação correspondente a cada posição angular do objeto.

A transformação do referencial da câmera para o referencial da mesa é realizada utilizando os parâmetros estimados durante a calibração geométrica do sistema. Esses

parâmetros definem uma transformação rígida composta por uma matriz de rotação \mathbf{R}_{ref} e um vetor de translação \mathbf{t}_{ref} , que relacionam o sistema da câmera ao sistema da mesa. Assim, um ponto reconstruído \mathbf{X}_c no referencial da câmera é transformado para o referencial da mesa segundo:

$$\mathbf{X}_m = \mathbf{R}_{\text{ref}} \mathbf{X}_c + \mathbf{t}_{\text{ref}}. \quad (21)$$

Essa transformação garante que todos os pontos reconstruídos estejam corretamente posicionados em relação ao plano da mesa e ao eixo de rotação previamente estimados.

3.7.5.1 Rotação Incremental e Empilhamento das Seções

Uma vez expressos no sistema de coordenadas global da mesa, os pontos reconstruídos a partir de cada quadro são rotacionados em torno do eixo Z de acordo com o ângulo associado à posição da mesa no instante da captura. Considerando um escaneamento composto por n incrementos angulares igualmente espaçados, o passo angular é definido por:

$$\Delta\theta = \frac{2\pi}{n}, \quad n = 1600. \quad (22)$$

Para o quadro de índice k , a rotação aplicada é dada por $\theta_k = k \Delta\theta$, e a matriz de rotação correspondente é:

$$\mathbf{R}_z(\theta_k) = \begin{bmatrix} \cos \theta_k & -\text{sen} \theta_k & 0 \\ \text{sen} \theta_k & \cos \theta_k & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (23)$$

Assim, cada ponto \mathbf{X}_m é transformado para sua posição final na nuvem global por:

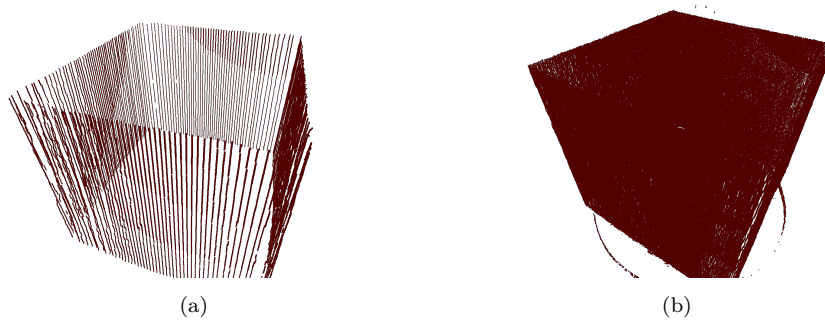
$$\mathbf{X}_f = \mathbf{R}_z(\theta_k) \mathbf{X}_m. \quad (24)$$

Os pontos transformados de cada quadro são então acumulados em uma estrutura global de dados, formando progressivamente a nuvem de pontos tridimensional do objeto. Esse procedimento é repetido para todos os quadros do vídeo, resultando em uma representação densa e coerente da geometria do objeto escaneado.

3.7.5.2 Exportação, Visualização e Persistência da Nuvem de Pontos

No sistema desenvolvido, a nuvem de pontos é representada como um conjunto de coordenadas tridimensionais no sistema de referência global da mesa, armazenadas em uma estrutura vetorial. Essa representação facilita tanto a aplicação de filtros e transformações adicionais quanto a exportação para formatos padronizados. O formato adotado para persistência é o *Polygon File Format* (PLY), com extensão `.ply`, devido à sua simplicidade, ampla compatibilidade e suporte a atributos adicionais, como cores e normais.

Figura 33 – (a) Nuvem de pontos de objeto cúbico (200 imagens). (b) Nuvem de pontos de objeto cúbico (1600 imagens).



Fonte: Desenvolvido pelo autor.

A exportação da nuvem de pontos em formato PLY permite sua visualização imediata em bibliotecas e *softwares* dedicados, como a biblioteca Open3D e o *MeshLab*. A inspeção visual desempenha papel fundamental na validação qualitativa do processo de reconstrução, possibilitando identificar artefatos, regiões com baixa densidade de pontos, ruídos residuais ou falhas decorrentes de oclusões, reflexos e má calibração.

Durante o processo de desenvolvimento, a visualização da nuvem reconstruída também é utilizada como ferramenta de depuração, permitindo avaliar o impacto de parâmetros como resolução angular, limiarização da linha do laser e correções geométricas aplicadas. Para aumentar a velocidade com que os resultados eram gerados, foi inicialmente utilizado o motor de passo sem *microstepping*, ou seja, com 200 passo por revolução. A partir de resultados ajustados e um modelo de sistema mais alinhado com o objetivo, empregar o *microstepping* foi essencial, passando por 400, 800 e por fim 1600 passos por revolução, garantindo alta densidade na nuvem de pontos, o que é requisito para uma boa reconstrução tridimensional. A seguir na Figura 33, é possível visualizar exemplos de nuvens de pontos, e como o aumento de passos por revolução aumenta significativamente a densidade de pontos, tornando a nuvem cada vez mais adequada para reconstrução tridimensional com malhas triangulares ou poligonais. Na Figura 33 (a) é possível observar a nuvem de pontos de um objeto cúbico obtida através de 200 imagens, contendo exatamente 46.887 pontos. Por outro lado na Figura 33 (b), é possível ver o mesmo objeto submetido a 1600 imagens, formando uma nuvem com 565.415 pontos.

A persistência dos dados em arquivos externos garante a reprodutibilidade dos resultados e possibilita que as etapas de pós-processamento e geração de malhas sejam realizadas de forma independente da aquisição. Dessa forma, o sistema separa claramente as fases de captura, reconstrução e refinamento geométrico, favorecendo a modularidade e a extensibilidade do projeto.

3.8 PÓS-PROCESSAMENTO E GERAÇÃO DE MALHAS

A nuvem de pontos obtida ao final do processo de reconstrução tridimensional representa uma amostragem discreta da superfície do objeto escaneado. Embora essa representação seja adequada para inspeção visual e análises preliminares, muitas aplicações práticas — como manufatura aditiva, engenharia reversa e simulação — exigem uma representação contínua da superfície na forma de uma malha poligonal.

A geração de malhas a partir de nuvens de pontos envolve uma sequência de etapas de processamento geométrico destinadas a reduzir ruídos, estimar propriedades diferenciais da superfície e reconstruir uma representação topologicamente consistente. No sistema desenvolvido, esse processo é realizado por meio de um algoritmo composto pelas seguintes etapas: (i) suavização da nuvem de pontos, (ii) estimação das normais, (iii) reconstrução de superfície pelo método de Poisson (Kazhdan; Bolitho; Hoppe, 2006), (iv) limpeza topológica da malha e (v) suavização final para melhoria da qualidade geométrica.

Cada uma dessas etapas desempenha papel fundamental na qualidade do modelo final. A suavização inicial reduz irregularidades provenientes do processo de triangulação, a estimação de normais fornece informação direcional essencial para a reconstrução, enquanto o método de Poisson permite obter uma superfície fechada e contínua mesmo na presença de pequenas lacunas na nuvem. As etapas finais de limpeza e suavização visam eliminar artefatos topológicos e melhorar o aspecto visual do modelo, sem comprometer significativamente sua geometria global.

3.8.1 Pós-processamento da Nuvem de Pontos e Geração da Malha

A etapa de pós-processamento tem como objetivo converter a nuvem de pontos tridimensional reconstruída em uma representação contínua da superfície do objeto, na forma de uma malha poligonal adequada para visualização, análise geométrica e aplicações práticas, como manufatura aditiva e engenharia reversa. Para isso, são aplicadas uma sequência de processamentos geométricos compostos por etapas consolidadas na literatura e amplamente suportadas por bibliotecas especializadas, como a Open3D.

Inicialmente, realiza-se a suavização da nuvem de pontos com o objetivo de reduzir ruídos locais introduzidos durante o processo de aquisição e triangulação. Essa etapa promove uma regularização espacial da nuvem, atenuando irregularidades de alta frequência sem alterar de forma significativa a geometria global do objeto. No sistema desenvolvido, essa suavização é realizada por meio de um método baseado em *Moving Least Squares* (MLS), cuja implementação está disponível na biblioteca Open3D.

Em seguida, procede-se à estimação das normais da superfície associadas a cada ponto da nuvem suavizada. As normais fornecem informação direcional local essencial para os algoritmos de reconstrução de superfície. Essa etapa é realizada considerando vizinhanças espaciais locais, obtidas de forma eficiente por meio de estruturas de busca

espacial, como *k-d trees*, amplamente utilizadas em áreas como visão computacional e *machine learning*. O resultado esperado dessa etapa é a obtenção de um campo de normais consistente, alinhado com a geometria local da superfície do objeto.

Com a nuvem de pontos suavizada e orientada, aplica-se a reconstrução da superfície pelo método implícito de Poisson, utilizando a função da biblioteca Open3D. Esse método permite a geração de uma malha contínua e fechada a partir da nuvem de pontos orientada, mesmo na presença de pequenas lacunas ou irregularidades na amostragem. Como resultado, obtém-se uma malha inicial que representa de forma global a superfície do objeto escaneado.

Para a reconstrução da superfície, utilizou-se o algoritmo de Poisson com a profundidade da *octree* configurada em $d = 10$. A escolha desse parâmetro baseou-se no compromisso entre a preservação de detalhes geométricos e a filtragem de ruídos remanescentes da aquisição. Observou-se que valores reduzidos de profundidade resultavam em superfícies excessivamente suavizadas, com perda de definição de arestas e vértices, enquanto valores elevados aumentavam significativamente o custo computacional e a sensibilidade a ruídos, sem ganhos proporcionais na fidelidade geométrica. Assim, o valor adotado mostrou-se adequado para representar de forma consistente a geometria global dos objetos escaneados, preservando características relevantes, como arestas do cubo e da pirâmide, conforme recomendado na literatura clássica sobre reconstrução por Poisson (Kazhdan; Bolitho; Hoppe, 2006).

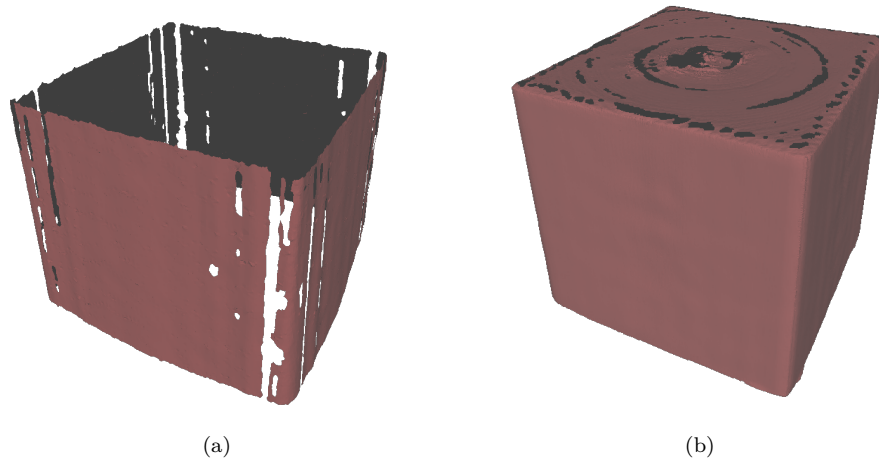
Devido ao caráter global do método de Poisson, a malha reconstruída pode conter regiões pouco suportadas pelos dados ou pequenos artefatos topológicos. Assim, realiza-se uma etapa de limpeza topológica, na qual componentes desconectados, superfícies disformes e faces degeneradas são removidas com base em critérios de conectividade e tamanho. Essa etapa visa garantir maior coerência topológica e estabilidade geométrica da malha resultante.

Por fim, aplica-se uma etapa de suavização diretamente sobre a malha poligonal, com o objetivo de melhorar sua qualidade geométrica e aspecto visual. No sistema desenvolvido, utiliza-se o filtro de Taubin, disponível na Open3D, que permite suavizar irregularidades locais sem provocar o encolhimento global da geometria. Essa etapa final resulta em uma malha mais uniforme, visualmente consistente e adequada para exportação, análise e aplicações práticas (Taubin, 1995).

3.8.2 Exportação do Modelo Final

Concluídas as etapas de reconstrução, limpeza e suavização, o modelo tridimensional final é exportado para arquivos externos, permitindo sua utilização em ferramentas de visualização, edição e manufatura. A malha é armazenada em formatos amplamente utilizados, como PLY ou STL, garantindo compatibilidade com softwares de modelagem, análise e impressão 3D.

Figura 34 – (a) Malha 3D de objeto cúbico (200 imagens). (b) Malha 3D de objeto cúbico (1600 imagens).



Fonte: Desenvolvido pelo autor.

A exportação do modelo final marca o encerramento do fluxo de operações do sistema, consolidando os dados adquiridos e processados em uma representação geométrica contínua do objeto escaneado. A separação entre as etapas de aquisição, reconstrução da nuvem de pontos e geração de malhas favorece a modularidade do sistema e permite que ajustes e melhorias sejam realizados de forma independente em cada estágio. Em seguida na Figura 34 é possível comparar e evidenciar a importância da densidade da nuvem de pontos na reconstrução do objeto. Na Figura 34 (a) o algoritmo gerou a malha a partir da nuvem de pontos menos densa (formada por 200 imagens do objeto), apresentando alguns buracos que não puderam ser preenchidos por falta de informação. Já ao lado na Figura 34 (b), visualiza-se uma malha contínua, construída a partir da nuvem de pontos mais densa (formada por 1600 imagens do objeto).

4 RESULTADOS E DISCUSSÕES

Este capítulo apresenta e discute os resultados obtidos a partir da implementação do sistema de escaneamento tridimensional proposto. O objetivo é avaliar o desempenho do sistema sob diferentes condições experimentais, bem como analisar o impacto das escolhas de projeto e das etapas de calibração na qualidade final da reconstrução 3D.

Os experimentos realizados podem ser agrupados em três categorias principais. Inicialmente, são avaliados os resultados da calibração geométrica da câmera, com ênfase na análise da estabilidade dos parâmetros estimados em função do número de imagens utilizadas no processo de calibração. Em seguida, são analisados cenários de sensibilidade do sistema, incluindo a avaliação dos efeitos decorrentes de estimativas geométricas imprecisas. Por fim, são apresentados resultados de reconstrução tridimensional de diferentes objetos, considerando variações na resolução angular do escaneamento, expressa pelo número de passos da mesa rotativa por revolução.

Durante os experimentos, os parâmetros intrínsecos da câmera, o plano da mesa e o plano do laser foram mantidos fixos após a calibração, salvo nos casos em que alterações controladas foram introduzidas propositalmente para análise de sensibilidade. A aquisição dos dados foi realizada sob condições semelhantes de iluminação e posicionamento, de modo a garantir a comparabilidade entre os resultados obtidos.

As métricas utilizadas para avaliação incluem o erro médio de reprojeção considerando a calibração da câmera em conjunto com a matriz de distorção da lente, o número de pontos da nuvem reconstruída, o número de faces da malha gerada e a análise visual da preservação geométrica dos objetos escaneados. Sempre que pertinente, os resultados numéricos são acompanhados de visualizações tridimensionais que auxiliam na interpretação qualitativa do desempenho do sistema.

4.1 VALIDAÇÃO DA CALIBRAÇÃO GEOMÉTRICA

A calibração geométrica da câmera constitui a base para todas as etapas subsequentes do sistema de escaneamento tridimensional. Dessa forma, antes de avaliar os resultados de reconstrução, torna-se necessário analisar quantitativamente a qualidade dos parâmetros intrínsecos estimados, bem como a consistência do modelo de projeção adotado.

4.1.1 Resultados Numéricos da Calibração da Câmera

Os parâmetros intrínsecos da câmera foram estimados a partir do conjunto de imagens de calibração utilizando o método de calibração planar proposto por Zhang, disponível e implementado no OpenCV. Como métrica principal de qualidade do ajuste, utilizou-se o erro médio de reprojeção, definido como a distância média, em *pixels*, entre

os pontos observados na imagem e os pontos projetados pelo modelo da câmera a partir das coordenadas conhecidas do padrão ChArUco.

O erro médio de reprojeção obtido para a calibração final foi de aproximadamente $\varepsilon_{\text{rms}} = 0.4866$ *pixels*, valor compatível com sistemas de visão computacional baseados em câmeras convencionais e adequado para aplicações de reconstrução tridimensional por triangulação. Valores dessa ordem indicam que o modelo da câmera representa de forma consistente o processo de formação da imagem, fornecendo uma base confiável para as etapas de estimação geométrica subsequentes.

Contudo, os resultados apresentados na Tabela 1 evidenciam que o erro médio de reprojeção não apresenta comportamento monotônico com o aumento do número de imagens utilizadas na calibração. Observa-se, inclusive, que calibrações realizadas com um número reduzido de imagens podem resultar em valores baixos de erro de reprojeção, sem que isso implique necessariamente em um modelo geométrico confiável, uma vez que a minimização dessa métrica pode ocorrer mesmo na presença de parâmetros intrínsecos fisicamente inconsistentes, conforme discutido por Jiang e Tao (Jiang; Tao, 2017).

Como métrica complementar de robustez, foi realizada uma análise de consistência dos parâmetros intrínsecos estimados a partir de múltiplas calibrações independentes, variando o número de imagens utilizadas. Em particular, avaliou-se a estabilidade da distância focal horizontal f_x , uma vez que este parâmetro exerce influência direta na etapa de triangulação. Os resultados indicam que, a partir de aproximadamente 15 imagens, a variação relativa de f_x permanece próxima ou inferior a 0,1%, evidenciando alta consistência entre as calibrações realizadas.

Na Equação (25) e Equação (26) é possível visualizar o método utilizado para determinar a métrica de comparação usada na Tabela 1, onde f_x é a distância focal calculada para cada execução e f_{xr} representa a distância focal de referência calculada para a execução com 42 imagens.

$$\|\mathbf{K}_r\| = \sqrt{k_1^2 + k_2^2 + k_3^2}. \quad (25)$$

$$\Delta f_x / f_{xr} = \frac{|f_x - f_{xr}|}{f_{xr}} \times 100\% \quad (26)$$

Tabela 1 – Influência do número de imagens na calibração da câmera.

Nº de imagens	ε_{rms} (px)	$\ \mathbf{k}_r\ $	$\Delta f_x / f_{xr}$ (%)
1	0,2955	66,80	13,53
5	0,3451	196,30	1,16
15	0,3981	2,24	0,14
30	0,4866	0,51	0,09
42	0,5450	0,52	0,00

Fonte: Autor.

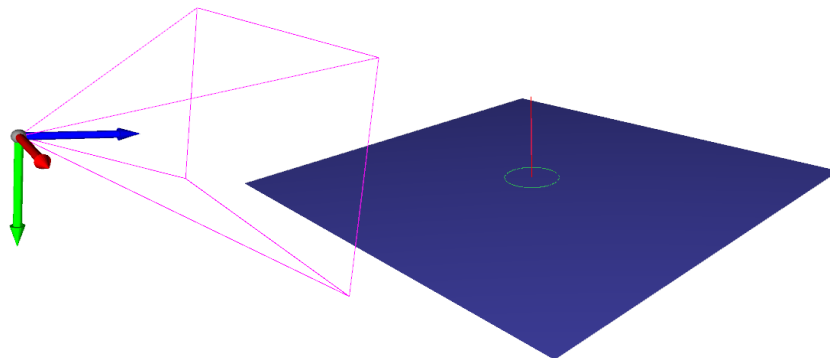
Por outro lado, a magnitude dos coeficientes de distorção radial $\|\mathbf{k}_r\|$ e a variação relativa dos parâmetros intrínsecos $\Delta f_x/f_{xr}$ revelam um comportamento distinto. Para calibrações realizadas com poucas imagens, os coeficientes de distorção assumem valores elevados e fisicamente pouco plausíveis, indicando sobreajuste do modelo aos dados disponíveis. À medida que o número de imagens aumenta, observa-se uma convergência progressiva tanto da distorção quanto dos parâmetros intrínsecos, com variações residuais inferiores a 0,1% a partir de aproximadamente 15 imagens, indicando que, embora o erro de projeção seja uma métrica importante, ele não deve ser utilizado de forma isolada para avaliar a qualidade da calibração. A estabilidade e a consistência dos parâmetros estimados mostram-se mais representativas da robustez do modelo geométrico, especialmente em aplicações de reconstrução tridimensional baseadas em triangulação.

4.2 VISUALIZAÇÃO DA CALIBRAÇÃO COMPLETA DO SISTEMA

A fim de validar de forma qualitativa a coerência geométrica das estimativas realizadas, foi desenvolvida uma visualização tridimensional do sistema completo no referencial da câmera. Essa visualização integra, em um único ambiente virtual, os principais elementos geométricos estimados durante as etapas de calibração: a posição e orientação da câmera, o plano da mesa rotativa, o eixo de rotação e o plano do laser.

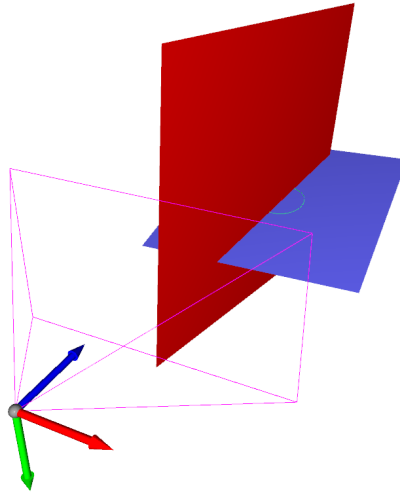
A Figura 35 apresenta a visualização tridimensional do sistema calibrado. O plano da mesa é representado como a superfície azul e plana no espaço, enquanto o eixo de rotação é indicado por um vetor vermelho alinhado aproximadamente à direção vertical. O plano do laser é exibido na Figura 36 como uma superfície inclinada que intersecta o volume de trabalho do sistema, evidenciando sua posição relativa em relação à câmera e à mesa.

Figura 35 – Visualização 3D da calibração completa do sistema (câmera, plano da mesa rotativa e eixo de rotação).



Fonte: Desenvolvido pelo autor.

Figura 36 – Visualização 3D da calibração completa do sistema (câmera, plano da mesa rotativa e plano do laser).



Fonte: Desenvolvido pelo autor.

Essa visualização integrada fornece boa validação das estimativas realizadas, pois permite verificar de forma intuitiva a coerência espacial entre os diferentes elementos do sistema e comparar com o mundo físico. Além disso, a visualização criada facilita a identificação de possíveis inconsistências geométricas que poderiam comprometer o processo de triangulação, como desalinhamentos excessivos ou interseções incorretas entre os planos estimados.

4.3 RESULTADOS DA RECONSTRUÇÃO TRIDIMENSIONAL

Nesta seção são apresentados os resultados obtidos a partir da reconstrução tridimensional de diferentes objetos utilizando o sistema desenvolvido. O objetivo é avaliar o desempenho do *scanner* em termos de densidade de pontos, complexidade da malha gerada e preservação da geometria do objeto, considerando variações na resolução angular do escaneamento causadas por falhas de estimação.

4.3.1 Objetos Avaliados e Condições de Aquisição

Para a avaliação do sistema, foram selecionados objetos com diferentes características geométricas, de modo a explorar distintos desafios do processo de reconstrução. Os objetos incluem peças com geometrias regulares e dimensões conhecidas (como mostra Figura 37), utilizadas para análise de consistência e repetibilidade, bem como objetos com formas mais complexas, empregados para avaliação qualitativa da preservação morfológica.

Durante os experimentos, os objetos foram posicionados sobre a mesa rotativa, aproximadamente centrados em relação ao eixo de rotação estimado. A câmera e o emissor

Figura 37 – Objetos utilizados para validação do *scanner*.

Fonte: Desenvolvido pelo autor.

de laser foram mantidos fixos, utilizando os parâmetros de calibração previamente obtidos. As condições de iluminação ambiente permaneceram constantes ao longo das aquisições, adotando-se a configuração de parâmetros fixos da câmera descrita no Capítulo 3.

4.3.2 Análise e Comparação Final

A avaliação qualitativa da reconstrução tridimensional foi realizada por meio da inspeção visual das nuvens de pontos e das malhas geradas para os objetos escaneados. Essa análise tem como objetivo verificar a coerência geométrica global, a preservação da forma do objeto e a presença de artefatos típicos do processo de aquisição por luz estruturada. Também foi utilizada a análise dimensional de objetos com dimensão conhecida, usando o *software* MeshLab para medir arestas nas malhas geradas. Por fim foi feita a análise de densidade da nuvem de pontos e sua relação com a qualidade e o número de faces geradas na reconstrução 3D, permitindo empiricamente determinar o limite físico de resolução do sistema.

4.3.2.1 Objeto I - Cubo

A Figura 38 mostra o primeiro objeto elaborado para validação do sistema. Trata-se de um cubo de dimensão nominal de $100 \times 100 \times 100 \text{ mm}$, desenvolvido no *software* SolidWorks e impresso em impressora 3D. Em seguida na Figura 39 (a) é possível observar a nuvem de pontos obtida e na Figura 39 (b) a malha gerada a partir dela.

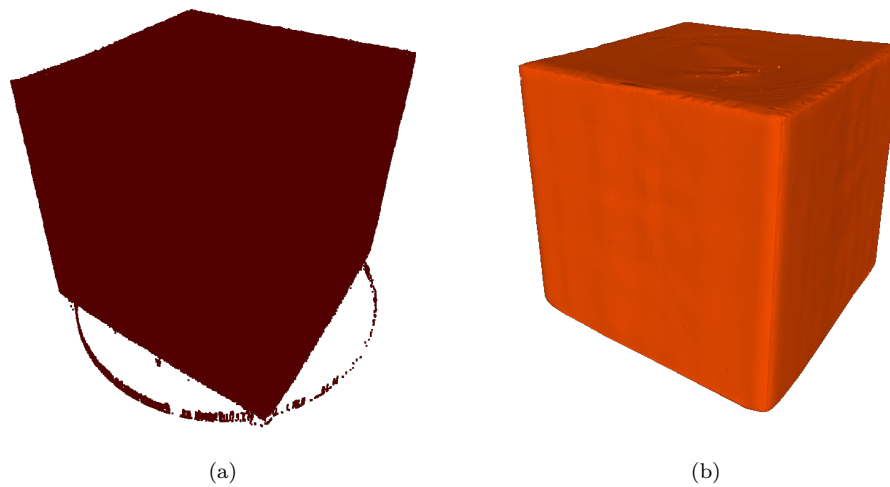
Por conseguinte, a Figura 40 apresenta a reconstrução tridimensional do cubo com a aplicação da ferramenta de medição do *software*, utilizada para a obtenção das cotas diretamente sobre o modelo reconstruído. Essas medições permitem validar as dimensões estimadas pelo *scanner* por meio da comparação com as dimensões reais do objeto, determinadas experimentalmente com auxílio de um paquímetro.

Para a análise dimensional, considera-se o valor médio das arestas ao longo dos eixos x , y e z , expressos por $L_x = (100.6 \pm 0.2) \text{ mm}$, $L_y = (100.5 \pm 0.1) \text{ mm}$ e $L_z =$

Figura 38 – Cubo elaborado em *SolidWorks* e impresso em 3D.

Fonte: Desenvolvido pelo autor.

Figura 39 – (a) Nuvem de pontos do Cubo. (b) Reconstrução 3D do Cubo.

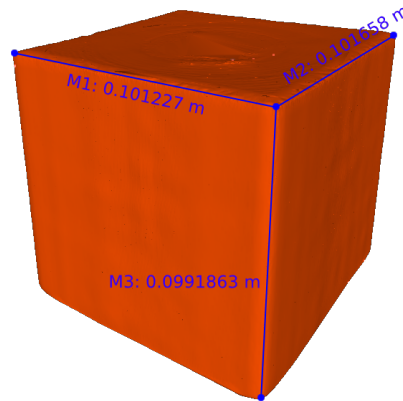


Fonte: Desenvolvido pelo autor.

(100.6 ± 0.3) mm. A partir desses valores de referência, é apresentada na Tabela 2 a relação entre as dimensões reais e aquelas estimadas pelo sistema, bem como os respectivos erros absoluto (em milímetros) e percentual associados à reconstrução tridimensional da primeira amostra.

Além da comparação pontual apresentada, realizou-se uma análise de repetibilidade do sistema a partir de múltiplas reconstruções independentes do mesmo objeto, conforme explicitado na Tabela 3. Essa abordagem permite avaliar não apenas a precisão média das dimensões estimadas, mas também a variabilidade dos erros associados ao processo de aquisição, calibração e reconstrução tridimensional.

Figura 40 – Malha do cubo com cotas de validação de dimensões.



Fonte: Desenvolvido pelo autor.

Tabela 2 – Comparação entre dimensões reais e reconstruídas do cubo em milímetros

Dimensão	Referência	Estimadas	Erro abs.	Erro (%)
L_x	100.6 ± 0.2	101.227	0.627	0.62
L_y	100.5 ± 0.1	101.658	1.158	1.15
L_z	100.6 ± 0.3	99.186	1.414	1.41

Fonte: Desenvolvido pelo autor.

Tabela 3 – Erros absolutos (mm) em cinco reconstruções independentes do cubo.

Reconstrução	$ e_x $ (mm)	$ e_y $ (mm)	$ e_z $ (mm)
1	0.627	1.158	1.414
2	0.588	1.203	1.350
3	0.659	1.104	1.482
4	0.600	1.183	1.392
5	0.634	1.128	1.446

Fonte: Desenvolvido pelo autor.

A análise estatística dos erros absolutos indica que o sistema apresenta boa repetibilidade nas reconstruções realizadas. Os valores de erro médio absoluto (*Mean Absolute Error* – MAE), calculados individualmente para cada eixo, permaneceram abaixo de 1.5 mm em todas as direções, enquanto os coeficientes de variação (*CV*) situaram-se abaixo de 4,4%, evidenciando baixa variabilidade entre as reconstruções independentes, conforme mostra Tabela 4.

Observa-se ainda que, embora os erros médios apresentem diferenças entre os eixos, essas variações mantêm-se consistentes e controladas, não comprometendo a preservação das características geométricas do objeto. Esses resultados demonstram que o sistema é capaz de reconstruir o cubo com precisão dimensional adequada e comportamento estável, preservando as dimensões do objeto com erro inferior a 1,5%, mesmo considerando as

Tabela 4 – Estatísticas descritivas dos erros absolutos por eixo para o cubo.

Eixo	MAE (mm)	σ (mm)	CV (%)
<i>x</i>	0.622	0.027	4.34
<i>y</i>	1.155	0.038	3.29
<i>z</i>	1.417	0.049	3.46

Fonte: Desenvolvido pelo autor.

incertezas inerentes ao processo de aquisição, reconstrução tridimensional e múltiplas execuções.

Por fim, foi elaborada uma comparação que justifica o número de imagens que seria usado para a geração das nuvens de pontos e, posteriormente, das malhas tridimensionais. Na Tabela 5 é possível visualizar a relação entre o número de imagens obtidas do objeto e a quantidade de pontos na nuvem e a quantidade de faces geradas na malha. Esses dados indicam que, ao aumentar a densidade da malha, não só há uma maior preservação de detalhes do objeto, como também uma mitigação de perda de informação, que ocorre quando a distância entre a interseção do laser no objeto em duas imagens subsequentes é maior que a resolução do sistema, causando buracos na malha (ocorre quando a distância é muito maior que a resolução) ou ocultando detalhes que não são escaneados e acabam sendo reconstruído como superfícies planas (ocorre quando a distância é pouco maior que a resolução).

Tabela 5 – Objeto: cubo. Influência do número de imagens na geração da nuvem de pontos e malha 3D.

Nº de imagens	Nº pontos	nº Faces
200	46.887	234.504
400	101.126	477.901
1600	422.926	1.312.068

Fonte: Desenvolvido pelo autor.

4.3.2.2 Objeto II - Pirâmide

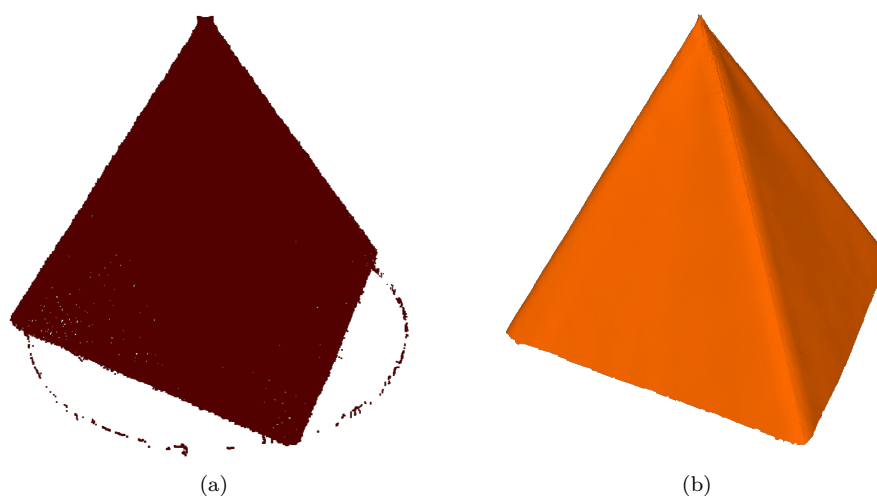
A Figura 41 mostra o segundo objeto elaborado para validação do sistema. Trata-se de uma pirâmide de dimensão nominal $100 \times 100 \times 122,35$ mm, desenvolvida no *software* SolidWorks e produzida através de impressão 3D. Em seguida, na Figura 42 (a) é possível observar a nuvem de pontos obtida e na Figura 42 (b) a malha gerada a partir dela.

A seguir, assim como foi feito para o objeto anterior, a Figura 43 mostra a reconstrução da Pirâmide. Também com auxílio do *software*, foi utilizada a ferramenta de régua para gerar as cotas, validando as dimensões reais do objeto. Para essas medidas considera-se o valor médio das arestas da base e das arestas diagonais, representados por $L_b = (99.625 \pm 0.15)$ mm e $L_d = (119.25 \pm 0.2)$ mm. Dessa forma é possível observar a

Figura 41 – Pirâmide elaborada em *SolidWorks* e impressa em 3D.

Fonte: Desenvolvido pelo autor.

Figura 42 – (a) Nuvem de pontos da Pirâmide. (b) Reconstrução 3D da Pirâmide.

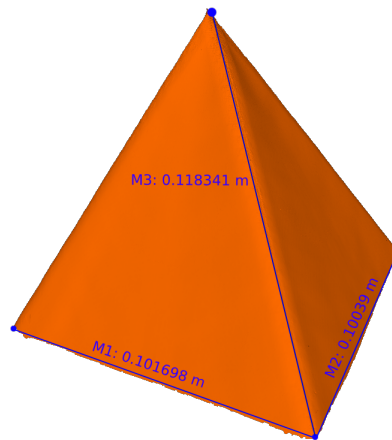


Fonte: Desenvolvido pelo autor.

Tabela 6 com a relação dos erros medidos entre o objeto real e o estimado pelo *scanner* (em milímetros), considerando a média das medidas estimadas da base e das diagonais da primeira amostra.

Além da comparação pontual apresentada, realizou-se uma análise de repetibilidade do sistema a partir de múltiplas reconstruções independentes da pirâmide, conforme apresentado na Tabela 7. Essa análise permite avaliar não apenas a precisão média das dimensões estimadas, mas também a variabilidade dos erros associados ao processo de aquisição e reconstrução tridimensional, especialmente em um objeto com geometria mais complexa e presença de faces inclinadas.

Figura 43 – Malha da pirâmide com cotas de validação de dimensões.



Fonte: Desenvolvido pelo autor.

Tabela 6 – Comparação entre dimensões reais e reconstruídas da pirâmide em milímetros

Dimensão	Referência	Estimadas	Erro abs.	Erro (%)
L_b	99.625 ± 0.15	101.04	1.42	1.42
L_d	119.25 ± 0.2	118.34	0.91	0.57

Fonte: Desenvolvido pelo autor.

Tabela 7 – Erros absolutos (mm) em cinco reconstruções independentes da pirâmide.

Reconstrução	$ e_b $ (mm)	$ e_d $ (mm)
1	1.42	0.91
2	1.38	0.95
3	1.46	0.88
4	1.40	0.93
5	1.43	0.90

Fonte: Desenvolvido pelo autor.

A análise estatística dos erros absolutos indica que o sistema apresenta boa reprodutibilidade também para a reconstrução da pirâmide. Os valores de erro médio absoluto (MAE), calculados individualmente para as dimensões analisadas, permaneceram abaixo de 1.5 mm, enquanto os coeficientes de variação situaram-se abaixo de 3.5%, evidenciando baixa variabilidade entre as reconstruções independentes, conforme mostra Tabela 8.

Observa-se que os erros associados às arestas da base são ligeiramente superiores aos das arestas diagonais, comportamento esperado devido à maior extensão das faces inclinadas e à influência da geometria do objeto no processo de triangulação. Ainda assim, os resultados demonstram que o sistema é capaz de preservar adequadamente as características geométricas da pirâmide, mantendo precisão dimensional compatível com aplicações de validação geométrica e prototipagem.

Tabela 8 – Estatísticas descritivas dos erros absolutos para a pirâmide.

Dimensão	MAE (mm)	σ (mm)	CV (%)
Base (L_b)	1.42	0.03	2.1
Diagonal (L_d)	0.91	0.03	3.1

Fonte: Desenvolvido pelo autor.

Por fim, para poder comparar a relação do número de imagens obtidas pelo *scanner* com a qualidade da reconstrução, foi elaborada a Tabela 9, onde é possível visualizar a quantidade de pontos na nuvem e a quantidade de faces geradas na malha. Esses dados estão coerentes com a análise feita com o cubo, indicando que ao aumentar a densidade da malha, não só há uma maior preservação de detalhes do objeto, como também mitigação da perda de informação.

Tabela 9 – Objeto: pirâmide. Influência do número de imagens na geração da nuvem de pontos e malha 3D.

Nº de imagens	Nº pontos	nº Faces
200	49.077	282.335
400	99.349	565.794
1600	398.965	1.571.582

Fonte: Autor.

4.3.2.3 Objeto III - Ampulheta

O próximo objeto, com formato semelhante ao de uma ampulheta (conforme Figura 44), foi elaborado com a finalidade de validar a preservação de características esféricas, diferentemente dos objetos anteriores. Na Figura 45 (a) é possível observar a nuvem de pontos obtida para esse objeto, onde nota-se que se mantém boa a densidade da nuvem de pontos. Já na Figura 45 (b) é possível verificar a malha resultante, apresentando ótima precisão as características morfológicas do objeto.

Abaixo, na Tabela 10, é possível efetuar a mesma análise dos objetos anteriores, trazendo um paralelo entre a quantidade de imagens usadas pelo *scanner* e a densidade da nuvem de pontos e portanto o número de faces do objeto reconstruído.

Tabela 10 – Objeto: ampulheta. Influência do número de imagens na geração da nuvem de pontos e malha 3D.

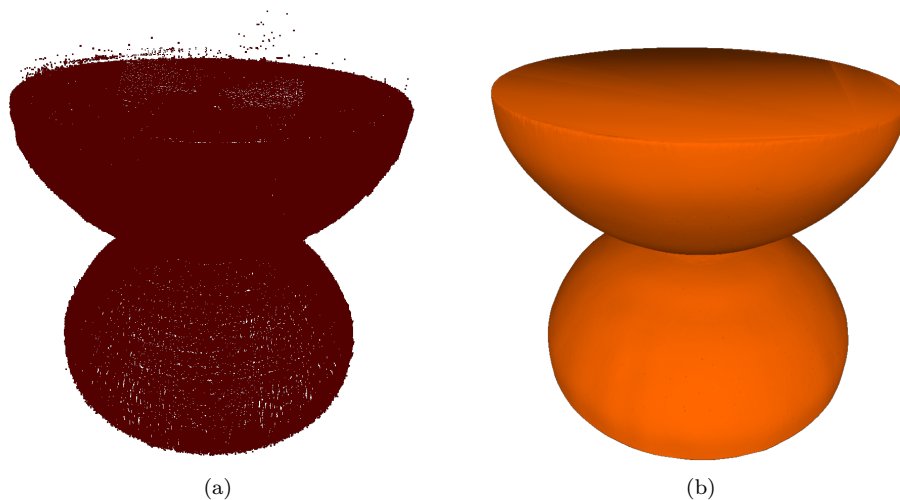
Nº de imagens	Nº pontos	nº Faces
200	42.228	273.934
400	85.889	547.779
1600	352.298	1.522.147

Fonte: Autor.

Figura 44 – Ampulheta elaborada em *SolidWorks* e impressa em 3D.

Fonte: Desenvolvido pelo autor.

Figura 45 – (a) Nuvem de pontos da Ampulheta. (b) Reconstrução 3D da Ampulheta.



(a)

(b)

Fonte: Desenvolvido pelo autor.

4.3.2.4 Objeto IV - Abajur

Na sequência, optou-se pela utilização de um objeto que apresentasse uma combinação de características geométricas distintas — regiões retilíneas com ângulos retos e superfícies curvas com textura — conforme ilustrado na Figura 46. O objeto selecionado é composto por uma base de madeira com geometria predominantemente prismática e, sobre ela, uma esfera com superfície levemente rugosa.

Na Figura 47 (a) é apresentada a nuvem de pontos obtida a partir do processo de digitalização. Já na Figura 47 (b) observa-se a reconstrução tridimensional correspondente.

A análise da malha reconstruída evidencia a presença de regiões com oclusão, especialmente na base do objeto, onde a geometria não pôde ser completamente observada durante a aquisição, resultando em lacunas na reconstrução.

Além disso, verifica-se que irregularidades de superfície com dimensões abaixo da resolução espacial do *scanner* não foram preservadas na malha final, ao passo que rugosidades mais pronunciadas foram parcialmente reconstruídas. Esse comportamento é consistente com as limitações inerentes à resolução do sistema de aquisição e ao método de reconstrução empregado.

Figura 46 – Abajur esférico.



Fonte: Desenvolvido pelo autor.

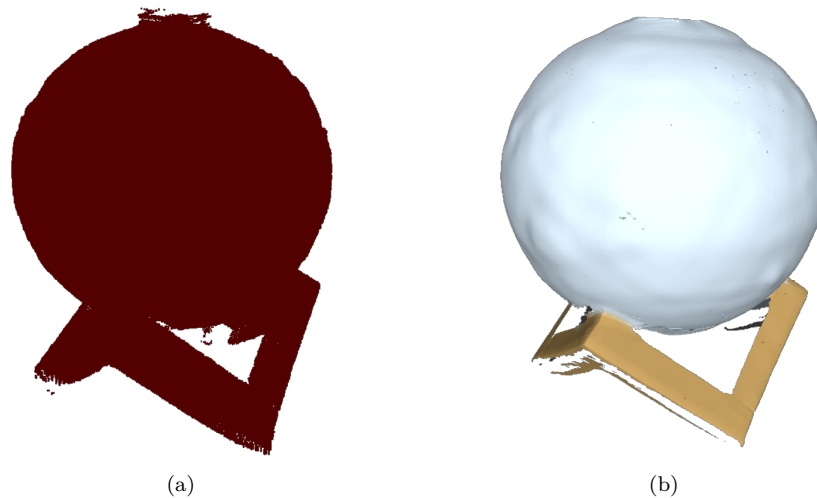
Abaixo, na Tabela 11, é possível visualizar a mesma relação de quantidade de imagens obtidas no escaneamento com a densidade da nuvem de pontos e quantidade de faces geradas a partir dela.

Tabela 11 – Objeto: abajur. Influência do número de imagens na geração da nuvem de pontos e malha 3D.

Nº de imagens	Nº pontos	nº Faces
200	66.429	252.139
400	127.385	499.232
1600	554.335	1.428.285

Fonte: Autor.

Figura 47 – (a) Nuvem de pontos do Abajur. (b) Reconstrução 3D do Abajur.



Fonte: Desenvolvido pelo autor.

4.3.2.5 Objeto V - Cantil

Na sequência foi utilizado um cantil para continuar com as validações do *scanner*, conforme Figura 49. Na Figura 47 (a) é possível observar a nuvem de pontos obtida e ao lado na Figura 49 (b) a reconstrução 3D obtida. A análise dos resultados indica que o material do objeto exerce influência direta na qualidade da reconstrução e na preservação de detalhes geométricos.

No caso do cantil, observa-se que regiões como a tampa e a área próxima ao gargalo apresentaram maior perda de detalhes na reconstrução tridimensional. Esse comportamento está associado à geometria dessas regiões, que combina superfícies com baixo contraste geométrico e acabamento relativamente uniforme, dificultando a correta detecção e triangulação do feixe laser. Em contrapartida, as regiões de curvatura mais suave e contínua do corpo do objeto foram reconstruídas de forma mais consistente, refletindo a adequação do sistema para superfícies principalmente com variação geométrica gradual.

Na Tabela 12 apresenta-se a relação entre a quantidade de imagens adquiridas durante o escaneamento e a densidade da nuvem de pontos, bem como o número de faces geradas na reconstrução da malha.

Tabela 12 – Objeto: cantil. Influência do número de imagens na geração da nuvem de pontos e malha 3D.

Nº de imagens	Nº pontos	nº Faces
200	100.691	526.279
400	193.478	1.074.343
1600	743.217	2.830.479

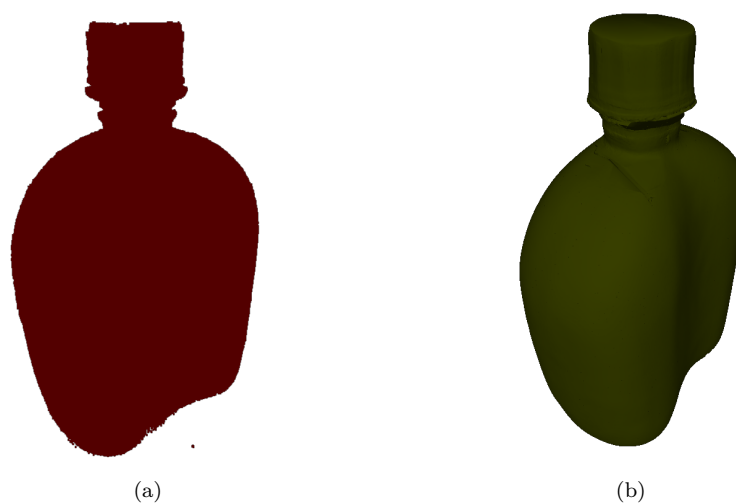
Fonte: Autor.

Figura 48 – Cantil.



Fonte: Desenvolvido pelo autor.

Figura 49 – (a) Nuvem de pontos do Cantil. (b) Reconstrução 3D do Cantil.



Fonte: Desenvolvido pelo autor.

4.4 DISCUSSÃO, SUGESTÕES E MELHORIAS

Os resultados apresentados neste capítulo demonstram que o sistema de escaneamento tridimensional desenvolvido é capaz de reconstruir, de forma consistente, a geometria de objetos a partir da técnica de luz estruturada baseada em linha de laser. A integração entre a calibração geométrica, o controle da mesa rotativa, o processo de aquisição sincronizada e o fluxo de reconstrução permitiu obter nuvens de pontos e malhas coerentes, com boa preservação morfológica dos objetos escaneados.

A análise dos resultados de calibração evidenciou que métricas tradicionais, como

o erro médio de reprojeção não são suficientes para avaliar isoladamente a qualidade do modelo geométrico. Observou-se que calibrações realizadas com um número reduzido de imagens podem apresentar baixos valores de erro de reprojeção, mas resultam em parâmetros intrínsecos e coeficientes de distorção fisicamente incoerentes. A estabilidade dos parâmetros estimados mostrou-se uma métrica mais representativa da robustez da calibração, reforçando a importância de conjuntos de imagens suficientemente diversificados, conforme previamente discutido na Seção 4.1.1.

No que diz respeito à reconstrução tridimensional, verificou-se que o aumento da resolução angular do escaneamento, obtido por meio de um maior número de passos por revolução da mesa rotativa, resulta em nuvens de pontos mais densas e malhas com maior nível de detalhe, conforme apresentado na Seção 3.7.5.2. Contudo, esse ganho ocorre às custas de maior tempo de aquisição e aumento da complexidade computacional do processamento, evidenciando uma relação entre qualidade da reconstrução e custo computacional, sendo possível notar que se o número de passos se muito elevado, pode superar a resolução e não apresentar nenhum ganho real para a reconstrução. Para aplicações práticas, a escolha da resolução angular deve considerar esse equilíbrio, de acordo com os requisitos do objeto a ser escaneado.

Apesar dos resultados satisfatórios, algumas limitações foram observadas ao longo do desenvolvimento e dos experimentos. Superfícies altamente refletivas ou translúcidas apresentaram maior dificuldade de aquisição, devido à dispersão e saturação da luz do laser. Além disso, regiões sujeitas a oclusões geométricas ou ângulos de incidência desfavoráveis tendem a apresentar menor densidade de pontos, podendo gerar artefatos na reconstrução da superfície, como visto no objeto da Seção 4.3.2.4. Tais limitações são inerentes a sistemas de escaneamento baseados em luz estruturada de linha única.

Como trabalho futuro, destaca-se a possibilidade de incorporar uma análise estatística mais aprofundada dos resultados experimentais, de modo a avaliar de forma quantitativa a influência dos principais parâmetros do sistema na qualidade da reconstrução tridimensional. Métricas como densidade da nuvem de pontos, número de faces da malha, erro de reconstrução e tempo de processamento podem ser analisadas sob diferentes configurações de aquisição, tais como número de imagens, resolução angular da mesa rotativa e parâmetros de reconstrução.

Nesse contexto, a aplicação de métodos estatísticos inferenciais, como a Análise de Variância (ANOVA), permitiria verificar se as diferenças observadas entre configurações distintas são estatisticamente significativas, bem como identificar quais fatores exercem maior impacto na qualidade final da reconstrução. Essa abordagem possibilitaria uma avaliação mais rigorosa do compromisso entre custo computacional e ganho geométrico, atualmente analisado de forma predominantemente empírica.

5 CONCLUSÃO

Este trabalho apresentou o desenvolvimento e a validação de um sistema de escaneamento tridimensional baseado na técnica de luz estruturada por linha de laser, integrando conceitos de visão computacional, geometria tridimensional e controle de sistemas. O objetivo principal de construir um *scanner 3D* funcional, de baixo custo e com fundamentação geométrica consistente foi alcançado, conforme demonstrado pelos resultados experimentais apresentados.

Ao longo do desenvolvimento, foi implementado um fluxo completo de digitalização, abrangendo desde a aquisição sincronizada de imagens, calibração geométrica da câmera, estimação do plano da mesa e do plano do laser, até a reconstrução da nuvem de pontos e a geração de malhas tridimensionais. A abordagem adotada priorizou a modelagem explícita das relações geométricas envolvidas, evitando suposições idealizadas e permitindo capturar imperfeições reais do sistema físico, como desalinhamentos mecânicos e inclinações residuais.

Os resultados obtidos evidenciaram a importância de uma calibração geométrica robusta para a confiabilidade do sistema. A análise experimental demonstrou que métricas isoladas, como o erro médio de reprojeção, não são suficientes para avaliar isoladamente a qualidade da calibração, sendo a estabilidade dos parâmetros intrínsecos e dos coeficientes de distorção um indicador mais representativo da robustez do modelo.

Apesar do desempenho satisfatório, o sistema apresenta limitações inerentes à técnica de luz estruturada por linha única, especialmente em superfícies altamente refletivas, translúcidas ou sujeitas a oclusões geométricas. Também apresenta limitações de *hardware*, principalmente na câmera, por alta distorção na lente e resolução que não equipara à câmeras industriais de visão computacional. Contudo, tais limitações não invalidam o método proposto, mas indicam oportunidades claras para aprimoramentos futuros, tanto em nível de *hardware* quanto de *software*.

Como sugestões de melhoria, destaca-se a possibilidade de incorporar múltiplas fontes de laser ou padrões de projeção mais complexos, visando reduzir o efeito de oclusões e aumentar a cobertura da superfície do objeto. A utilização de técnicas de correção adaptativa de exposição e ganho da câmera também pode contribuir para uma detecção mais robusta da linha do laser em diferentes condições de refletância. A câmera, como coração do projeto, também poderia ser substituída por um modelo industrial desenvolvido para visão computacional, apresentando pouquíssima distorção na lente e alta resolução. Do ponto de vista algorítmico, métodos mais avançados de filtragem da nuvem de pontos e reconstrução de superfície, bem como estratégias de calibração automática contínua, podem ser explorados para aumentar a precisão e a robustez do sistema.

Desse modo, a modularidade do sistema desenvolvido permite sua extensão para diferentes cenários de aplicação, incluindo a integração com sensores adicionais, a automação

completa do processo de aquisição e a adaptação para ambientes industriais ou de pesquisa. Essas possibilidades reforçam o potencial do sistema como plataforma experimental para estudos futuros em digitalização tridimensional e visão computacional.

Além disso, avanços recentes na área de visão computacional indicam oportunidades promissoras para a integração do sistema desenvolvido com técnicas baseadas em *deep learning*. Modelos de aprendizado profundo podem ser empregados, por exemplo, na detecção e segmentação mais robusta da linha do laser, na filtragem adaptativa de ruídos da nuvem de pontos e na compensação de falhas causadas por reflexões ou oclusões. Adicionalmente, abordagens baseadas em projeção de múltiplos padrões de luz estruturada e estratégias multimodais, combinando diferentes sensores ou fontes de informação, podem ampliar a cobertura da superfície e aumentar a robustez da reconstrução tridimensional. Essas extensões representam caminhos naturais para a evolução do sistema, alinhando-o às tendências atuais de pesquisa em digitalização 3D e sistemas inteligentes (Lu et al., 2024; Zhao et al., 2024).

Por fim, o sistema apresentado neste trabalho constitui uma plataforma experimental sólida para estudos em digitalização tridimensional e visão computacional aplicada, podendo ser adaptado e expandido para diferentes contextos acadêmicos e industriais. O conhecimento adquirido ao longo deste projeto contribui não apenas para a compreensão dos fundamentos da reconstrução 3D, mas também para o desenvolvimento de soluções práticas baseadas em princípios geométricos bem estabelecidos.

REFERÊNCIAS

- AARANACORP. **Utilizar um controlador de motor passo a passo DRV8825 • AranaCorp**. 2025. Disponível em: <https://www.aranacorp.com/pt/utilizar-um-controlador-de-motor-passo-a-passo-drv8825/>. Acesso em: 20 dez. 2025.
- BLAIS, François. Review of 20 years of range sensor development. **Journal of Electronic Imaging**, v. 13, n. 1, p. 231–240, 2004.
- BRADSKI, Gary; KAEHLER, Adrian. **Learning OpenCV: Computer Vision with the OpenCV Library**. Sebastopol, CA: O’Reilly Media, 2008. ISBN 978-0-596-51613-0.
- CHEN, Jia; PING, Shantao; LIANG, Xiaowei; MA, Xulong; PANG, Shiyan; HE, Ying. Line-Structured Light-Based Three-Dimensional Reconstruction Measurement System with an Improved Scanning-Direction Calibration Method. **Remote Sensing**, v. 17, n. 13, p. 2236, 2025. ISSN 2072-4292.
- FORSYTH, David A.; PONCE, Jean. **Computer Vision: A Modern Approach**. 2. ed. Upper Saddle River, NJ: Prentice Hall, 2011. ISBN 9780136085928.
- GARRIDO-JURADO, S.; MUÑOZ-SALINAS, R.; MADRID-CUEVAS, F.; MEDINA-CARNICER, R. Automatic generation and detection of highly reliable fiducial markers under occlusion. **Pattern Recognition**, v. 47, n. 6, p. 2280–2292, 2014.
- GENG, Jason. Structured-light 3D surface imaging: a tutorial. **Advances in Optics and Photonics 2011-mar 31 vol. 3 iss. 2**, v. 3, 2011.
- HAN, Ming; ZHANG, Chao; ZHANG, Zhen; LI, Xiang. Review of MEMS Vibration-Mirror-Based 3D Reconstruction of Structured Light. **Sensors**, v. 25, n. 6, p. 6296, 2025. ISSN 1424-8220.
- HARTLEY, Richard; ZISSERMAN, Andrew. **Multiple View Geometry in Computer Vision**. 2. ed. [S.l.]: Cambridge University Press, 2003. ISBN 9780521540513.
- HIKDISTRIBUTION. **HIKVISION DS-2CD3656G2T-IZS 5 MP AcuSense Varifocal Bullet Network Camera**. 2025. Disponível em: <https://www.hikdistribution.com/products/5-mp-acusense-varifocal-bullet-network-camera-hikvision-ds-2cd3656g2t-izs/?om=16998&srsltid=AfmBOoqIzXYFnpxezsp2JdZdGvTtX2Tkg-9XbfoJ0T3FA8H9gYL3oYsgtA4&variant=44434137022741>. Acesso em: 17 dez. 2025.
- JIANG, Dong; TAO, Zhang. Monocular Camera Calibration Based on Particle Swarm Optimization with All-Parameter Adaptive Mutation Mechanism. chinese. **Journal of Southeast University (Natural Science Edition)**, v. 47, S1, p. 193–198, 2017.

KAZHDAN, Michael; BOLITHO, Matthew; HOPPE, Hugues. Poisson Surface Reconstruction. **Proceedings of the Fourth Eurographics Symposium on Geometry Processing**, 2006.

LI, Lei; XIAO, Zelong; HU, Taiyang. Camera Calibration Optimization Algorithm Based on Nutcracker Optimization Algorithm. **Sensors**, v. 25, p. 3521, 2025. ISSN 1424-8220.

LIN, Wenwei; LIANG, Peidong; LUO, Guantai; ZHAO, Ziyang; ZHANG, Chentao. Research of Online Hand–Eye Calibration Method Based on ChArUco Board. **Sensors**, MDPI, v. 22, n. 13, p. 5035, 2022.

LIU, Yonghuai; PEARS, Nick; ROSIN, Paul L.; HUBER, Patrik. **3D Imaging, Analysis and Applications**. Cham: Springer International Publishing, 2019. ISBN 9783030184994.

LU, Lei; BU, Chenhao; SU, Zhilong; GUAN, Banglei; YU, Qifeng; PAN, Wei; ZHANG, Qinghui. Generative Deep-Learning-Embedded Asynchronous Structured Light for Three-Dimensional Imaging. **Advanced Photonics**, SPIE, v. 6, n. 4, p. 046004, 2024. ISSN 2577-5463.

OPENCV. **Camera Calibration and 3D Reconstruction**. 2025. Disponível em: https://docs.opencv.org/4.x/d9/d0c/group___calib3d.html. Acesso em: 21 dez. 2025.

PAULY, Mark; GROSS, Markus; KOBELT, Leif. Efficient simplification of point-sampled surfaces. **Proceedings of the IEEE Visualization**, 2002.

PETER CORKE WITOLD JACHIMCZYK, Remo Pillat. **Robotics, Vision and Control: Fundamental Algorithms in MATLAB**. 3. ed. [S.l.]: Springer, 2023. (Springer Tracts in Advanced Robotics ^o147). ISBN 9783031072611.

PROESI. **MD-LSR650NM - Módulo Laser 650NM 5MW Alinhamento em Linha**. 2025. Disponível em: <https://www.proesi.com.br/md-lsr650nm-modulo-laser-650nm-5mw-alinhamento-em-linha>. Acesso em: 17 dez. 2025.

REMONDINO, Fabio; EL-HAKIM, Sabry F.; GRUEN, Armin; ZHANG, Li. Turning images into 3-D models. **IEEE Signal Processing Magazine**, v. 25, n. 4, p. 55–65, 2008.

RICOLFE-VIALA, Carlos. The Influence of the Focus in the Camera Calibration Process. **IEEE Transactions on Instrumentation and Measurement**, v. 61, n. 9, p. 2598–2611, 2012. ISSN 0018.

ROBOCORE. **ESP32 - WiFi + Bluetooth**. 2025. Disponível em: https://www.robocore.net/wifi/esp32-wifi-bluetooth?srsltid=AfmBOoqBiu3c47OZbtcYmRK5l95RrnRGKjg2CM5KXvF7_iowzqOkDbOO. Acesso em: 20 dez. 2025.

- ROCCHINI, C.; CIGNONI, P.; MONTANI, C.; PINGI, P.; SCOPIGNO, R. A low cost 3D scanner based on structured light. **Computer Graphics Forum**, v. 20, n. 3, p. 299–308, 2001.
- SANSONI, Giovanna et al. State-of-the-art and applications of 3D imaging sensors in industry, cultural heritage, medicine, and criminal investigation. **Sensors**, v. 9, n. 1, p. 568–601, 2009.
- SHAO, Mingwei; DONG, Junyu; MADESSA, Amanuel Hirpa. A New Calibration Method for Line-Structured Light Vision Sensors Based on Concentric Circle Feature. **Journal of the European Optical Society – Rapid Publications**, v. 15, n. 1, p. 1, 2019. ISSN 2193-9680.
- TAUBIN, Gabriel. 3D Scanning for Personal 3D Printing: Build Your Own Desktop 3D Scanner. **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)**, IEEE, p. 1–8, 2014.
- TAUBIN, Gabriel. Curve and Surface Smoothing Without Shrinkage. In: PROCEEDINGS of the Fifth International Conference on Computer Vision. [S.l.]: IEEE, 1995. p. 852–857.
- YU, Jiahui; LIU, Ziheng; YANG, Hengxin; YANG, Hao. The Line-Structured Light Plane Calibration based on the Plücker Line. In: 2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE). [S.l.]: IEEE, 2021. p. 1106–1109.
- ZHANG, Zhengyou. A Flexible New Technique for Camera Calibration. **Proceedings of IEEE International Conference on Computer Vision**, p. 666–673, 2000.
- ZHAO, Xinyue; YU, Tiancheng; LIANG, Dong; HE, Zaixing. A review on 3D measurement of highly reflective objects using structured light projection. **The International Journal of Advanced Manufacturing Technology**, Springer, v. 132, p. 4205–4222, 2024. ISSN 0268-3768.

APÊNDICE A – Repositório do Código-Fonte

O código-fonte completo desenvolvido neste trabalho encontra-se disponível em um repositório público na plataforma GitHub, contendo os *scripts* utilizados na calibração, reconstrução tridimensional e pós-processamento dos dados.

O repositório pode ser acessado por meio do seguinte endereço:

https://github.com/BrunoBronzeri/Scanner3D_TCC/tree/main

O código está organizado em diretórios correspondentes às principais etapas do pipeline descrito ao longo do trabalho, permitindo a reprodução dos experimentos e a análise detalhada das implementações.