

Universidade de Brasília
Faculdade de Tecnologia

**Sensor Visual de Alta Precisão para
Calibração de Robôs Industriais:
Posicionamento, Orientação e Controle de
Movimento**

Alex Sandro dos Santos Dias

DISSERTAÇÃO DE MESTRADO
SISTEMAS MECATRÔNICOS

Brasília
2024

Universidade de Brasília
Faculdade de Tecnologia

**Sensor Visual de Alta Precisão para
Calibração de Robôs Industriais:
Posicionamento, Orientação e Controle de
Movimento**

Alex Sandro dos Santos Dias

Dissertação de Mestrado submetida ao Departamento de Engenharia Mecânica como requisito parcial para obtenção do grau de Mestre em Sistemas Mecatrônicos

Orientador: Prof. Dr. José Maurício Santos Torres da Motta

Brasília

2024

FICHA CATALOGRÁFICA

Dias, Alex Sandro dos Santos.

Sensor Visual de Alta Precisão para Calibração de Robôs Industriais: Posicionamento, Orientação e Controle de Movimento / Alex Sandro dos Santos Dias; orientador José Maurício Santos Torres da Motta. -- Brasília, 2024.

152 p.

Dissertação de Mestrado (Sistemas Mecatrônicos) -- Universidade de Brasília, 2024.

1. Sensor de Visão Ativa. 2. Medição de Posição de Manipuladores. 3. Controle Visual por Imagem. 4. Controle de Movimento de Robôs. 5. Calibração de Robôs. I. Motta, José Maurício Santos Torres da, orient. II. Sistemas Mecatrônicos

**Universidade de Brasília
Faculdade de Tecnologia**

**Sensor Visual de Alta Precisão para Calibração de
Robôs Industriais:
Posicionamento, Orientação e Controle de
Movimento**

Alex Sandro dos Santos Dias

Dissertação de Mestrado submetida ao Departamento de Engenharia Mecânica como requisito parcial para obtenção do grau de Mestre em Sistemas Mecatrônicos

Trabalho aprovado. Brasília, 16 de dezembro de 2024:

**Prof. Dr. José Maurício Santos Torres da
Motta, UnB/FT/ENM**
Orientador

**Prof. Dr. Gerardo Antonio Idrobo Pizo,
UnB/FGA**
Examinador interno

**Prof. Dr. Juan Sebastian Toquica Arenas,
UFC/Campus Itapajé**
Examinador externo

Brasília
2024

Dedico este trabalho a Deus e à minha família, especialmente aos meus pais, cujo esforço incansável foi fundamental em cada etapa da minha educação.

Agradecimentos

Expresso minha gratidão em primeiro lugar a Deus, o criador e detentor de toda sabedoria. Agradeço profundamente aos meus pais, que foram verdadeiros pilares ao longo da minha longa jornada acadêmica.

Um agradecimento especial à minha querida filha, Bianca Keroly, por sua contribuição crucial durante este processo de pós-graduação. Reconheço o apoio inestimável da minha amada esposa, Gil, que sempre esteve ao meu lado, me incentivando e inspirando a seguir em frente. Sua presença foi fundamental para minha perseverança e conclusão deste curso.

Não posso deixar de expressar minha gratidão a todos os professores da UnB que, de alguma forma, contribuíram para o meu crescimento acadêmico. Em particular, desejo agradecer ao professor José Maurício Santos Torres da Motta, meu orientador neste trabalho, cujos conhecimentos foram essenciais para a sua conclusão. Também sou grato ao professor Guilherme Caribé, que compartilhou seu profundo conhecimento em sistemas dinâmicos, enriquecendo minha formação.

Reconheço que superar os desafios de um curso de pós-graduação demanda uma jornada árdua, que não pode ser percorrida sozinho. Portanto, estendo meu sincero agradecimento a todos os envolvidos neste processo. Não há palavras suficientes para expressar a minha gratidão a todos aqueles que ajudaram a transformar um sonho em realidade.

*“Clame por inteligência e peça entendimento.
Pois o Senhor concede sabedoria; de sua boca vêm conhecimento e entendimento.”
(Provérbios 2.3,6)*

Resumo

A presente dissertação apresenta o desenvolvimento de um sensor baseado em visão que realiza o controle de posição e orientação do efetuador de um manipulador robótico, de maneira que o efetuador atinja pontos a distâncias sempre constantes do centro do alvo, com orientação determinada, e com precisão suficiente para garantir alta repetibilidade de medição. Um sensor como este pode ser utilizado para tarefas de calibração de robôs em que é imprescindível medir a posição e orientação do manipulador em diversos locais do espaço de trabalho para comparar com as posturas fornecidas pelo controlador. O sensor desenvolvido utiliza o controle servo-visual baseado em imagem, que dispensa a necessidade de estimar a posição e a orientação do manipulador. Ele se baseia exclusivamente nas características extraídas diretamente da imagem captada pela câmera, permitindo emitir sinais de controle para que o robô alcance o alvo. Este método produz medições muito mais precisas do que o método tradicional de medição com controle de posição, devido à sua menor dependência da exatidão dimensional de alvos, das imagens serem de mesmo tamanho e de não haver necessidade de calibrar os parâmetros extrínsecos da câmera previamente ou a cada medição. O sensor é composto de uma câmera e um emissor de plano de luz laser fixo à plataforma da câmera, em configuração pré-estabelecida, de maneira que o plano de luz, ao atingir o centro do alvo, mantém a distância pré-estabelecida da câmera ao alvo ao longo do eixo óptico. A orientação pode ser medida a partir da distribuição lateral de formas em torno do ponto central do alvo. Este tipo de medição permite incluir a distância constante ao alvo como se fosse parte do efetuador, em uma configuração que permite movimento esférico de raio constante em torno do alvo. Resultados preliminares demonstram que o método tem grande potencial de precisão de medição para uso em metodologias de calibração de robôs, com exatidão de medição na ordem de décimos de milímetro.

Palavras-chave: Sensor de Visão Ativa. Medição de Posição de Manipuladores. Controle Visual por Imagem. Controle de Movimento de Robôs. Calibração de Robôs.

Abstract

This dissertation presents the development of a vision-based sensor that controls the position and orientation of the effector of a robotic manipulator, so that the effector reaches points at constant distances from the center of the target, with determined orientation, and with sufficient precision to ensure high repeatability of measurement. The developed sensor employs image-based visual servoing control, which eliminates the need to estimate the manipulator's position and orientation. It relies exclusively on features extracted directly from the image captured by the camera, enabling the generation of control signals for the robot to reach the target. This method produces much more accurate measurements than the traditional method of position-controlled measurement, due to its lower reliance on the dimensional accuracy of targets, the images being the same size, and the need to calibrate the camera's extrinsic parameters beforehand or with each measurement. The sensor is composed of a camera and a laser light plane emitter fixed to the camera platform, in a pre-established configuration, so that the light plane, when reaching the center of the target, maintains the pre-established distance from the camera to the target along the optical axis. Orientation can be measured from the lateral distribution of forms around the central point of the target. This type of measurement allows the inclusion of constant distances to the target as if they were part of the effector, in a configuration that allows for constant radius spherical motion around the target. Preliminary results demonstrate that the method has great potential for measurement accuracy for use in robot calibration methodologies, with measurement accuracy in the order of tenths of a millimeter.

Keywords: Active Vision Sensor. Manipulator Pose Measurement. Image Visual Control. Robot Motion Control. Robot Calibration.

Lista de figuras

Figura 1.1	Problemas inerentes aos manipuladores. Fonte: Elaboração própria.	23
Figura 2.1	Conjunto de palavras chaves para busca.	28
Figura 2.2	Publicações por ano - base <i>SCOPUS</i>	29
Figura 2.3	Publicações e citações por ano - base <i>WEB OF SCIENCE</i>	29
Figura 2.4	Intensidade de citações em mapa de calor - base <i>SCOPUS</i>	30
Figura 2.5	Intensidade de citações em mapa de calor - base <i>WEB OF SCIENCE</i>	30
Figura 3.1	Aquisição e digitalização de imagem (coppe (2024)).	39
Figura 3.2	Sistemas de calibração (Fonte: Vianna (2009) adaptado).	41
Figura 3.3	Configurações para o controle servo-visual e as relevâncias para coordenadas globais a partir dos quadros da imagem: efetuador final {E}, câmera {C} e destino {O}. a) configuração com câmera no efetuador do robô <i>Eye-in-Hand</i> ; b) Configuração com câmera fixa em coordenadas globais <i>end-point open-loop</i>	42
Figura 3.4	As duas classes distintas para os sistemas de controle servo visual.	43
Figura 3.5	Modelo de câmera perspectiva. Fonte: Hartley e Zisserman (2003).	44
Figura 3.6	(a) plano da imagem com o sistema de coordenadas da câmera $(u, v)^T$ e a imagem $(x, y)^T$; (b) imagem inclinada e as coordenadas da imagem com diferentes escalas na direção dos eixos x e y . Fonte: Morvan (2009).	48
Figura 3.7	(a) imagem com distorção radial; (b) imagem com distorção radial corrigida. Fonte: Fry e Pusateri (2010).	48
Figura 3.8	Estimação da distorção radial utilizando um padrão de calibração. Fonte: Kim <i>et al.</i> (2010).	49
Figura 3.9	Transformação Euclidiana entre o sistema de coordenadas da câmera e do mundo. Fonte: Hartley e Zisserman (2003).	49
Figura 3.10	representação da equação da reta (<i>declive-intercepte</i>).	54
Figura 3.11	(a) Plano $x - y$ (imagem) e (b) Plano $a - b$ (parâmetros).	55
Figura 3.12	representação da equação da reta (coordenada polar).	55
Figura 3.13	Transformação do espaço $x - y$ original para o espaço parametrizado $\theta - \rho$	56
Figura 3.14	Movimento do órgão terminal devido ao i -ésimo membro.	67
Figura 4.1	Equipamentos utilizados para construção do sensor: a) Câmera acA2500 - $14\mu\text{m}$; b) Laser LDM115 e gerador de linha óptico LGO115 e c) Lente CFFL F1.4 f16mm 2/3".	74
Figura 4.2	Desenho técnico do robô industrial ABB IRB 140 em milímetros (Robotics, 2003).	75
Figura 4.3	Sensor de visão: a) Vista isométrica; b) Vista interna.	76
Figura 4.4	Sensor de Medição baseado em triangulação. Fonte: Elaboração própria.	77

Figura 4.5	Robô IRB140 com sensor de visão acoplado.	79
Figura 4.6	Alocação de eixos da notação Denavit-Hartenberg para o robô IRB140 com sensor de visão acoplado.	80
Figura 5.1	Fluxograma do sistema.	83
Figura 5.2	Alvo utilizado para obtenção de pontos característicos.	85
Figura 5.3	Demonstração Esquemática de Posições Iniciais da Câmera.	86
Figura 5.4	Imagens capturadas pela câmera no decorrer do movimento - Experimento 1.	88
Figura 5.5	Valor absoluto da posição no eixo x - Experimento 1.	89
Figura 5.6	Valor absoluto da posição no eixo y - Experimento 1.	89
Figura 5.7	Valor absoluto da posição no eixo z - Experimento 1.	90
Figura 5.8	Valor absoluto da Profundidade Estimada - Experimento 1.	90
Figura 5.9	Valor absoluto da orientação Roll - Experimento 1.	91
Figura 5.10	Valor absoluto da orientação Pitch - Experimento 1.	91
Figura 5.11	Valor absoluto da orientação Yaw - Experimento 1.	92
Figura 5.12	Valor absoluto da rotação da Junta 1 - Experimento 1.	92
Figura 5.13	Valor absoluto da rotação da Junta 2 - Experimento 1.	93
Figura 5.14	Valor absoluto da rotação da Junta 3 - Experimento 1.	93
Figura 5.15	Valor absoluto da rotação da Junta 4 - Experimento 1.	94
Figura 5.16	Valor absoluto da rotação da Junta 5 - Experimento 1.	94
Figura 5.17	Valor absoluto da rotação da Junta 6 - Experimento 1.	95
Figura 5.18	Exemplo de trajetória do manipulador no espaço com posição e orientação inicial A para final B - 1ª Repetição do Experimento 4	97
Figura D.1	Imagens capturadas pela câmera no decorrer do movimento (1ª a 15ª interação) - Experimento 2.	122
Figura D.2	Imagens capturadas pela câmera no decorrer do movimento (16ª a 22ª interação) - Experimento 2.	123
Figura D.3	Valor absoluto da posição no eixo x - Experimento 2.	123
Figura D.4	Valor absoluto da posição no eixo y - Experimento 2.	124
Figura D.5	Valor absoluto da posição no eixo z - Experimento 2.	124
Figura D.6	Valor absoluto da Profundidade Estimada - Experimento 2.	125
Figura D.7	Valor absoluto da orientação Roll - Experimento 2.	125
Figura D.8	Valor absoluto da orientação Pich - Experimento 2.	126
Figura D.9	Valor absoluto da orientação Yaw - Experimento 2.	126
Figura D.10	Valor absoluto da rotação da Junta 1 - Experimento 2.	127
Figura D.11	Valor absoluto da rotação da Junta 2 - Experimento 2.	127
Figura D.12	Valor absoluto da rotação da Junta 3 - Experimento 2.	128
Figura D.13	Valor absoluto da rotação da Junta 4 - Experimento 2.	128
Figura D.14	Valor absoluto da rotação da Junta 5 - Experimento 2.	129

Figura D.15	Valor absoluto da rotação da Junta 6 - Experimento 2.	129
Figura D.16	Imagens capturadas pela câmera no decorrer do movimento (1ª a 15ª interação) - Experimento 3.	130
Figura D.17	Imagens capturadas pela câmera no decorrer do movimento (16ª a 19ª interação) - Experimento 3.	131
Figura D.18	Valor absoluto da posição no eixo x - Experimento 3.	131
Figura D.19	Valor absoluto da posição no eixo y - Experimento 3.	132
Figura D.20	Valor absoluto da posição no eixo z - Experimento 3.	132
Figura D.21	Valor absoluto da Profundidade Estimada - Experimento 3.	133
Figura D.22	Valor absoluto da orientação Roll - Experimento 3.	133
Figura D.23	Valor absoluto da orientação Pitch - Experimento 3.	134
Figura D.24	Valor absoluto da orientação Yaw - Experimento 3.	134
Figura D.25	Valor absoluto da rotação da Junta 1 - Experimento 3.	135
Figura D.26	Valor absoluto da rotação da Junta 2 - Experimento 3.	135
Figura D.27	Valor absoluto da rotação da Junta 3 - Experimento 3.	136
Figura D.28	Valor absoluto da rotação da Junta 4 - Experimento 3.	136
Figura D.29	Valor absoluto da rotação da Junta 5 - Experimento 3.	137
Figura D.30	Valor absoluto da rotação da Junta 6 - Experimento 3.	137
Figura D.31	Imagens capturadas pela câmera no decorrer do movimento (1ª a 15ª interação) - Experimento 4.	138
Figura D.32	Imagens capturadas pela câmera no decorrer do movimento (16ª a 20ª interação) - Experimento 4.	139
Figura D.33	Valor absoluto da posição no eixo x - Experimento 4.	139
Figura D.34	Valor absoluto da posição no eixo z - Experimento 4.	140
Figura D.35	Valor absoluto da Profundidade Estimada - Experimento 4.	140
Figura D.36	Valor absoluto da orientação Roll - Experimento 4.	141
Figura D.37	Valor absoluto da orientação Pitch - Experimento 4.	141
Figura D.38	Valor absoluto da orientação Yaw - Experimento 4.	142
Figura D.39	Valor absoluto da rotação da Junta 1 - Experimento 4.	142
Figura D.40	Valor absoluto da rotação da Junta 2 - Experimento 4.	143
Figura D.41	Valor absoluto da rotação da Junta 3 - Experimento 3.	143
Figura D.42	Valor absoluto da rotação da Junta 4 - Experimento 4	144
Figura D.43	Valor absoluto da rotação da Junta 5 - Experimento 4.	144
Figura D.44	Valor absoluto da rotação da Junta 6 - Experimento 4.	145
Figura D.45	Imagens capturadas pela câmera no decorrer do movimento - Experimento 5	145
Figura D.46	Valor absoluto da posição no eixo x - Experimento 5.	146
Figura D.47	Valor absoluto da posição no eixo y - Experimento 5.	146
Figura D.48	Valor absoluto da posição no eixo z - Experimento 5.	147

Figura D.49	Valor absoluto da Profundidade Estimada - Experimento 5.	147
Figura D.50	Valor absoluto da orientação Roll - Experimento 5.	148
Figura D.51	Valor absoluto da orientação Pitch - Experimento 5.	148
Figura D.52	Valor absoluto da orientação Yaw - Experimento 5.	149
Figura D.53	Valor absoluto da rotação da Junta 1 - Experimento 5.	150
Figura D.54	Valor absoluto da rotação da Junta 2 - Experimento 5.	150
Figura D.55	Valor absoluto da rotação da Junta 3 - Experimento 5.	151
Figura D.56	Valor absoluto da rotação da Junta 4 - Experimento 5.	151
Figura D.57	Valor absoluto da rotação da Junta 5 - Experimento 5.	152
Figura D.58	Valor absoluto da rotação da Junta 6 - Experimento 5.	152

Lista de tabelas

Tabela 2.1	Resultado para artigos na base <i>SCOPUS</i>	28
Tabela 2.2	Resultado para artigos na base <i>WEB OF SCIENCE</i>	28
Tabela 2.3	Atributos de visão computacional.	33
Tabela 2.4	Atributos robóticos.	34
Tabela 2.5	Atributos construtivos.	35
Tabela 4.1	Especificações da Câmera.	74
Tabela 4.2	Especificações da Lente.	74
Tabela 4.3	Especificações do Laser.	75
Tabela 4.4	Hardware e Software Utilizados	75
Tabela 4.5	Parâmetros de Denavit-Hartenberg para o robô IRB140	81
Tabela 5.1	Posições iniciais das juntas do manipulador para os 5 experimentos (em graus).	85
Tabela 5.2	Posições e orientações iniciais da câmera para os 5 experimentos.	85
Tabela 5.3	Erro entre as posições finais e iniciais da posição <i>X</i>	95
Tabela 5.4	Erro entre as posições finais e iniciais da posição <i>Y</i>	95
Tabela 5.5	Erro entre as posições finais e iniciais da posição <i>Z</i>	95
Tabela 5.6	Erro entre as orientações finais e iniciais da orientação Roll.	96
Tabela 5.7	Erro entre as orientações finais e iniciais da orientação Pitch.	96
Tabela 5.8	Erro entre as orientações finais e iniciais da orientação Yaw.	96
Tabela 5.9	Erro entre as posições finais e iniciais da Junta 1.	96
Tabela 5.10	Erro entre as posições finais e iniciais da Junta 2.	96
Tabela 5.11	Erro entre as posições finais e iniciais da Junta 3.	96
Tabela 5.12	Erro entre as posições finais e iniciais da Junta 4.	97
Tabela 5.13	Erro entre as posições finais e iniciais da Junta 5.	97
Tabela 5.14	Erro entre as posições finais e iniciais da Junta 6.	97

Lista de abreviaturas e siglas

A/D	Analógico para digital
CCD	<i>charge-coupled device</i>
CMOS	<i>complementary metal oxide semiconductor</i>
DH	Denavit-Hartengerg
HM	homem-máquina
IA	Inteligência Artificial
IBVS	Controle Servo Visual Baseado em Imagem
IFR	International Federation of Robotics
P&D	investigação e Desenvolvimento
PBVS	Controle Servo Visual Baseado em Posição
RGB	<i>red, green e blue</i>
SMPI	Sistema de Medição de Profundidade da Imagem
SNR	<i>Signal to noise ratio</i>
TSCSV	Tipo de Sistema de Controle Servo-visual

Lista de símbolos

Símbolos romanos

DH	Matriz de transformação homogênea para notação de Denavit-Hartengerg
\mathbf{d}_0^n	Posição do sistema do órgão terminal em relação ao sistema da base
E	<i>vetor gradiente local</i>
I	<i>Matriz identidade</i>
\mathbf{J}_m	Jacobiano do Manipulador
\mathbf{J}_s	Jacobiano da imagem
k	Vetor unitário
p	ponto da imagem capturada
\mathbf{p}^*	ponto da imagem desejada
Q	Matriz de projeção da imagem
q	Variáveis das juntas robótica
\mathbf{q}_i	Variável <i>i</i> -ésima da junta robótica
\mathbf{R}_0^n	Orientação do sistema do órgão terminal em relação ao sistema da base
\mathbf{T}_0^n	Transformação do sistema do órgão terminal em relação ao sistema da base
\mathbf{v}_c	velocidade da câmera
\mathbf{v}_t	velocidade linear da câmera
\tilde{C}	Parâmetro extrínseco da câmera - centro de projeção da câmera no mundo
C_x	Coordenada em pixel horizontal do ponto principal
C_y	Coordenada em pixel vertical do ponto principal
G	Filtro de Gauss
K	Parâmetro intrínseco da câmera
p_x	Coordenada horizontal do ponto principal calibrado
p_y	Coordenada vertical do ponto principal calibrado
R	Parâmetro extrínseco da câmera - Matriz de rotação
u	Coordenada horizontal em pixel na imagem formada
v	Coordenada vertical em pixel na imagem formada
v_x	velocidade linear da câmera na direção <i>x</i>
v_y	velocidade linear da câmera na direção <i>y</i>
v_z	velocidade linear da câmera na direção <i>z</i>
X	Coordenada horizontal no espaço
x	Coordenada horizontal no plano da imagem
Y	Coordenada vertical no espaço
y	Coordenada vertical no plano da imagem
Z	Coordenada de profundidade no espaço

Símbolos gregos

α_i	Rotação em torno do eixo X da junta i-ésima
δ	Fator de escala homogêneo
$\dot{\theta}$	Velocidade Angular das Juntas do Manipulador
η_x	largura da imagem
η_y	altura da imagem
λ	Controlador proporcional de velocidade
ω_x	velocidade angular da câmera em relação ao eixo x
ω_y	velocidade angular da câmera em relação ao eixo y
ω_z	velocidade angular da câmera em relação ao eixo z
σ	Desvio padrão
τ	parâmetro de inclinação da imagem
\mathbf{v}_Ω	velocidade angular da câmera
θ_i	Rotação em torno do eixo Z da junta i-ésima

Sumário

1	INTRODUÇÃO	19
1.1	A Robótica e a Humanidade Hoje	19
1.2	Os Desafios da Robótica Moderna	21
1.3	Definição do Problema	22
1.4	Hipóteses do Trabalho	24
1.5	Objetivos	25
1.5.1	Objetivos Específicos	25
1.6	Estrutura do Texto	26
2	BIBLIOMETRIA COM LITERATURAS RELACIONADAS	27
2.1	Palavras Chaves para Pesquisa Bibliométrica	27
2.2	Tabela comparativa de temas relacionados	28
2.2.1	Parâmetros para Visão Computacional	31
2.2.2	Atributos para Robótica	31
2.2.3	Parâmetros Construtivos	32
3	FUNDAMENTAÇÃO TEÓRICA E BIBLIOGRÁFICA	36
3.1	Visão Computacional	36
3.1.1	Aquisição de Imagens	36
3.1.2	Digitalização de Imagem	38
3.2	Robótica e a Utilização de Câmeras	40
3.2.1	Controle Servo-Visual	41
3.3	Conceito em Projeção de Imagem	43
3.3.1	Modelo de Câmera Perspectiva	43
3.4	Conceitos de Processamento de Imagem	50
3.4.1	Alteração de Dados da Imagem Digitalizada	50
3.4.2	Filtragem Gaussiana	50
3.4.3	Detecção de Bordas	51
3.4.4	Detecção de Linhas com Transformada de Hough	53
3.4.5	Detecção de Círculos e de Elipses	56
3.5	Controle Servo Visual Baseado em Imagem - IVBS	57
3.5.1	Matriz Jacobiano da Imagem	58
3.6	Robótica	61
3.6.1	Cinemática Direta: A Notação de Denavit-Hartenberg DH	63
3.6.2	Jacobiano Analítico de Manipuladores Robóticos	64
3.6.3	Dedução do Jacobiano (Spong, Hutchinson e Vidyasagar (2020))	65

3.7	Calibração de Robôs Industriais	69
4	MEDIÇÃO DE POSIÇÃO E ORIENTAÇÃO DE ROBÔS POR VISÃO E CONTROLE DE MOVIMENTO	73
4.1	Equipamentos	73
4.2	Modelagem do Sensor para Obtenção da Profundidade da Imagem	73
4.3	Modelagem do Sensor para o Controle Servo-Visual Baseado em Imagem IBVS	78
4.4	Manipulador ABB IRB140	79
5	EXPERIMENTOS E RESULTADOS	82
5.1	Metodologia Experimental	82
5.2	Planejamento para coleta de dados de interesse	82
5.2.1	Configuração do Alvo e Condições Experimentais	82
5.2.2	Aquisição de Dados	83
5.2.3	Critérios de Avaliação	83
5.2.4	Utilização dos Resultados para Calibração Robótica	84
5.3	Resultados	84
5.3.1	Definição do Alvo e Pontos Característicos	84
5.3.2	Execução dos Experimentos	85
5.3.3	Coleta de Resultados	87
5.4	Limitações do Método Experimental	98
5.5	Análise Geral	99
6	CONCLUSÕES E PERSPECTIVAS DE TRABALHOS FUTUROS	102
6.1	Conclusões	102
6.2	Perspectivas de Trabalhos Futuros	103
	Referências	104
	Apêndices	113
	Apêndice A Cinemática Robótica	114
	A.1 Cinemática da Velocidade e da Aceleração	114
	Apêndice B Códigos de programação	120
	Apêndice C Dados Experimentais	121
	Apêndice D Resultados Experimentais	122

1 INTRODUÇÃO

A robótica industrial tem desempenhado um papel essencial na automação de processos produtivos, revolucionando o ambiente industrial e agregando inúmeras vantagens, como aumento da produtividade, precisão e redução de desperdícios (Brown e Bessant (2003)). Nos últimos anos, avanços em tecnologias como sensores de visão, sistemas de controle servo-visual e periféricos inteligentes ampliaram as possibilidades dos robôs industriais, permitindo sua aplicação em cenários que demandam maior flexibilidade e integração com ambientes dinâmicos (Qin, Liu e Grosvenor (2016)).

Além disso, a robótica encontra aplicação em setores como aeroespacial, militar, medicinal e doméstico, bem como em interações humanas. Segundo o IFR (International Federation of Robotics), o mundo atingiu um novo recorde de 3,9 milhões de robôs operacionais em 2022 impulsionado pelo alto volume de instalações de robôs industriais. Os países mais automatizados medidos pela densidade de robôs são: A República da Coreia (1.012 robôs por 10.000 funcionários), Singapura (730 unidades) e Alemanha (415 unidades). Isto está de acordo com o relatório World Robotics 2023, apresentado pela IFR (2022).

A robótica industrial conquistou significativa relevância na manufatura desde o surgimento dos robôs elétricos, ocorrido há mais de 45 anos. O avanço tecnológico da robótica industrial tem caminhado lado a lado com o progresso da eletrônica, microeletrônica, PCs industriais, software, interfaces e mecânica de precisão, além de materiais e metais especiais, todos elementos essenciais para o desenvolvimento da mecatrônica. Um robô industrial é uma complexa integração de diversas tecnologias, que, quando harmonizadas, resultam em uma máquina singular em termos de processos de fabricação e grande flexibilidade em suas múltiplas e variadas aplicações.

Com os avanços ocorridos nos últimos anos, os limites do que um robô industrial pode realizar tornaram-se cada vez mais amplos. Atualmente, com a vasta gama de tecnologias disponíveis, como periféricos, sistemas de visão computacional 2D/3D, sensores de diversas finalidades, softwares de aplicação específicos, interfaces, entre outros, os robôs industriais têm potencial para operar em praticamente qualquer ambiente ou contexto.

1.1 A Robótica e a Humanidade Hoje

O segmento da indústria 4.0 voltado para a robótica tem se destacado consideravelmente no ambiente de trabalho, conquistando a aceitação do público, como destacado por Qin, Liu e Grosvenor (2016). Isso se deve principalmente à eficiência e ao aumento da produtividade que essa solução proporciona, contribuindo para uma operação mais precisa e eficaz.

Na indústria de manufatura moderna, o controle de qualidade do produto desempenha um papel crucial na melhoria da qualidade, na redução das taxas de rejeição e na diminuição de custos. A tecnologia de inspeção/medição 3D baseada em visão é amplamente utilizada para supervisionar a qualidade devido às suas vantagens, como operação sem contato, rápida aquisição de dados, baixo custo e estabilidade (Chen, Brown e Song (2000), Yang *et al.* (2008), Biber *et al.* (2004)). Essa tecnologia consiste em duas partes principais: o sensor visual e o dispositivo de orientação. O dispositivo de orientação posiciona o sensor visual corretamente em relação ao objeto em destaque. Para aplicações industriais, é preferível que o dispositivo de orientação seja flexível e controlável, permitindo inspeções eficientes e livres de intervenção humana. Nesse contexto, a adoção de robôs industriais como dispositivos orientadores é uma ideia promissora (Kjellander; Rahayem, 2009). Esses sistemas robóticos de inspeção visual combinam a alta flexibilidade dos robôs industriais com a precisão do sensor visual e são amplamente aplicados nas indústrias automotiva e aeronáutica (Larsson e Kjellander (2004), Liu *et al.* (2004)).

O conceito de indústria 4.0 foi apresentado ao público pela primeira vez na Feira de Hannover, na Alemanha, em 2011, onde o governo local o definiu como um projeto estratégico de alta tecnologia. Esse projeto visa informatizar a manufatura por meio da comunicação em tempo real entre seres humanos, robôs, logística de fábrica e administração da empresa, conforme abordado por Rainer e Alexander (2014). Diversas tecnologias digitais, conhecidas como habilitadoras, como robótica avançada, manufatura aditiva, simulação, realidade aumentada, internet das coisas e nuvem de dados, entre outras, têm contribuído para integrar as organizações ao contexto da indústria 4.0, como apontado por Wang *et al.* (2016).

Desde então, o uso de robôs tem testemunhado um rápido desenvolvimento no ambiente industrial, impulsionando a automação industrial. Através da automação, os robôs são capazes de executar tarefas cada vez mais complexas com precisão elevada e de maneira economicamente viável. Em suma, a robótica industrial está se tornando essencial no contexto da Indústria 4.0.

No ambiente industrial cada vez mais influenciado pela Indústria 4.0, a robótica industrial desempenha diversas funções vitais, sendo as principais:

Montagem robótica – O uso de robôs em operações de montagem é uma aplicação comum na indústria. Complementando as atividades dos trabalhadores em tarefas específicas, especialmente aquelas mais repetitivas, os robôs garantem maior precisão, redução de erros e aumento da produtividade;

Robô colaborativo – Apesar dos avanços na área da robótica, muitas atividades ainda demandam a intervenção humana. Os robôs colaborativos surgem como uma solução viável, sendo capazes de interagir de forma segura com as pessoas. Esta abordagem facilita a coexistência harmoniosa entre humanos e máquinas Kootbally (2016). A vantagem dos robôs

colaborativos reside na sua capacidade de se adaptar a uma variedade de setores industriais, além de possuírem uma rápida adaptação a novas tarefas Bloss (2016), Ceriani *et al.* (2015), Michalos *et al.* (2014);

Transferências de itens – Existem robôs projetados e programados especificamente para realizar transferências de materiais, peças e produtos ao longo da linha de produção de uma fábrica, em um conceito que está sendo denominado como logística 4.0. Esses robôs industriais substituem atividades que antes eram realizadas por humanos, especialmente aquelas que são perigosas ou fisicamente exaustivas;

Tarefas de corte – Os robôs têm a capacidade de seguir com precisão instruções pré-programadas, tornando-os ideais para tarefas como o corte de materiais. Com controle preciso de pressão e velocidade, esses robôs conseguem aparar, cortar ou polir superfícies de maneira uniforme e eficiente;

Tarefas de soldagem – A soldagem é uma das primeiras tarefas delegadas aos robôs. Sua eficiência é notável, uma vez que podem ser programados para posicionar a tocha de soldagem na mesma orientação em todos os ciclos. Além disso, sua alta capacidade de repetição assegura que cada peça seja soldada com precisão e qualidade consistentes.

Diante das inúmeras possibilidades e aplicações, a robótica industrial proporciona uma série de vantagens significativas para o ambiente industrial, tais como:

1- Maior produtividade dentro do mesmo espaço fabril. Os robôs são capazes de realizar trabalhos repetitivos de forma ininterrupta e consistente. Isso possibilita a divisão das tarefas de produção entre humanos e máquinas, liberando os operadores de atividades monótonas e repetitivas. Dessa forma, eles podem se concentrar em operações de maior valor agregado Ibarguren *et al.* (2015);

2- Há uma melhoria significativa na qualidade quando os robôs são utilizados, pois eles executam as tarefas com consistência, mantendo a mesma força, intensidade, velocidade e padrão em todas as repetições;

3- Precisão na execução das tarefas de manufatura;

4- Redução de resíduos e perdas em geral;

5- Redução de desperdícios de produtos acabados, principalmente com menor intervenção associados a erros humanos e às oscilações no fluxo produtivo;

7- Aumento da competitividade na indústria.

1.2 Os Desafios da Robótica Moderna

As características tecnológicas dos robôs industriais, como sua precisão em relação aos operadores, são de extrema importância. Eles desempenham tarefas com ciclos de trabalho consistentes e repetitivos, lidam com atividades perigosas e são recomendados

para ambientes insalubres [Groover \(2016\)](#), [Brown e Bessant \(2003\)](#). Contudo, desafios ainda persistem. Estudos mais recentes destacam a necessidade de maior flexibilidade e integração de tecnologias avançadas, como gêmeos digitais e colaboração homem-máquina, para superar limitações como a dificuldade de adaptação a diferentes produtos na célula de produção [Wabner et al. \(2022\)](#), [Sanneman, Fourie e Shah \(2020\)](#), [Chen et al. \(2022\)](#), [Ajewole et al. \(2023\)](#).

Um dos principais desafios enfrentados pela indústria de robótica industrial é simplificar ainda mais a programação, operação e manutenção dos robôs na linha de produção. Uma tendência notável é o aumento da conectividade dos robôs na indústria. Um equívoco comum na automação robótica é a crença de que sua implementação elimina completamente a necessidade de colaboração humana no processo. Pelo contrário, a presença de colaboradores na indústria é fundamental, permitindo que pensem de forma criativa e crítica sobre o processo de automatização [Ajewole et al. \(2023\)](#).

O controle baseado em sensores de visão é uma das áreas de destaque, oferecendo soluções eficazes para superar desafios como erros de posicionamento e orientação dos manipuladores. Tecnologias convencionais, como sensores mecânicos, apresentam limitações relacionadas ao custo elevado e à complexidade de instalação ([Larsson e Kjellander \(2004\)](#)). Por outro lado, sistemas baseados em visão computacional, em especial o controle servo-visual baseado em imagem (Image-Based Visual Servoing – IBVS), destacam-se por sua precisão, custo relativamente reduzido e capacidade de adaptação a diferentes cenários ([Zou, Yu e Zhang \(2022\)](#), [Ramirez e Ortega \(2020\)](#), [Jensen e Nielsen \(2021\)](#), [Zhang, Li e Wang \(2023\)](#)).

1.3 Definição do Problema

A aplicação de robôs no ambiente industrial não é simplesmente uma questão resolvida ou uma questão de integração superficial. Pelo contrário, ela apresenta desafios fascinantes que abrangem uma ampla área de Pesquisa e Desenvolvimento (P&D), com potencial para gerar avanços tecnológicos significativos. Estes desafios surgem principalmente da necessidade de estabelecer uma interação eficaz entre os operadores humanos e os sistemas automatizados, uma vez que ambos devem coexistir e colaborar harmoniosamente no ambiente fabril. De fato, nas modernas fábricas contemporâneas, é evidente uma integração complexa entre o trabalho humano e o trabalho realizado por máquinas automáticas, tais como robôs manipuladores e móveis, máquinas-ferramenta, autômatos programáveis, bem como sistemas pneumáticos e hidráulicos, entre outros.

Essa realidade apresenta desafios significativos no que diz respeito aos dispositivos e software de interface homem-máquina, os quais ainda não foram completamente superados e são atualmente alvo de intensa Pesquisa e desenvolvimento (P&D).

Alguns dos principais problemas inerentes aos manipuladores são demonstrados na

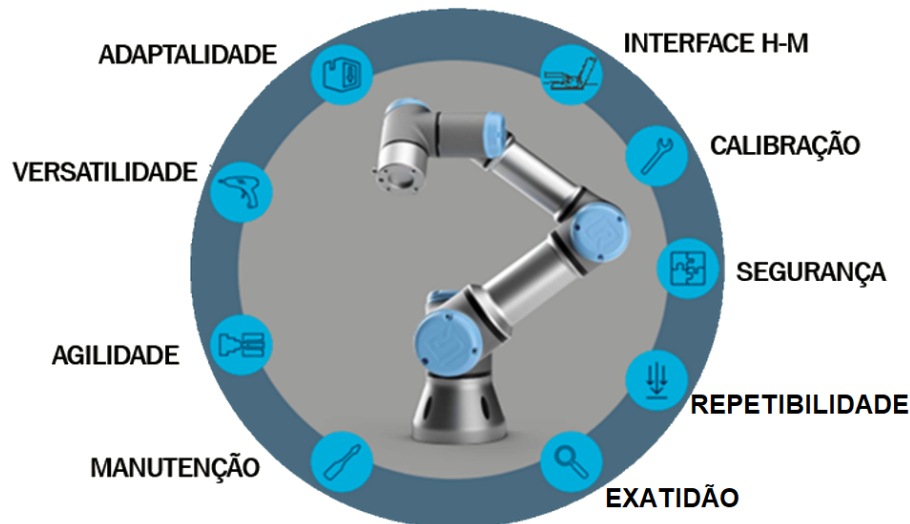


Figura 1.1 – Problemas inerentes aos manipuladores. Fonte: Elaboração própria.

figura 1.1. Robôs industriais apresentam erros inerentes de precisão. Para garantir trabalhos eficientes e de alta qualidade com programação off-line, a calibração de robôs é um método eficaz e econômico para melhorar os erros de posicionamento [Kjellander e Rahayem \(2009\)](#). A calibração visa melhorar a precisão da posição do efetuador final do robô, considerando discrepâncias do modelo nominal no controlador do robô e das especificações do produto, que podem ignorar erros decorrentes do processo de fabricação ou montagem. O processo de calibração geralmente envolve quatro etapas: modelagem, medição, identificação e compensação ([Kjellander e Rahayem \(2009\)](#), [Abderrahim et al. \(2004\)](#), [Dombre e Khalil \(2007b\)](#), [Shiakolas, Conrad e Yih \(2002\)](#), [Roth, Mooring e Ravani \(1987\)](#), [Schröer \(1993\)](#)).

No contexto industrial, a calibração robótica é um aspecto crítico para melhorar a precisão e a eficiência operacional. Métodos tradicionais de calibração, como aqueles baseados em cinemática direta e inversa, enfrentam desafios relacionados à integração e ao custo ([Dombre e Khalil \(2007a\)](#)). Assim, o desenvolvimento de sensores visuais acoplados a manipuladores robóticos emerge como uma alternativa promissora, combinando alta precisão, flexibilidade e viabilidade econômica ([Chen e Smith \(2000\)](#)).

É crucial simplificar, tornar mais intuitiva, menos formal e, acima de tudo, mais segura a interface entre humanos e máquinas. Além disso, o fator humano, devido à sua imprevisibilidade, pode apresentar desafios complexos para os sistemas automáticos, especialmente em contextos nos quais a interrupção frequente não é viável e altos padrões de segurança devem ser mantidos constantemente. Isso ressalta a importância crítica das soluções de interface homem-máquina para o sucesso de sistemas automáticos no ambiente industrial.

Um desafio significativo adicional é a busca pela agilidade. Isso implica aproveitar de maneira eficaz a flexibilidade inerente aos modernos equipamentos de automação industrial. Esta não é uma tarefa simples e requer uma exploração meticulosa dos diversos equipamentos

disponíveis, com a distribuição inteligente de tarefas e coordenação de seus desempenhos. A agilidade tornou-se um fator crítico nos dias de hoje, considerando a rápida obsolescência dos produtos e a necessidade de sistemas produtivos capazes de se adaptar à produção de uma variedade de produtos e modelos diferentes.

Atualmente, os manipuladores precisam ser altamente programáveis e versáteis, equipados com diversos mecanismos de interface com outros equipamentos. Essa característica confere a eles uma flexibilidade excepcional, permitindo que se adaptem a uma ampla gama de tarefas. Essa flexibilidade aumenta a capacidade desses equipamentos de se ajustarem rapidamente a mudanças nas demandas de produção e operações, tornando-se essencial para responder de forma ágil às variações do mercado e à introdução de novos produtos.

Outro desafio crucial é desenvolver sensores que permitam aos robôs uma maior adaptabilidade às mudanças no processo produtivo. Cada etapa da fabricação realizada por um robô industrial pode requerer ajustes para a produção de um produto diferente, ou até mesmo para a fabricação de um produto completamente novo. Criar sensores capazes de se adaptar a essa nova programação representa um desafio significativo.

A manutenção regular e as práticas de segurança são elementos essenciais para o funcionamento eficaz dos manipuladores robóticos (com prevenção de falhas, aumento de vida útil, melhoria na eficiência e redução de custos). Elas não apenas aumentam a eficiência e a longevidade dos equipamentos, mas também garantem um ambiente de trabalho seguro e produtivo. Investir em programas de manutenção e segurança é, portanto, fundamental para qualquer indústria que dependa desses sistemas robóticos avançados (proteção dos trabalhadores colaborativos, conformidade com regulamentações, minimização de riscos de acidentes, estabilidade operacional e proteção de investimentos).

Este trabalho propõe o desenvolvimento de um sensor de visão ativa inovador para orientar e posicionar manipuladores robóticos em tarefas industriais. Este sensor se diferencia dos sistemas convencionais ao combinar medição de profundidade em tempo real com maior resolução e robustez, além de apresentar custos reduzidos quando comparado a sensores mecânicos ou sistemas baseados exclusivamente em tecnologia laser. Esse sensor busca preencher lacunas das tecnologias existentes, proporcionando uma solução mais acessível, robusta e precisa para aplicações em ambientes dinâmicos. Além disso, este sistema não apenas melhora a precisão do posicionamento, mas também permite a adaptação rápida a mudanças nos produtos, eliminando a necessidade de ajustes manuais dispendiosos e prevenindo colisões (Kjellander e Nilsson (2009)).

1.4 Hipóteses do Trabalho

Na literatura apresentada anteriormente, observa-se a existência de diversos trabalhos correlatos a esta dissertação; no entanto, ainda há uma lacuna significativa no desenvolvimento de controle servo-visual de manipuladores robóticos, especialmente para robôs

com seis graus de liberdade, como o IRB140 da ABB. Com base nesse contexto, as seguintes hipóteses podem ser formuladas:

a) A instalação de um sensor de visão acoplado ao efetuador do robô permitirá um direcionamento mais preciso nas tarefas a serem executadas, sugerindo que a visão computacional pode servir como um "olho" para o robô, melhorando sua eficiência operacional.

b) A obtenção da distância de forma satisfatória entre o sensor de visão e o alvo, por meio de um algoritmo capaz de medir a profundidade da imagem em relação ao centro óptico da câmera, utilizando laser, pode demonstrar a viabilidade do controle servo-visual baseado em imagem.

c) O desenvolvimento de um algoritmo avançado para gerar velocidades e orientações necessárias para que o robô alcance o alvo final pode demonstrar que o controle servo-visual baseado em imagem é eficaz em ambientes dinâmicos.

Com base nessas hipóteses, a questão de pesquisa emergente é: de que maneira a utilização exclusiva de sensores de visão com retroalimentação de sinal de imagem impacta a eficácia do controle de manipuladores robóticos em ambientes dinâmicos e em diferentes tipos de tarefas? Esta análise considera a necessidade de alta resolução dos sensores, algoritmos avançados de visão computacional e uma arquitetura de controle adaptativa capaz de interpretar e reagir rapidamente às variações no ambiente e nas condições operacionais.

1.5 Objetivos

Desenvolver um sistema de controle servo-visual baseado em imagem, utilizando sensores de visão ativa, para melhorar a precisão de posicionamento e orientação de manipuladores robóticos em ambientes industriais dinâmicos. O sistema deverá alcançar pontos a distâncias constantes do centro do alvo, garantindo posição e orientação determinadas, com alta exatidão e repetibilidade de medição robótica. O objetivo é atingir erros inferiores a 1 mm no posicionamento e aumentar a eficiência em pelo menos 30% em relação a métodos convencionais, atendendo aos requisitos de alta precisão, repetibilidade e custo reduzido.

1.5.1 Objetivos Específicos

Os objetivos específicos e derivados desse trabalho podem ser apresentados como:

- Projetar e produzir um sensor de visão ativa acoplado ao manipulador robótico IRB 140, com capacidade de medir profundidade e corrigir erros de posicionamento em tempo real;
- Desenvolver algoritmos para controle servo-visual, utilizando a técnica IBVS, que assegurem a precisão do posicionamento com erros inferiores a 1 mm. A técnica IBVS se destaca por operar diretamente no espaço de imagem, permitindo maior robustez a ruídos, menor sensibilidade a erros de calibração e uma resposta mais rápida em comparação a

técnicas baseadas no espaço cartesiano, como Position-Based Visual Servoing (PBVS);

- Validar o desempenho do sistema em experimentos práticos, demonstrando a viabilidade do sensor proposto em termos de custo, robustez e eficiência;
- Avaliar o impacto econômico e operacional da solução desenvolvida, destacando seus benefícios em relação às tecnologias tradicionais de calibração;
- Assegurar movimentos repetitivos e precisos com o sensor de visão recebendo sinal de feedback contínuo em tempo real.

1.6 Estrutura do Texto

No [capítulo 2](#), são apresentados os principais trabalhos relacionados ao tema desta dissertação, oferecendo um panorama das literaturas e pesquisas relevantes. O [capítulo 3](#) aborda as fundamentações teóricas essenciais que sustentam o desenvolvimento do estudo.

O [capítulo 4](#) inicia com a descrição detalhada dos equipamentos e componentes utilizados na construção do sensor de visão. Na sequência, são apresentados o desenvolvimento do sensor e a metodologia adotada, incluindo os aspectos matemáticos fundamentais da robótica relacionados à movimentação do manipulador IRB140.

Os experimentos realizados e os respectivos resultados obtidos são discutidos no [capítulo 5](#), destacando as contribuições e observações feitas durante o estudo. Finalmente, o [capítulo 6](#) conclui o trabalho com a exposição das principais conclusões e sugestões para estudos futuros.

Além disso, os apêndices [A](#) e [B](#) complementam a estrutura do texto, fornecendo informações adicionais e suporte técnico diretamente relacionados ao desenvolvimento deste trabalho.

2 BIBLIOMETRIA COM LITERATURAS RELACIONADAS

A visão computacional vem sendo utilizada há mais de setenta anos. No entanto, a literatura ainda apresenta poucas abordagens focadas no uso da visão em robôs, especialmente no controle em malha fechada de manipuladores robóticos. Esta dissertação tem como objetivo desenvolver um sensor de visão computacional para o controle de posição e orientação final de um manipulador específico da ABB, embora o procedimento possa ser aplicado a qualquer robô.

2.1 Palavras Chaves para Pesquisa Bibliométrica

Nessa dissertação é abordado o estado da arte em dois principais campos: visão computacional e robótica, além de explorar a interconexão entre esses domínios (figura 2.1). Portanto, foram escolhidas as seguintes frases:

1. Visão computacional;
2. Uso de visão em robôs;
3. Sensor de visão ativa;
4. Câmera no efetuador;
5. Controle em malha fechada de manipuladores robóticos;
6. Sensor de visão computacional;
7. Controle de posição e orientação final do manipulador;
8. Cinemáticas de manipuladores robóticos;
9. Medição da posição do manipulador;
10. Calibração Robótica;
11. Controle servo-visual baseado em imagem.

Sendo que a pesquisa foi realizada com as respectivas frases traduzidas para o inglês:

1. *Computer vision*;
2. *Vision for robots*;
3. *Active Vision Sensor*;
4. *Eye-in-hand*;
5. *Closed-loop control of robotic manipulators*;
6. *Computer vision sensor*;

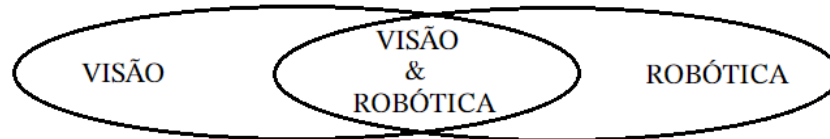


Figura 2.1 – Conjunto de palavras chaves para busca.

7. *Position control and manipulator end-effector orientation;*
8. *Kinematics of robotic manipulators;*
9. *Measurement of Manipulator Position;*
10. *Robot Calibrator;*
11. *Image-Based Visual Servoing.*

A seguir são mostrados resultados em tabelas com utilização de lógica booleana para busca nas revistas científicas *SCOPUS* e *WEB OF SCIENCE*.

Utilizando o programa "VOSviewer", foi possível destacar informações importantes dessas buscas bibliométricas por meio de gráficos. A [figura 2.2](#) e [figura 2.3](#) apresentam a quantidade de trabalhos publicados por ano, enquanto a [figura 2.4](#) e [figura 2.5](#) exibem mapa de calor para intensidade de citações para as revistas *SCOPUS* e *WEB OF SCIENCE*, respectivamente.

Tabela 2.1 – Resultado para artigos na base *SCOPUS*

Resultado para busca <i>SCOPUS</i>				
Número de documentos	Intervalo ano de publicação	Textos em Português	Textos em Inglês	Textos em outras línguas
1708	1980-2023	1	1643	64

Tabela 2.2 – Resultado para artigos na base *WEB OF SCIENCE*

Resultado para busca <i>WEB OF SCIENCE</i>				
Número de documentos	Intervalo ano de publicação	Textos em Português	Textos em Inglês	Textos em outras línguas
1326	1990-2023	0	1322	4

2.2 Tabela comparativa de temas relacionados

As palavras-chave utilizadas nas pesquisas identificaram, entre 1980 e 2023, um total de 1.708 documentos publicados na base *SCOPUS* e mais 1.326 documentos na base *WEB OF SCIENCE*, entre 1980 e 2024.

Documents by year

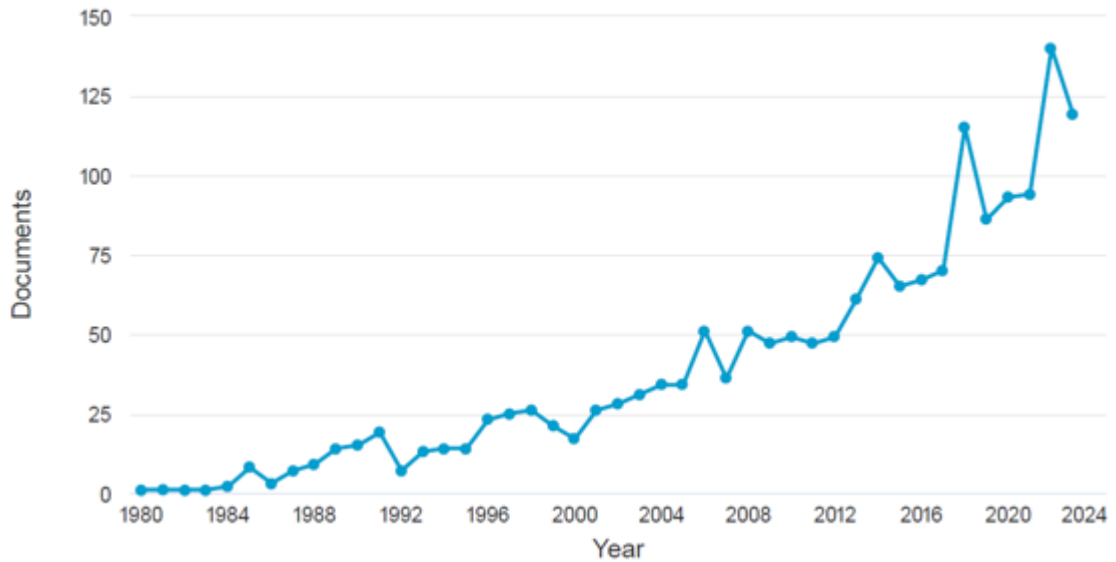


Figura 2.2 – Publicações por ano - base SCOPUS.

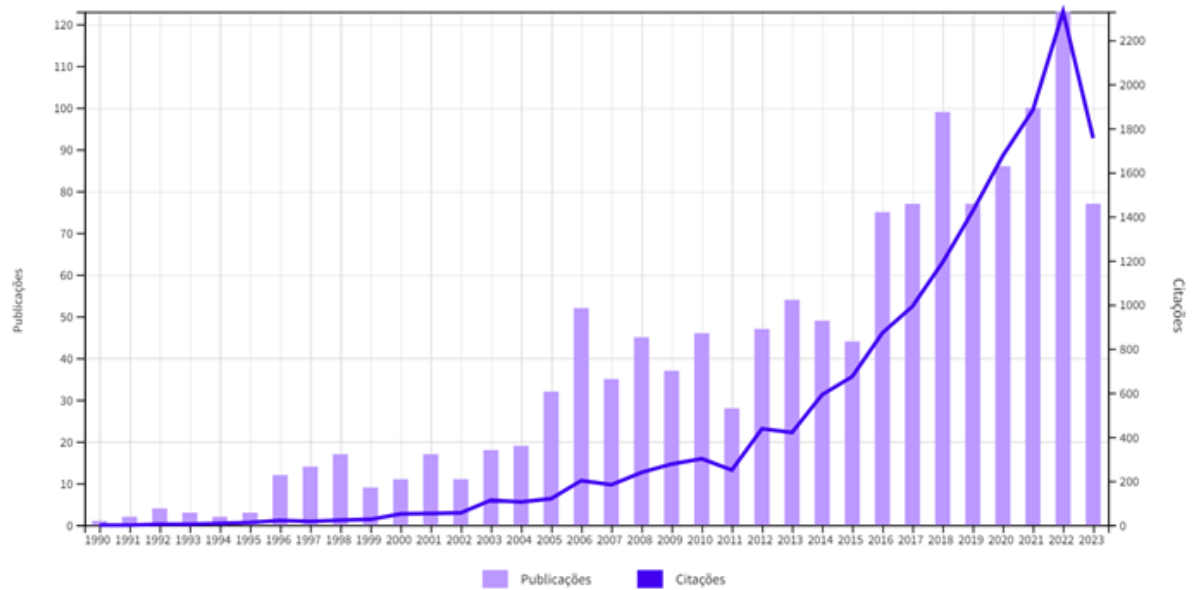


Figura 2.3 – Publicações e citações por ano - base WEB OF SCIENCE.

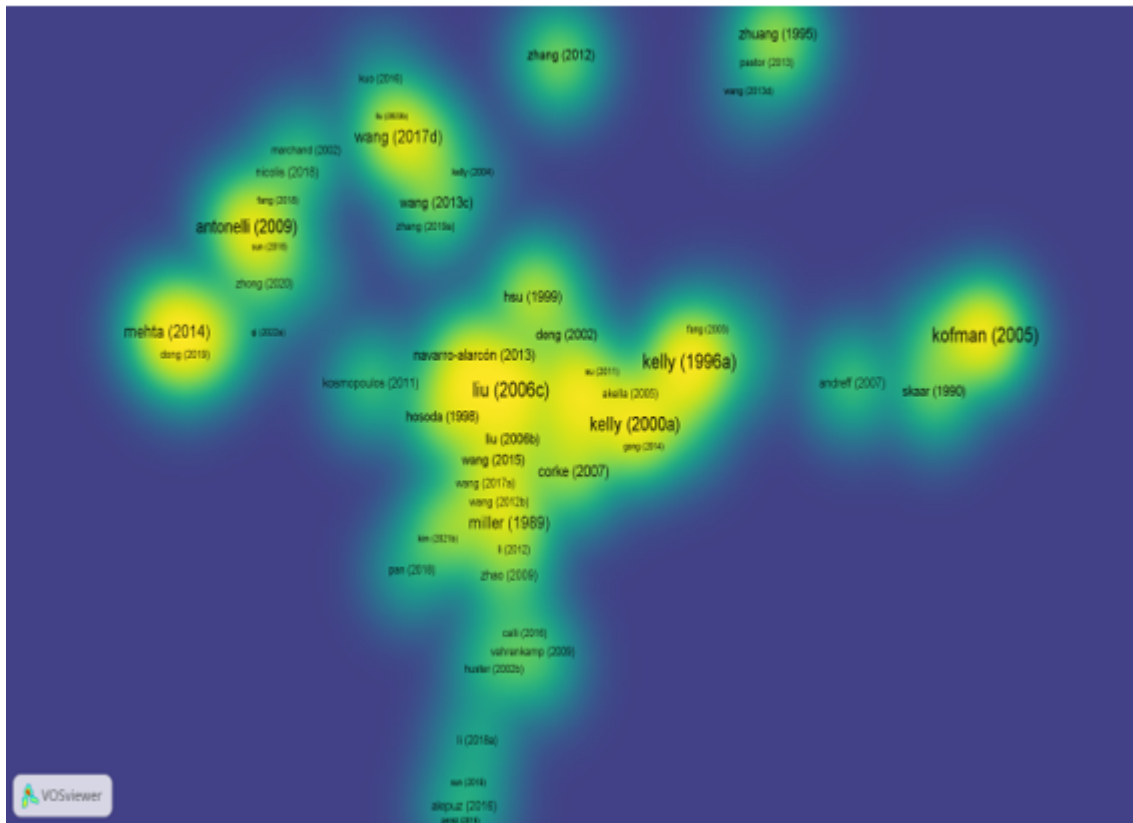


Figura 2.4 – Intensidade de citações em mapa de calor - base *SCOPUS*.

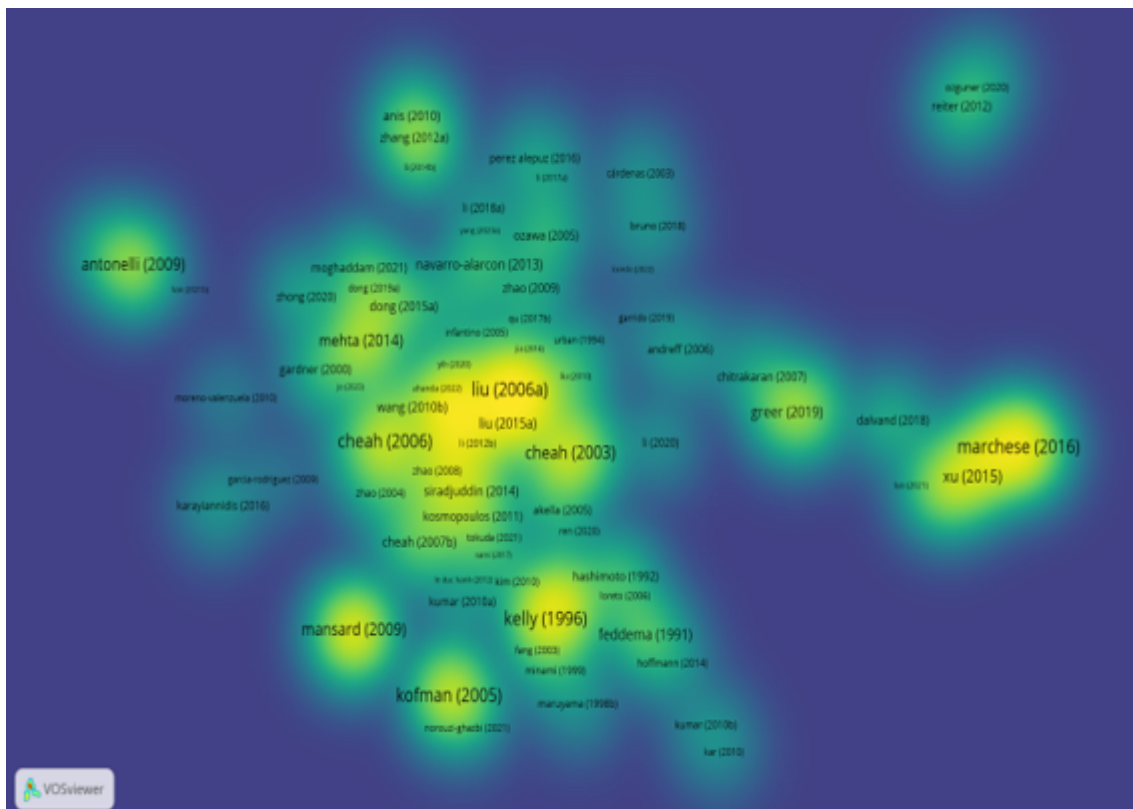


Figura 2.5 – Intensidade de citações em mapa de calor - base *WEB OF SCIENCE*.

Entre os artigos analisados, existem tópicos que, em diferentes níveis, estão relacionados à proposta aqui apresentada. Para identificar de forma clara o limiar do conhecimento entre o trabalho proposto e as pesquisas previamente publicadas, as abordagens empregadas nos estudos foram utilizadas como atributos nas tabelas apresentadas. Esses atributos serviram como critérios para a seleção de publicações que compartilham metodologias, estratégias ou resultados pertinentes ao tema em questão.

Dado o vasto universo de publicações, caracterizado por uma diversidade de temas e abordagens, foi essencial estabelecer parâmetros que permitissem distinguir as contribuições mais relevantes e alinhadas ao escopo deste trabalho. A análise dessas abordagens possibilitou uma síntese eficiente das características metodológicas e resultados apresentados nos estudos.

Com base nesse levantamento, as tabelas foram estruturadas para organizar e apresentar os tópicos correlacionados de maneira clara e objetiva. Essas tabelas sintetizam as informações, destacando os pontos em comum entre os trabalhos e as contribuições do presente estudo, além de levar em conta a relevância e o impacto dos artigos no campo, mensurado, por exemplo, pelo número de citações.

2.2.1 Parâmetros para Visão Computacional

As qualidades definidas neste estudo para a visão computacional representam características intrínsecas ao modelo concebido para o controle completo do manipulador robótico proposto. Assim, a [tabela 2.3](#) destaca os fatores essenciais da pesquisa. A coluna "Programa Utilizado" na tabela indica qual software foi empregado para desenvolver os algoritmos de visão computacional. A categoria "Obtenção de Alvo" descreve a forma do alvo utilizado para orientar o movimento do robô. A configuração do alvo é crucial para que o robô alcance sua posição e orientação finais. A maneira como o robô "percebe" o alvo neste estudo determina tanto o movimento de translação quanto a rotação que o robô precisa realizar para alcançar seu estado final. As colunas subsequentes, "Posicionamento da Câmera" e "Tipo de Sistema de Controle Servo-visual (TSCSV)", resumem quais estudos na literatura apresentada estão mais alinhados com as lacunas de conhecimento identificadas neste trabalho e, portanto, merecem maior atenção.

2.2.2 Atributos para Robótica

Uma segunda etapa mostrada na [tabela 2.4](#) foi definida para os quesitos de manipuladores robóticos. As colunas da [tabela 2.4](#) serão usadas para definição da arte dentro da área de robótica. A coluna iniciada por "Robô" define qual marca e modelo de robô foi utilizado para a pesquisa. O termo "Número de juntas" indica a quantidade de juntas do robô utilizadas para o cálculo da cinemática direta e inversa do manipulador. A coluna "Algoritmo Característico" se refere ao código peculiar do robô utilizado. E por fim a coluna definida

por “Graus de Liberdade” se referem a liberdade de movimentação possíveis do robô nas três direções do espaço x , y , z e rotações em torno desses eixos. O grau de liberdade robótico está intimamente ligado ao código computacional de controle deste trabalho e também deve ser dada maior ênfase aos trabalhos correlatos obtidos na pesquisa bibliométrica.

2.2.3 Parâmetros Construtivos

É importante incluir uma terceira tabela para ilustrar a interação entre visão e robótica, bem como alguns equipamentos essenciais empregados no processo de construção do sensor. A [tabela 2.5](#) tem em suas colunas os seguintes textos: “Modelo da Câmera”; “Hardware Utilizado” e “Sistema de Medição de Profundidade da Imagem (SMPI)”. O primeiro termo “Modelo da Câmera” é para obter qual câmera foi utilizada no trabalho. O segundo termo “Hardware Utilizado” define diretamente a velocidade entre as plataformas utilizadas. O Hardware pode ser um computador com todas suas características (processador, memória, placa mãe, programa operacional, etc) ou mesmo um microcontrolador embarcado no robô. O termo final “Sistema de Medição de Profundidade da Imagem” define a distância entre o alvo e a câmera, sendo fundamental para o processo de controle servo-visual.

Dois desses trabalhos merecem destaque. O primeiro focado em dinâmica robótica, realizado por [Cheah et al. \(2006\)](#), utiliza um Jacobiano adaptativo para o rastreamento de trajetórias de robôs com cinemática e dinâmica incertas, de modo a garantir que o efetuador do robô converja para a trajetória desejada. Foram realizados dois experimentos com resultados bastante satisfatórios: o primeiro utilizando sombreamento para o rastreamento de trajetória e o segundo, um percurso circular.

O segundo trabalho focado em visão computacional, [SS \(2014\)](#), realizado em 2014, aborda o desenvolvimento de um sistema de estimativa e controle baseado em visão para colheita robótica de frutas, com uma análise rigorosa de estabilidade. Foi desenvolvido um robô específico para a colheita, equipado com um sistema de visão sofisticado capaz de obter a posição 3D das frutas. O destaque deste trabalho é o método utilizado para detecção do alvo tridimensional por meio de uma nuvem de pontos, permitindo a localização precisa no espaço e um rápido posicionamento do efetuador em torno deste alvo.

Tabela 2.3 – Atributos de visão computacional.

Ref.	Ano	Programa Utilizado	Obtenção de Alvo	Posicionamento da Câmera	TSCSV ¹
(R.Kelly ; R. Carelli ; O. Nasisi ; B. Kuchen ; F. Reis)	2000	Próprio	Alvo Estático	Efetuator Final	PBVS ²
(R.Kelly)	1996	Próprio	Alvo Estático	Fixa	PBVS ²
(Yun-Hui Liu; Hesheng Wang; Chengyou Wang; Kin Kwan Lam)	2006	Próprio	Alvo predefinido	Fixa	IBVS ³
(CC Cheah; M. Hirano; S. Kawamura; S. Arimoto)	2003	Próprio	Alvo Estático	-	-
(HeshengWang; Yun Hui Liu; Dong Xiang Zhou)	2008	Próprio	Ponto e Linha	Efetuator Final	IBVS ³
(WT Miller)	1989	Própria com uso de rede neural e interface VAX	-	Efetuator Final	PBVS ²
(CC Cheah; C. Liu; JJE Slotine)	2006	Próprio	Objeto Real	Fixa	PBVS ²
(SS Mehta; TF Burks)	2014	Matlab	Objeto Real	Fixa	PBVS ²
(J. Kofman; Xianghai Wu; TJ Luu; S.Verma)	2005	Criação de Interface movimento homem-robô em C++	Objeto Real	Várias câmeras Fixas	PBVS ²
(Mahdi Aziziano; Mahta Khoshnam; Nima Najmaei; Rajni V . Patel)	2014	OpenCV Matlab C++ (generalizado para imagem e robótica médica)	Objeto fixo; Alvo predefinido; Objeto Real Imersivo	Efetuator Final e Câmera fixa	PBVS ² e IBVS ³
Nossa Proposta		Matlab	Alvo predefinido	Efetuator Final	IBVS ³

- (1) TSCSV - Tipo de Sistema de Controle Servo-visual.
(2) PBVS - Controle Servo-Visual Baseado em Posição.
(3) IBVS - Controle Servo-Visual Baseado em Imagem.

Tabela 2.4 – Atributos robóticos.

Ref.	Ano	Robô	Número de juntas	Algoritmo Característico	Graus de Liberdade
(R.Kelly; R. Carelli; O. Nasisi; B. Kuchen ; F. Reis)	2000	Criação própria	2	próprio	2
(R.Kelly)	1996	Criação própria	3	próprio	3
(Yun-Hui Liu; Hesheng Wang; Chengyou Wang; Kin Kwan Lam)	2006	Criação própria	3	próprio	3
(CC Cheah; M. Hirano; S. Kawamura; S. Arimoto)	2003	Criação própria porém metodologia pra qualquer robô	n	próprio	n
(HeshengWang; Yun Hui Liu; Dong Xiang Zhou)	2008	Criação própria	3	próprio	3
(WT Miller)	1989	Criação própria	4	próprio	6
(CC Cheah; C. Liu; JJE Slotine)	2006	Sony SCARA	4	-	3
(SS Mehta; TF Burks)	2014	K1207	6	Próprio	6
(J. Kofman; Xianghai Wu; TJ Luu ; S.Verma)	2005	Thermo CRS	6	CRS Active Robo	6
(Mahdi Aziziano; Mahta Khoshnam; Nima Najmaei; Rajni V. Patel)	2014	Generalizado	6	Generalizado	6
Nossa Proposta		ABB – IRB140	6	RAPID	6

Tabela 2.5 – Atributos construtivos.

Ref.	Modelo da Câmera	Hardware Utilizado	SMPI ¹
Kelly <i>et al.</i> (2000)	Apresentado: Foco, distorções	Computador conf. específicas	-
Kelly (1996)	Apresentado: Foco, distorções	Computador conf. específicas	-
Liu <i>et al.</i> (2006)	Ptgrey (sem necessidade de parâmetros da câmera)	Computador conf. específicas	Matriz de interação independente de profundidade
Cheah <i>et al.</i> (2003)	Apresentado: Foco, distorções	Computador conf. específicas e operacional Linux	-
Miller (1989)	-	Embutido com microcontrolador TMS32010	-
Cheah <i>et al.</i> (2006)	PSD Hamamatsu	Computador conf. específicas	-
SS (2014)	KT&C, KPCS20-CP1 focal length of 4.3 mm	Computador conf. específicas	Transformação de Perspectiva
Kofman <i>et al.</i> (2005)	Watec America Corporation	Computador conf. específicas	Triangulação com câmeras estéreo
Azizian <i>et al.</i> (2014)	Generalizado	Computador conf. específicas	Generalizado
Nossa Proposta	Basler acA2500 - 14 μ m e Lente CFFL F1.4 f16mm 2/3"	Computador conf. específicas	Triangulação com utilização de lazer na imagem

(1) SMPI - Sistema de Medição de Profundidade da Imagem.

3 FUNDAMENTAÇÃO TEÓRICA E BIBLIOGRÁFICA

Neste capítulo, serão elucidados os conceitos fundamentais de óptica e visão computacional relevantes para este estudo, assim como a base teórica do controle servo-visual baseado em imagem (IBVS). Além disso, serão abordados aspectos relacionados à cinemática e velocidade robótica, com referências provenientes de trabalhos disponíveis na literatura especializada. Esta exposição será conduzida por meio de uma revisão concentrada nos domínios da visão, robótica e suas variações.

3.1 Visão Computacional

Grandes esforços têm sido realizados para dotar as máquinas da habilidade de visão humana, um dos sentidos mais complexos e que tem sido reproduzido para aplicação em sistemas robóticos e de automação. Embora sensores de pressão, proximidade, torque e força desempenhem um papel importante na melhoria do desempenho de sistemas automatizados, a visão computacional é amplamente reconhecida como o sistema de sensoriamento com maior capacidade para medir e extrair informações, conforme afirma [Fu et al. \(1987\)](#).

3.1.1 Aquisição de Imagens

Uma imagem digital não é interpretada da mesma forma que um ser humano identifica uma cena, mas a partir de atributos que são extraídos da mesma e que estão relacionados entre si. Portanto o processo de tratamento de imagens, busca-se capturar algumas características e usá-las para minimizar o espaço de pesquisa, conforme [Yihong, Hongjiang e Chuan \(1994\)](#).

Utilizar-se de técnicas computacionais para estimar ou tornarem explícitas propriedades geométricas ou dinâmicas de imagens do ambiente que nos cerca, pode ser aplicada tanto na robótica fixa, quanto na móvel, em processos de medição, localização e identificação de objetos e determinação de suas formas geométricas, bem como no auxílio da navegação de robôs com o reconhecimento do ambiente, estimativa de velocidade, aceleração dentre outros.

A associação de inteligência e flexibilidade deste sistema, permite que a resposta se adeque de acordo com as variações do meio onde está implementado. Os três principais níveis, que auxiliam o uso da visão computacional mediante a área são:

- a) baixo Nível:
 - Percepção \Rightarrow Aquisição da imagem;

- Pré-Processamento \Rightarrow Redução de ruído e/ou realce de detalhes.

b) médio nível:

- Segmentação \Rightarrow Particionar a imagem em regiões de interesse;
- Descrição \Rightarrow Determinar propriedades de objetos (tamanho, forma, etc.).

c) alto nível:

- Reconhecimento \Rightarrow Identificação de objetos a partir de um banco de dados.

O sistema de visão computacional é dividido em duas partes: hardware que é composto por equipamentos e circuitos eletrônicos destinados à aquisição, digitalização e armazenamento das imagens, e do software que possibilita a extração de informações desejadas. É possível que o processamento da imagem seja realizado no próprio hardware com uso de circuitos programáveis, o que proporciona respostas mais rápidas e aumenta a eficiência do processo. Este tipo de sistema não será abordado neste trabalho.

Para o uso de um sistema de visão computacional é necessária uma atenção especial com o ambiente ao qual será extraída a imagem, para que o custo computacional de processamento seja o menor possível no que diz respeito à quantidade de dados a serem computados, e de forma a ter uma maior velocidade de processamento. O uso de câmeras de alta definição, iluminação adequada, boa focalização e posicionamento do objeto são fatores relevantes para boa estruturação da cena. Como comparativo, em aplicações em ambientes móveis, onde não é possível planejar a cena que será capturada, uma vez que, este é dinâmico, não se pode prever o que será detectado na cena, com isso, será exigido um algoritmo mais complexo que utilize ferramentas computacionais mais poderosas ou com uso de sistema multifocais.

Uma câmera colorida tem menor sensibilidade do que as monocromáticas, porque precisam processar mais cores para gerar a imagem, portanto precisam de mais luz para gerar boas imagens. As câmeras com sensores de imagem funcionam de maneira similar a câmeras com filme. Em ambas, a luz é focalizada por uma lente (ou conjunto delas) e incide sobre o sensor de imagem sensível à luz CCD ou CMOS, que convertem a luz em elétrons através do efeito fotoelétrico conforme afirma [Litwiller \(2001\)](#).

Para tornar possível a análise computacional sobre imagens, é necessário que os dados estejam na forma digital, isto é, a luminosidade deve ser convertida em sinais elétricos distribuídos espacialmente. Para que o processo de aquisição de imagem seja bem compreendido é necessário tratar da imagem em si e suas formas de representação.

O fotodetector é capaz de transformar luz (fótons) que são sinais ópticos em sinais elétricos e pode ser construído utilizando materiais semicondutores. O dispositivo necessário para a aquisição da imagem é um elemento físico sensível à energia luminosa, ou seja, a câmera, é um transdutor que é responsável por transformar a energia luminosa em sinal

elétrico. Este sinal de saída é proporcional à intensidade luminosa incidente no dispositivo fotossensível. A sensibilidade da câmera é uma unidade fotométrica usada para medir o nível de iluminação, e é definida como sendo a iluminação sobre uma superfície quando o fluxo luminoso de 1 lúmen incide sobre uma área de 1 metro quadrado, é também conhecido por candela, e está diretamente relacionada com a quantidade de luz mínima necessária para garantir a qualidade de imagem. Porém, é aconselhável que o local a ser instalado a câmera tenha luminosidade aceitável para os olhos, assim o transdutor não irá apresentar nenhum problema.

Para aplicações de visão computacional, a otimização do contraste por meio da iluminação é essencial para melhorar a qualidade da imagem, permitindo a distinção clara entre objetos e seus fundos. A iluminação eficaz gera o contraste necessário entre as características do alvo e as áreas circundantes, possibilitando que os sistemas detectem, inspecionem e analisem imagens com maior precisão. De acordo com estudos recentes (Ariateja, Ardiyanto e Soesanti (2018), Ma *et al.* (2018), Sun *et al.* (2005), Zhao e Wan (2017)), uma boa iluminação é crucial para a visão computacional, pois melhora a visibilidade e minimiza problemas como brilho e sombras, especialmente em superfícies variadas. Em contextos específicos, o uso de luzes de diferentes cores, como vermelho ou verde, pode reduzir reflexos em superfícies brilhantes ou aprimorar características particulares do material.

Os sistemas modernos de visão computacional, incluindo aqueles voltados para inspeção e controle de qualidade, enfatizam a importância da iluminação uniforme para reduzir o impacto do ruído de fundo e garantir a consistência no contraste. A seleção cuidadosa da cor e do ângulo da luz pode destacar características do objeto e minimizar sombras, o que é especialmente benéfico para superfícies reflexivas ou texturizadas. Essas estratégias ajudam a manter o contraste adequado, crucial para a obtenção de resultados precisos na visão computacional.

3.1.2 Digitalização de Imagem

Na fotografia digital a imagem formada pela objetiva dentro da câmera não atinge uma emulsão fotossensível (o filme em câmeras antigas) e sim o dispositivo eletrônico sensível à luz, que é responsável por interpretar os impulsos luminosos da imagem quanto à sua intensidade (luminância) e coloração (crominância). Estas informações são codificadas de forma digital e armazenadas numa memória temporária (*buffer memory*) e posteriormente enviadas para um dispositivo de armazenagem (*memory stick*, *hard disk*, ou outro).

O sensor fotossensível é composto por milhares de elementos sensíveis a luz, e a imagem formada sobre ele é dividida em vários elementos de imagem, chamados pixels. Cada pixel contém informações correspondentes a aquela área da imagem. O princípio de funcionamento pode ser explicado por analogia a um tabuleiro de damas, onde cada quadrado representa um pixel, e em cada ponto existe uma informação. Quando a imagem

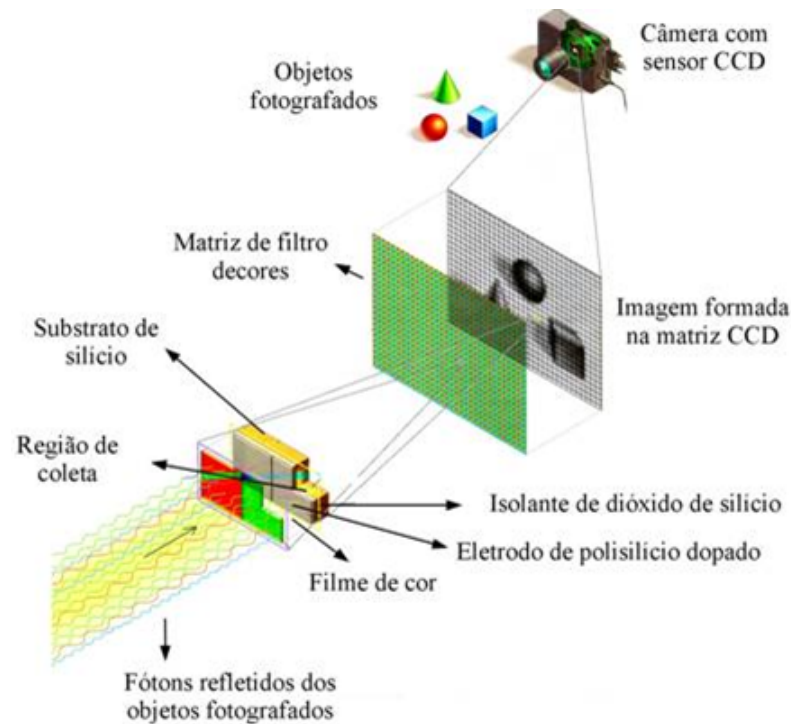


Figura 3.1 – Aquisição e digitalização de imagem (coppe (2024)).

está colocada sobre a superfície do tabuleiro, cada quadrado é lido de forma sequencial por um circuito eletrônico que faz a conversão dos dados analógicos para digitais, A/D e transforma o valor de cada pixel em um valor digital por meio da quantidade de carga de cada pixel e converte essa medição para a forma binária, segundo [Gonzalez, Woods e Eddins \(2004\)](#).

A [figura 3.1](#) mostra o esquema de um sistema de aquisição de imagem. Pode-se perceber que o sensor é uma matriz de elementos fotossensíveis. Assim a imagem de saída do sensor é uma discretização da cena do mundo real. Quanto maior for o número de elementos do sensor, melhor será a resolução da imagem obtida, já que cada elemento irá resultar em um pixel. Havendo uma maior quantidade de elementos significa que uma menor parte da cena do mundo será representada por um único valor de nível de cinza, fazendo com que não se perceba a transição entre um pixel e outro.

A resolução refere-se ao número de linhas e colunas na imagem formada na matriz CCD, sendo determinada pela câmera, em que as de varredura de área sua resolução é expressa como linhas e colunas de pixels, como por exemplo 640x480, já no caso de varredura por linha sua resolução é expressa apenas por colunas de pixels, as linhas possuem sempre o valor 1.

3.2 Robótica e a Utilização de Câmeras

Ao utilizar uma câmera para estimar a posição e orientação de um objeto em relação a base do manipulador, é crucial ter conhecimento da localização relativa entre seu efetuador e sua base, entre a câmera e o efetuador, e entre o objeto e a câmera. Com essas informações, é possível calcular a posição e orientação do objeto em relação à base do robô, que geralmente serve como referência para o sistema de coordenadas global. Como ilustrado na [figura 3.2](#), as tarefas que envolvem as estimativas dessas três relações são conhecidas respectivamente como calibração de juntas do manipulador, calibração efetuador-câmera e calibração da câmera.

Para posicionar a câmera em um robô manipulador de forma eficaz, estudos recentes recomendam o uso de estratégias baseadas em otimização da perspectiva e redução da incerteza na calibração, especialmente para configurações *eye-in-hand* onde a câmera é fixada diretamente no efetuador final do robô. Trabalhos como o de [Zhou et al. \(2023\)](#) exploram técnicas de calibração ativa para maximizar o ganho de informação durante a configuração da câmera, permitindo que o sistema selecione a próxima posição ideal para aprimorar a precisão da estimativa de posição e orientação do alvo. Esta abordagem é especialmente útil para tarefas de inspeção e montagem onde a exatidão é crítica.

Além disso, estudos recomendam posicionar a câmera de forma a minimizar problemas como o desfoque devido ao movimento e alinhar o campo de visão com o interesse específico da aplicação, seja para capturar detalhes de alta precisão em superfícies ou monitorar objetos em movimento rápido. Esse posicionamento otimizado permite que a câmera reduza interferências visuais e maximize o contraste, essenciais para a visão computacional em operações robóticas.

Essas técnicas modernas de calibração e posicionamento são especialmente eficazes em sistemas industriais, onde a câmera precisa estar sincronizada com sensores adicionais para garantir precisão e velocidade de captura adequadas ao ambiente e às necessidades de processamento visual.

Essas calibrações são realizadas por meio do cálculo das transformações nos sistemas de coordenadas de cada elo, levando em conta suas cinemáticas direta e inversa. Isso permite estabelecer uma relação matemática entre as coordenadas do sistema coordenado do efetuador do manipulador robótico e as coordenadas de sua base. Por questões de simplificação do problema e para obter melhor desempenho, a literatura adota o sistema coordenado de origem da base do robô e o de referência global coincidentes. Sempre que mencionado o sistema coordenado global, ele se refere ao sistema de coordenadas de referência fixo, ao qual todas as coordenadas dos pontos de interesse no resultado final estarão referenciadas, conforme [Vianna \(2009\)](#).

Quando o sistema de visão é fixado a um manipulador, em operações de *pick and*



Figura 3.2 – Sistemas de calibração (Fonte: Vianna (2009) adaptado).

place, por exemplo, o objeto não precisa estar sempre na mesma localização (posição e orientação) para que o manipulador possa executar sua tarefa adequadamente. Normalmente, em processos de soldagem, o objeto se encontra fixado sempre na mesma posição. Contudo, é interesse desse trabalho investigar a viabilidade econômica da utilização de um sistema de visão computacional para localização do objeto com a medição de posição e orientação, assim como utilizado nas mencionadas operações de manipulação.

Atualmente, os robôs industriais evoluíram significativamente em termos de capacidade sensorial e inteligência. As gerações atuais de robôs integram sensores de visão avançados, Inteligência Artificial (IA) e aprendizado de máquina, proporcionando maior flexibilidade e adaptabilidade nos processos produtivos (AI Magazine (2022), Google DeepMind (2023)). Diferente dos robôs das gerações anteriores, os modernos podem interpretar mudanças complexas no ambiente e ajustar suas ações em tempo real, permitindo operações mais autônomas e eficientes. Estudos recentes enfatizam a importância de sensores de imagem, algoritmos de visão computacional e IA para aumentar a precisão na manipulação de peças, adaptabilidade a variações dimensionais e melhoria no alinhamento e controle de qualidade em ambientes dinâmicos. Esses avanços transformam os robôs em elementos essenciais para a automação industrial inteligente e adaptável, ampliando sua utilidade em setores como manufatura, inspeção e montagem, onde a habilidade de interpretar e responder a mudanças no ambiente é crítica para o desempenho eficiente.

3.2.1 Controle Servo-Visual

O Controle Servo-visual é uma técnica amplamente adotada para o controle de manipuladores robóticos na atualidade. Essa abordagem utiliza dados de visão computacional coletados em cada ciclo de controle para ajustar continuamente a posição do manipulador até que a posição desejada seja alcançada, configurando, assim, um sistema de malha fechada.

O controle servo-visual visa controlar a posição e orientação do efetuador do robô em relação ao alvo, utilizando informações visuais extraídas da imagem. Aspectos relevantes

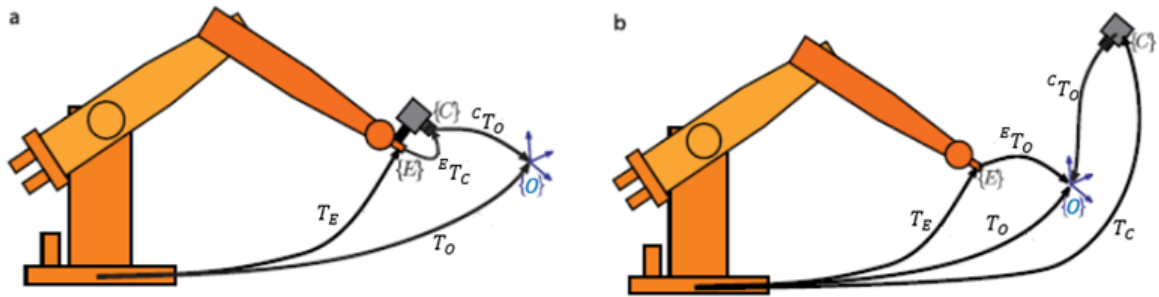


Figura 3.3 – Configurações para o controle servo-visual e as relevâncias para coordenadas globais a partir dos quadros da imagem: efetuador final $\{E\}$, câmera $\{C\}$ e destino $\{O\}$. a) configuração com câmera no efetuador do robô *Eye-in-Hand*; b) Configuração com câmera fixa em coordenadas globais *end-point open-loop*.

a serem considerados neste tipo de controle incluem a posição da câmera em relação ao robô, ou seja, se ela se movimenta junto com o robô ou está fixada em um ponto externo, e o número de câmeras no sistema, que pode ser monocular ou binocular "estéreo". Conforme ilustrado na figura 3.3, a câmera pode ser transportada pelo robô ou fixada em coordenadas globais. A configuração da figura 3.3a tem a câmera montada no efetuador final do robô observando o alvo, e é referida como *end-point closed-loop* ou *Eye-in-Hand*. A configuração da figura 3.3b, por outro lado, tem a câmera em um ponto fixo observando tanto o alvo quanto o efetuador final do robô, e é chamada de *end-point open-loop*. Neste trabalho, discutiremos apenas a configuração *Eye-in-Hand*.

Existem duas abordagens fundamentalmente diferentes para o controle servo-visual: o Controle Servo-Visual Baseado em Posição (PBVS, do inglês *Position-Based visual servoing*) e o Controle Servo-Visual Baseado em Imagem (IBVS, do inglês *image-based visual servoing*).

O PBVS, mostrado na figura 3.4a, utiliza recursos visuais capturados, uma câmera previamente calibrada e um modelo geométrico conhecido do alvo para determinar a posição e orientação do alvo em relação à câmera. Em seguida, o robô se move em direção a essa posição, e o controle é realizado no espaço da tarefa, geralmente expresso como uma transformação homogênea de translação e rotação no espaço \mathbb{R}^6 . Embora existam algoritmos eficazes para estimar a posição e orientação, essa abordagem é computacionalmente exigente e depende significativamente da precisão da calibração da câmera e do modelo geométrico do objeto.

O IBVS, em particular, ilustrado na figura 3.4b, dispensa a etapa de estimativa de posição e orientação, utilizando diretamente os pontos característicos da imagem. Este método opera no espaço de coordenadas da imagem \mathbb{R}^2 , onde a posição e orientação desejadas da câmera em relação ao alvo são implicitamente definidas pelos valores dos pontos característicos da imagem na posição e orientação do alvo. O IBVS representa um desafio significativo devido à natureza altamente não linear dos pontos característicos da imagem, os quais são diretamente influenciados pela posição e orientação da câmera. Nesse contexto, [Toquica e Motta \(2024\)](#) contribuem ao propor uma metodologia para medir a postura de robôs com

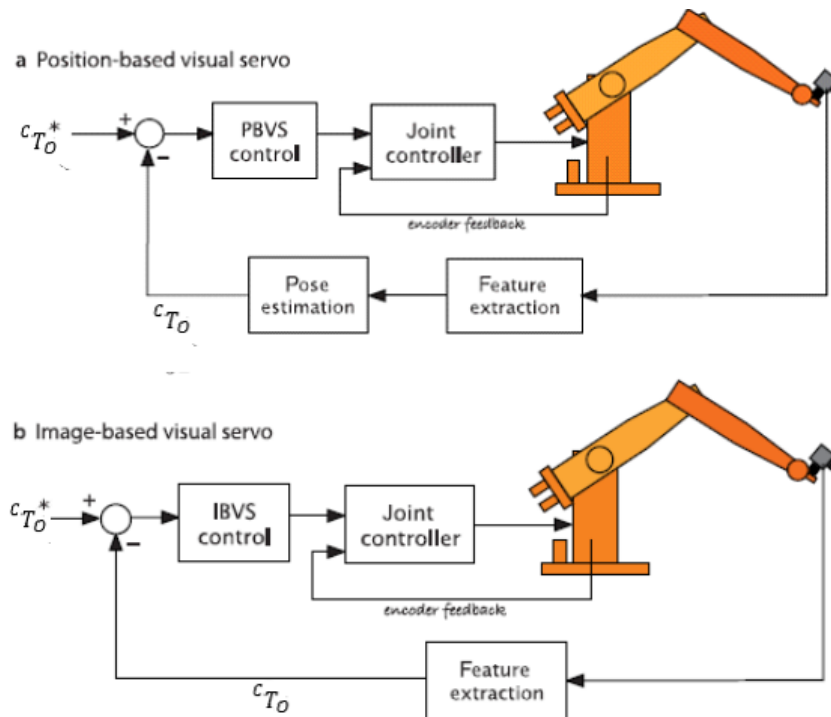


Figura 3.4 – As duas classes distintas para os sistemas de controle servo visual.

precisão em sistemas de calibração, utilizando um sistema óptico automatizado que controla a orientação e a posição em sub-regiões a uma distância constante, facilitando a calibração de robôs em ambientes industriais.

Neste trabalho, a abordagem baseia-se em conceitos de IBVS para controle do manipulador robótico IRB140, mas com uma interface adaptada entre o sistema óptico e o sistema de controle de movimento robótico. Com isso, o sistema visa à calibração automática e planejada da postura do robô, otimizando seu desempenho e aplicabilidade em tarefas que requerem precisão, o que amplia o alcance das técnicas apresentadas por [Toquica e Motta \(2024\)](#) para uma implementação prática em sistemas de controle de movimento.

3.3 Conceito em Projeção de Imagem

A projeção de imagem é uma técnica fundamental em visão computacional e gráficos computacionais, que envolve a transformação de um objeto tridimensional para um plano bidimensional. Este processo é crucial para a criação de representações visuais que podem ser exibidas em monitores ou analisadas por algoritmos de reconhecimento.

3.3.1 Modelo de Câmera Perspectiva

Todo sistema de aquisição de imagem, seja o sistema visual humano ou de máquina, por sua natureza realiza algum tipo de transformação do espaço 3D real ou simplesmente definido como espaço e o espaço 2D imaginário ou da imagem. Encontrar os parâmetros de tal transformação é fundamental para descrever o sistema de aquisição.

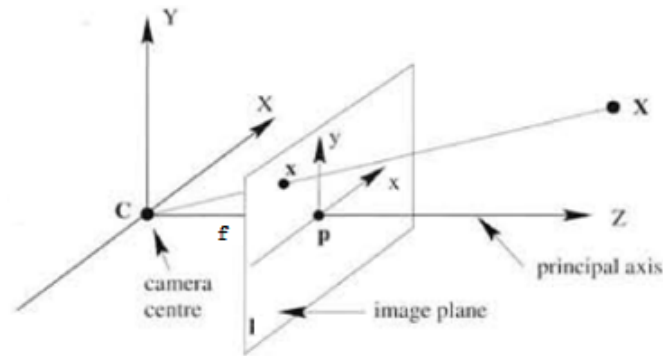


Figura 3.5 – Modelo de câmera perspectiva. Fonte: [Hartley e Zisserman \(2003\)](#).

No modelo mais simples de câmera, idealiza-se a luz como se estivesse sendo refletida da cena ou emitida por um objeto distante, mas apenas um único raio entra no orifício de qualquer ponto específico daquela cena. Em uma câmera física, esse ponto é então “projetado” em um plano de imagem [Kaehler e Bradski \(2016\)](#).

Esta seção descreve o método de aquisição de imagem conhecido como modelo de câmera perspectiva. Este modelo de câmera define um mapeamento geométrico do mundo 3D para o plano da imagem 2D, conhecido como projeção perspectiva. Esta seção está subdividida em duas partes. São descritos na [seção 3.3.1.1](#), os parâmetros intrínsecos da câmera, como distância focal e distorção das lentes e, na [seção 3.3.1.2](#), os parâmetros extrínsecos como a orientação e translação da câmera.

3.3.1.1 Parâmetros Intrínsecos

O modelo de câmera perspectiva é ilustrado na [figura 3.5](#) e detalhado a seguir. C representa o centro de projeção, ponto em que todos os raios se intersectam, também conhecido como centro da câmera ou centro óptico. Considerando que C está na origem, o plano I , conhecido como plano da imagem ou plano focal, é posicionado em frente ao centro de projeção e f é a distância focal. A linha que passa por C e é perpendicular ao plano da imagem é chamada de eixo principal ou eixo óptico e o ponto onde o eixo principal encontra o plano da imagem é chamado de ponto central (o ponto p ilustrado na [figura 3.5](#)).

Empregando o modelo de câmera perspectiva e considerando que o centro de projeção está localizado na origem do sistema de coordenadas 3D e o eixo óptico é colinear ao eixo Z , como mostra a [figura 3.5](#) [Hartley e Zisserman \(2003\)](#), um ponto no espaço com coordenadas $(X, Y, Z)^T$ é mapeado para um ponto no plano da imagem $(x, y)^T$ por meio da seguinte equação:

$$(x, y)^T = \left(f \frac{X}{Z}, f \frac{Y}{Z} \right)^T \quad (3.1)$$

Em uma câmera digital, o plano da imagem é uma grade de elementos sensíveis à luz como demonstrado na [figura 3.1](#) que correspondem diretamente aos elementos da imagem

(ou pixels). As coordenadas do pixel é um vetor 2×1 $(u, v)^T$ de valores não negativos inteiros e por convenção a origem está no canto superior esquerdo do plano da imagem. Os pixels são uniformes em tamanho e centralizados numa grade regular para que a coordenada do pixel esteja relacionada à coordenada do plano da imagem. Se os pontos no espaço 3D e os pontos no plano da imagem formada na câmera digital são representados por coordenadas homogêneas, a equação 3 pode ser escrita em notação matricial como:

$$\delta \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.2)$$

em que $\delta = Z$ é o fator de escala homogêneo.

Na equação (3.1) e na equação (3.2), considera-se que a origem das coordenadas no plano da imagem está posicionada no seu centro. No entanto, a maioria dos sistemas de imagem digitais define a origem como o pixel localizado na extremidade superior esquerda do plano. Essa discrepância exige um ajuste nas coordenadas, geralmente realizado por um processo de mapeamento que alinha o sistema de coordenadas da câmera com o referencial da imagem capturada. Esse procedimento é implementado utilizando coordenadas homogêneas, e sua formulação matemática é expressa por:

$$\delta \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & C_x & 0 \\ 0 & f & C_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.3)$$

em que $(C_x, C_y)^T$ são as coordenadas em pixel do ponto principal.

A equação (3.3) não leva em consideração distorções comuns em câmeras, que geralmente decorrem de imperfeições na fabricação das lentes. Para mitigar esses problemas, torna-se necessário incluir parâmetros de calibração que corrigem tais distorções. Em sistemas de visão robótica baseados em câmeras, como aqueles utilizados em algoritmos de controle visual, por exemplo, IBVS, a calibração da distância focal e do ponto principal da câmera desempenha um papel crucial. Esses parâmetros são essenciais para converter coordenadas de pixels em coordenadas reais e vice-versa, possibilitando medições precisas e um controle confiável do robô em relação ao seu alvo.

Os parâmetros intrínsecos de uma câmera são indispensáveis para interpretar adequadamente a perspectiva de uma imagem capturada, corrigir distorções e relacionar as dimensões do mundo real às dimensões expressas em pixels. Esses ajustes permitem que o sistema de visão robótica funcione com alta precisão em diversas aplicações.

A calibração da distância focal também é responsável por determinar a ampliação da imagem em relação à cena real. Esse parâmetro define o grau de ampliação dos objetos na imagem e é crucial para converter tamanhos expressos em pixels para distâncias reais. Uma calibração precisa da distância focal é indispensável em aplicações que exigem percepção de profundidade e posicionamento exato. Segundo [Zhang e Li \(2020\)](#), métodos modernos de calibração baseiam-se na captura de múltiplas imagens de padrões conhecidos, como tabuleiros de xadrez, posicionados em diferentes distâncias. Algoritmos de otimização refinam os valores calculados para minimizar os erros de projeção, conforme discutido em [Heikkilä \(2018\)](#) e [Ma e Zhou \(2021\)](#).

O ponto principal da câmera, geralmente identificado como o centro óptico, é o local onde o eixo óptico intercepta o plano de imagem. Idealmente, ele coincidiria com o centro da imagem, mas imperfeições no sensor ou na montagem óptica podem deslocá-lo. Esse deslocamento gera distorções que comprometem a precisão do sistema. A calibração do ponto principal corrige essas discrepâncias, possibilitando uma conversão precisa entre as coordenadas da imagem e o espaço real. Essa correção é especialmente útil em aplicações de visão computacional que exigem uma correspondência exata entre os sistemas de coordenadas da câmera e do ambiente, como destacado em [Zhao e Wan \(2019\)](#) e [Kim e Park \(2022\)](#).

Quando fatores corretivos são aplicados, como $f_x = f \cdot K_u$ e $f_y = f \cdot K_v$ para a calibração da distância focal, e $p_x = C_x \cdot A_u$ e $p_y = C_y \cdot A_v$ para a calibração do ponto principal, a [equação \(3.3\)](#) pode ser reescrita como:

$$\delta \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & p_x & 0 \\ 0 & f_y & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.4)$$

A relevância desses parâmetros melhora significativamente as aplicações em manipulação robótica, onde pequenas imprecisões podem acarretar erros expressivos na posição e orientação do robô. Para mitigar esses efeitos, métodos avançados de calibração têm sido amplamente adotados, incluindo técnicas de otimização baseadas em aprendizado de máquina e algoritmos de ajuste iterativo. Essas abordagens não apenas minimizam erros residuais, mas também aprimoram a precisão nas medições de distância e posição, como evidenciado por [Wang e Li \(2023\)](#). Esses métodos são essenciais para estabelecer a correspondência entre pontos tridimensionais no espaço real e suas projeções bidimensionais no plano da imagem.

Na [equação \(3.4\)](#), assume-se que as coordenadas da imagem possuem a mesma escala nos eixos u e v , caracterizando pixels com formato quadrado (razão de aspecto 1:1). No entanto, é possível que os pixels não sejam perfeitamente quadrados [Hartley e Zisserman \(2003\)](#), resultando em diferenças na escala entre os eixos. Essas variações tornam necessária

a consideração de fatores de resolução distintos, η_x e η_y , que representam a largura e a altura do pixel em relação aos eixos u e v , respectivamente. Incorporando esses fatores, a equação (3.4) pode ser reescrita para incluir essas correções, garantindo maior precisão na modelagem da projeção:

$$\delta \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f_x}{\eta_x} & 0 & p_x & 0 \\ 0 & \frac{f_y}{\eta_y} & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.5)$$

Outro fator a ser considerado é o de inclinação da imagem, embora que na maioria das câmeras esse parâmetro seja zero (onde não existe inclinação). A figura 3.6 (b) ilustra uma imagem com inclinação. Levando em consideração o parâmetro de inclinação τ , a equação (3.5) pode ser escrita como:

$$\delta \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f_x}{\eta_x} & \tau & p_x & 0 \\ 0 & \frac{f_y}{\eta_y} & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \left[K \mid \mathbf{0}_3 \right] Q \quad (3.6)$$

em que $\mathbf{0}_3$ é um vetor nulo, $Q = (X, Y, Z, 1)^T$ é um ponto no espaço 3D e K representa os parâmetros intrínsecos da câmera, também conhecidos como matriz de calibração da câmera:

$$K = \begin{bmatrix} \frac{f_x}{\eta_x} & \tau & p_x \\ 0 & \frac{f_y}{\eta_y} & p_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3.7)$$

Quando a razão de aspecto é 1 : 1, $\eta_x = 1$ e $\eta_y = 1$. Se a imagem não é inclinada, então $\tau = 0$.

Até aqui, os parâmetros intrínsecos empregados no processo de formação de imagem pelo modelo de câmera perspectiva foi descrito. Entretanto, as lentes das câmeras reais também podem sofrer com algum tipo de distorção radial. Para [Hartley e Zisserman \(2003\)](#), a modelagem exata das lentes é uma tarefa complexa, sendo a distorção radial a mais importante a ser considerada. A distorção radial causa o mapeamento de uma reta, para uma linha com uma determinada curvatura, como mostra a figura 3.7. Para [Morvan \(2009\)](#), a relação entre a posição dos pixels com distorção radial corrigida $(x_u, y_u)^T$ e com distorção radial $(x_d, y_d)^T$ é definida por:

$$\begin{bmatrix} x_u \\ y_u \end{bmatrix} = L(r_d) \begin{bmatrix} x_d - p_x \\ y_d - p_y \end{bmatrix} + \begin{bmatrix} p_x \\ p_y \end{bmatrix} \quad (3.8)$$

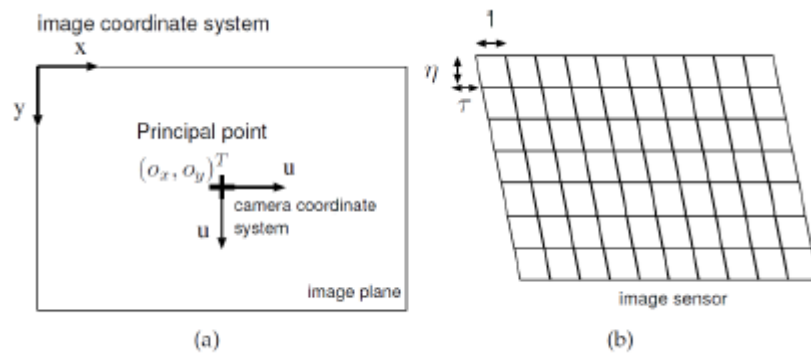


Figura 3.6 – (a) plano da imagem com o sistema de coordenadas da câmera $(u, v)^T$ e a imagem $(x, y)^T$; (b) imagem inclinada e as coordenadas da imagem com diferentes escalas na direção dos eixos x e y . Fonte: Morvan (2009).

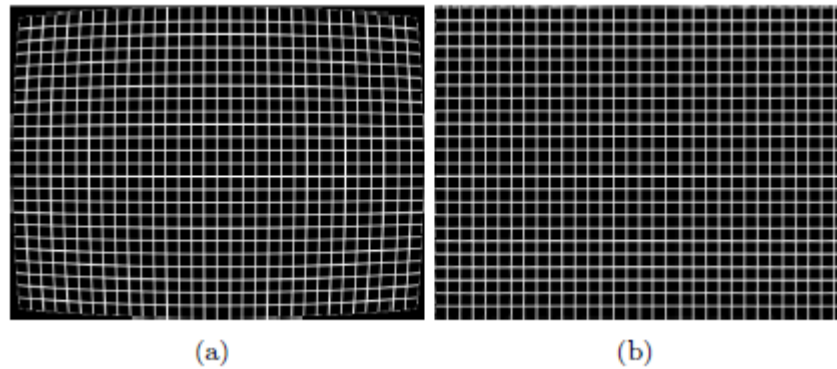


Figura 3.7 – (a) imagem com distorção radial; (b) imagem com distorção radial corrigida. Fonte: Fry e Pusateri (2010).

em que $(p_x, p_y)^T$ são as coordenadas do ponto principal e $L(r_d) = 1 + k_1 r_d^2$, em que k_1 é a quantidade de distorção radial presente na imagem e $r_d^2 = (x_d - p_x)^2 + (y_d - p_y)^2$.

A correção da distorção radial na imagem requer a estimação dos parâmetros k_1 e $(p_x, p_y)^T$. Estes parâmetros podem ser estimados pelo cálculo da curvatura de uma linha na imagem 2D, que é uma linha reta no espaço 3D. Utilizando um padrão de calibração, as 3 linhas destacadas na figura 3.7, no mundo real, são linhas retas, entretanto, a imagem capturada apresenta as mesmas linhas com uma determinada curvatura. Mais detalhes de como estimar os parâmetros para a correção da distorção radial são descritos em Fry e Pusateri (2010) e Kim *et al.* (2010).

3.3.1.2 Parâmetros Extrínsecos

Os parâmetros extrínsecos relacionam o sistema de coordenadas da câmera com o sistema de coordenadas global, descrevendo a posição e orientação da câmera no mundo 3D, como mostra a figura 3.9. A equação (3.6) considera que a câmera está na origem do sistema de coordenadas global e o plano $Z = f$ é o plano da câmera. Para Hartley e Zisserman (2003) a posição e orientação da câmera no sistema de coordenadas no mundo podem ser

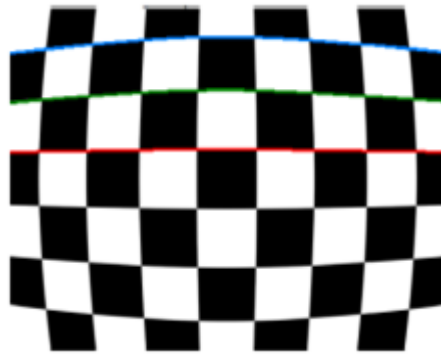


Figura 3.8 – Estimação da distorção radial utilizando um padrão de calibração. Fonte: Kim *et al.* (2010).

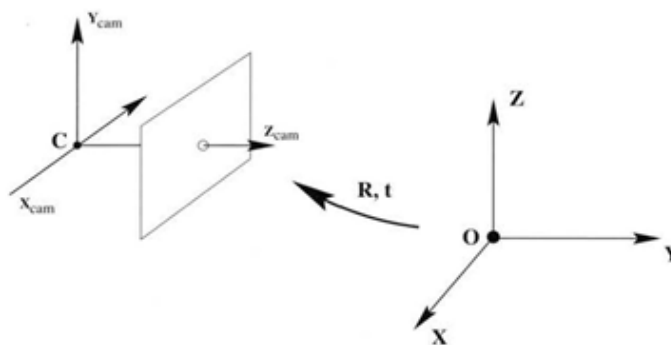


Figura 3.9 – Transformação Euclidiana entre o sistema de coordenadas da câmera e do mundo. Fonte: Hartley e Zisserman (2003).

escritas como:

$$\mathbf{p} = \begin{bmatrix} \mathbf{K} & | & \mathbf{0}_3 \\ \mathbf{0}_3^T & | & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R} & -\mathbf{R}\tilde{\mathbf{C}} \\ \mathbf{0}_3^T & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{KR} \begin{bmatrix} \mathbf{I} & | & -\tilde{\mathbf{C}} \end{bmatrix} \mathbf{P} \quad (3.9)$$

m que a matriz \mathbf{K} define os parâmetros intrínsecos, \mathbf{R} e $\tilde{\mathbf{C}}$ representam os parâmetros extrínsecos da câmeras, em que \mathbf{R} é uma matriz de rotação 3x3 e $\tilde{\mathbf{C}}$ define o centro de projeção da câmera no espaço em coordenadas não-homogêneas, \mathbf{I} é uma matriz identidade 3x3 e $\mathbf{p} = (x; y; 1)^T$ e $\mathbf{P} = (X; Y; Z; 1)^T$ representam o mesmo ponto nos sistemas de coordenadas da câmera e no espaço, respectivamente.

A equação (3.9) define o mapeamento completo da câmera estenopeica, incluindo os parâmetros intrínsecos e extrínsecos. A matriz \mathbf{Q} mostrada na equação (3.10), de dimensão 3x4, é conhecida como a *matriz de projeção da câmera*, em que um ponto \mathbf{P} do espaço 3D é mapeado para um ponto \mathbf{p} no plano da imagem 2D por $\mathbf{p} = \mathbf{QP}$.

$$\mathbf{Q} = \mathbf{KR} \begin{bmatrix} \mathbf{I} & | & -\tilde{\mathbf{C}} \end{bmatrix} \quad (3.10)$$

3.4 Conceitos de Processamento de Imagem

Para continuidade dessa dissertação é necessária revisão de alguns conceitos de processamentos de imagem. Essa seção é dedicada à apresentação desses conceitos. Todos eles serão executados em forma de comandos de linguagem MATLAB¹.

3.4.1 Alteração de Dados da Imagem Digitalizada

A imagem capturada pela câmera pode ser facilmente manipulada após ser digitalizada. Essas alterações são fundamentais para executar os processamentos desejados na imagem. A imagem analógica capturada pela câmera é convertida em sinal digital sendo alocado em pixel's. Cada pixel guarda um valor correspondente da intensidade apresentada na imagem analógica. Para imagens digitalizadas em cores os pixel's guardam esses valores em três canais RGB (canais com intensidade na cor vermelha, verde e azul). A câmera utilizada nesse trabalho (Basler acA2500) é monocromática apresentando imagens em escala de cinza sendo necessário somente um canal para guardar dados de intensidade.

Para alterar os dados digitalizados um algoritmo deve percorrer por cada pixel e executar as modificações necessárias. Geralmente são usadas convoluções numa certa área de pixel escolhida. No entanto, podem ser usadas mudanças de pixel a pixel como a normalização utilizada nesse trabalho para tratar os dados de forma mais simplória. A normalização em questão transforma a escala de cinza por um intervalo numérico de 0 a 1. Isso é possível multiplicando cada registro armazenado nos pixel's por uma variável conveniente. O comando utilizado no MATLAB que executa essa normalização é o *im2gray*.

Outra questão representativa é a alteração do tamanho da imagem. O processamento da imagem com resolução diminuída pode diminuir sensivelmente o processamento computacional tornando-o mais rápido e menos oneroso. A qualidade da imagem, no entanto, também é diminuída. Aqui, nessa dissertação, foi feito um balanceamento para esse quesito entre processamento e qualidade da imagem. Uma forma de executar essa diminuição é percorrer pela imagem guardando dados de pixel's intervalados de tal forma que a imagem fique com a resolução desejada. Esses intervalos devem ser idênticos e nas devidas proporções para manter a relação de aspecto. O comando em MATLAB é o *imresize*.

3.4.2 Filtragem Gaussiana

A filtragem da imagem pode ser utilizada para realizar desfoque suave, aumentar a nitidez dos detalhes, acentuar bordas ou remover ruídos [Szeliski \(2010\)](#).

Um filtro gaussiano é utilizado para borrar ou desfocar a imagem na qual ele é aplicado com o objetivo de reduzir os ruídos presentes na imagem. O resultado desta operação é a

¹ MATLAB (MATrix LABoratory) trata-se de um software interativo de alta performance voltado para o cálculo numérico

suavização da imagem, lembrando a visualização da mesma através de uma tela translúcida ou como se tivesse sendo vista através de uma lente fora de foco. A suavização gaussiana é largamente utilizada no estágio de pré-processamento da imagem a fim de realçar a estrutura da imagem em diferentes escalas [Jesus e Jr \(2015\)](#).

Matematicamente, a aplicação do filtro gaussiano é realizada da mesma forma que a convolução da imagem com uma função gaussiana, como um filtro passa baixa, ou como um filtro de média simples.

Esta função gaussiana expressa a distribuição normal em estatística. Como uma imagem é definida em duas dimensões, a função gaussiana utilizada no filtro gaussiano aplicado a ela, também deve ser definido em duas dimensões, ou seja, uma dimensão na direção horizontal x e outra dimensão em vertical y [equação \(3.11\)](#).

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3.11)$$

sendo σ o desvio padrão.

Em MATLAB o comando utilizado para essa filtragem gaussiana é *imgaussfilt*

3.4.3 Detecção de Bordas

As bordas constituem informação de alta frequência e encerram propriedades significativas de uma imagem. Estas propriedades incluem descontinuidades fotométricas, geométricas e as características físicas dos objetos. Tais propriedades do objeto são passadas à imagem pois, variações pertinentes ao objeto ocasionam variações nos tons de cinza da imagem [Maia e Porfírio \(2002\)](#).

Para que se obtenham resultados precisos nos passos subsequentes à detecção das bordas, é necessário que esta detecção seja eficiente e confiável. A fim de que as variações dos tons de cinza sejam detectadas (bordas) é necessário diferenciar a intensidade da imagem. Quando a imagem é diferenciada, as variações dos níveis de cinza são detectadas e, por consequência, detecta-se também ruído, que é uma forma indesejável de variação. Para que as bordas espúrias não sejam então detectadas, deve-se suavizar a imagem antes da detecção. Contudo, existem efeitos inoportunos ligados à suavização, perda de informação e deslocamento de estruturas de feições proeminentes no plano da imagem. Além disso, existem diferenças entre as propriedades dos operadores diferenciais comumente utilizados, o que ocasiona bordas diferentes. Logo, é difícil formular um algoritmo de detecção de bordas que possua um bom desempenho em diferenciados contextos e capture os requisitos necessários aos estágios subsequentes de processamento [Ziou, Tabbone et al. \(1998\)](#). Consequentemente, no tocante ao processamento de imagem digital, uma variedade de detetores de bordas tem sido desenvolvidos visando diferentes propósitos, com formulações matemáticas diferenciadas e com propriedades algorítmicas distintas.

Com base nos problemas acima mencionados, [Canny \(1986\)](#), desenvolveu um processo de detecção de bordas a partir de critérios de quantificação de desempenho de operadores de bordas conhecidos como: critério de detecção e critério de localização. Estes critérios de desempenho ainda estão sujeitos ao critério de resposta múltipla, que corresponde ao fato de que deve haver, na saída do operador, uma única resposta para uma única borda. Para que os critérios sejam aproximadamente atendidos, [Canny \(1986\)](#) aproxima o operador ótimo, obtido a partir dos três critérios de desempenho, pela primeira derivada da função Gaussiana. Em complemento a este operador, foi proposto um processo chamado supressão não máxima (supressão de valores de pixels que não forem máximos locais na direção transversal à borda), que causaria um afinamento da borda, atendendo à injunção de resposta múltipla, e uma limiarização adaptativa (histerese) com “complementação de bordas”, para eliminar a fragmentação dos contornos das bordas.

Dada sua importância, os detetores de borda (step edge detectors) fazem parte de muitos sistemas de visão computacional. Suas aplicações são incontáveis e uma de suas principais características é a de reduzir drasticamente a quantidade de dados a serem processados, preservando informações estruturais importantes sobre a fronteira dos objetos. Qualquer que seja o detetor e a técnica envolvida, é necessário que este atenda à três critérios básicos [Canny \(1986\)](#), a saber:

a) Taxa de Erro: (*Signal to noise ratio* - SNR)

- As bordas que ocorrem na imagem não devem ser confundidas e não se deve detectar bordas falsas, ou seja, o detetor deveria ter baixa probabilidade de:
 - Falhar ao detectar pontos de borda verdadeiros;
 - Detectar, falsamente, pontos não pertencentes à borda.
- Visto que ambas as probabilidades são funções monotonicamente decrescentes, em função da razão sinal/ruído, este critério equivale a maximizar a razão sinal/ruído;
- Maximizar a razão sinal/ruído implica em minimizar o ruído.

b) Localização:

- Pontos de borda devem estar bem localizados, isto é a distância entre os pontos extraídos pelo detetor e o "centro" verdadeiro da borda deve ser minimizada.

c) Resposta:

- O detetor de borda não deve identificar múltiplos pixels de borda onde existe um único, ou seja, deve-se obter uma única resposta para uma única borda. Este critério está implicitamente contido no 1º critério pois quando ocorrem duas respostas para uma mesma borda, então, uma deverá ser considerada falsa. Contudo, a formulação matemática do 1º critério não engloba necessariamente a questão de resposta múltipla, sendo necessário explicitá-la

Teoricamente estabelecidas as metas de desempenho de um detetor de bordas, o que se quer agora é sintetizá-las matematicamente de modo a se tornarem uma ferramenta aplicável. Sem perda de generalidade, os sinais serão trabalhados de forma unidimensional, pois o comportamento das bordas é análogo para o caso bidimensional, exceto no caso da ocorrência de cantos.

Uma forma matemática de definir a inclinação e a direção de uma superfície é através do seu gradiente (equação (3.12)).

$$\mathbf{E}(x) = \nabla \mathbf{I}(x) = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right) (x) \quad (3.12)$$

O vetor gradiente local \mathbf{E} aponta na direção da subida mais íngreme na função de intensidade. A sua magnitude é uma indicação da inclinação ou força da variação, enquanto a sua orientação aponta em uma direção perpendicular ao contorno local.

Como a diferenciação é uma operação linear, ela comuta com outras operações de filtragem linear. O gradiente da imagem suavizada pode, portanto, ser escrito como:

$$\mathbf{E}_\sigma(x) = \nabla [\mathbf{G}_\sigma * \mathbf{I}(x)] = [\nabla \mathbf{G}_\sigma] (x) * \mathbf{I}(x) \quad (3.13)$$

ou seja, podemos convolver a imagem com as derivadas horizontais e verticais da função Gaussiano Kernel (equação (3.14)).

$$\nabla \mathbf{G}_\sigma(x) = \left(\frac{\partial G_\sigma}{\partial x}, \frac{\partial G_\sigma}{\partial y} \right) (x) = [-x \ -y] \frac{1}{\sigma^3} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3.14)$$

Existem inúmeras literaturas com algoritmos detectores de bordas em processamento de imagem. O comando utilizado utilizado no MATLAB será *edge*.

3.4.4 Detecção de Linhas com Transformada de Hough

A transformada de Hough é um algoritmo criado para detecção de formas que são facilmente parametrizadas (linhas, círculos, elipses, etc.) em imagens computacionais Krishna (2017). Em geral, a transformada é aplicada após a imagem sofrer um pré-processamento, comumente a detecção de bordas como o apresentado na seção 3.4.3.

O conceito principal da transformada de Hough consiste em definir um mapeamento entre o espaço de imagem e o espaço de parâmetros. Cada borda de uma imagem é transformada pelo mapeamento para determinar células de pixels no espaço de parâmetros, indicadas pelas primitivas definidas através do ponto analisado. Essas células são incrementadas, e indicarão no final do processo, através da máxima local do acumulador, quais os parâmetros correspondentes a forma especificada procurada no algoritmo.

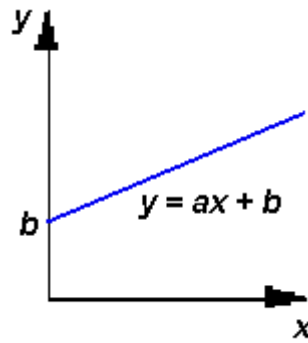


Figura 3.10 – representação da equação da reta (*declive-intercepte*).

A transformada de Hough consiste em mapear um pixel da imagem em uma curva no espaço de parâmetros, organizado em forma de um acumulador n dimensional, onde n corresponde ao número de parâmetros.

Há várias parametrizações possíveis para o espaço de linhas. Hough usou a equação declive-intercepte, definida por $y = a.x + b$, como representação paramétrica de uma linha (figura 3.10), o que conduziu dificuldade prática de um espaço de parâmetro ilimitado para linha que são paralelas ao eixo y , cuja solução será abordada posteriormente.

O algoritmo de Hough requer um acumulador de dimensão igual ao número de parâmetros desconhecidos na equação da família de curvas que são buscadas. Por exemplo, achar segmentos de linhas usando a equação $y = ax + b$ requer achar dois parâmetros para cada segmento: a e b . As duas dimensões da matriz acumuladora para esta família correspondem aos valores ‘quantizados’ para a e b .

Assim, usando uma matriz acumuladora \mathbf{A} , o procedimento de Hough examina cada pixel e calcula os parâmetros da curva (equação) especificada que passa pelo pixel. Caso esteja analisando uma imagem que não foi pré-processada com algoritmo de detecção de bordas, fato incomum na transformada de Hough, será examinado o pixel e sua vizinhança na imagem, para determinar se há evidência de extremidade naquele pixel. Somente se houver será realizado o cálculo dos parâmetros.

Após calculados os parâmetros de um determinado pixel, eles são ‘quantizados’ para um valor correspondente a e b , e o acumulador $\mathbf{A}(a,b)$ é incrementado.

Quando todos pixels tiverem sido processados, é procurado no acumulador \mathbf{A} os maiores valores (picos). Eles indicam os parâmetros de prováveis linhas na imagem.

Abaixo é demonstrado um pequeno exemplo utilizando a técnica supracitada, cujo cálculo é realizado apenas para dois pixels pertencentes a uma linha (reta) de uma imagem, apresentada na figura 3.11a, (plano $x - y$ ou espaço real). Na figura 3.11b (plano $a - b$ ou espaço de Hough), pode-se verificar as linhas geradas pelos parâmetros calculados, onde a linha verde refere-se ao ponto (1,1) e a azul ao ponto (2,2) do plano $x - y$.

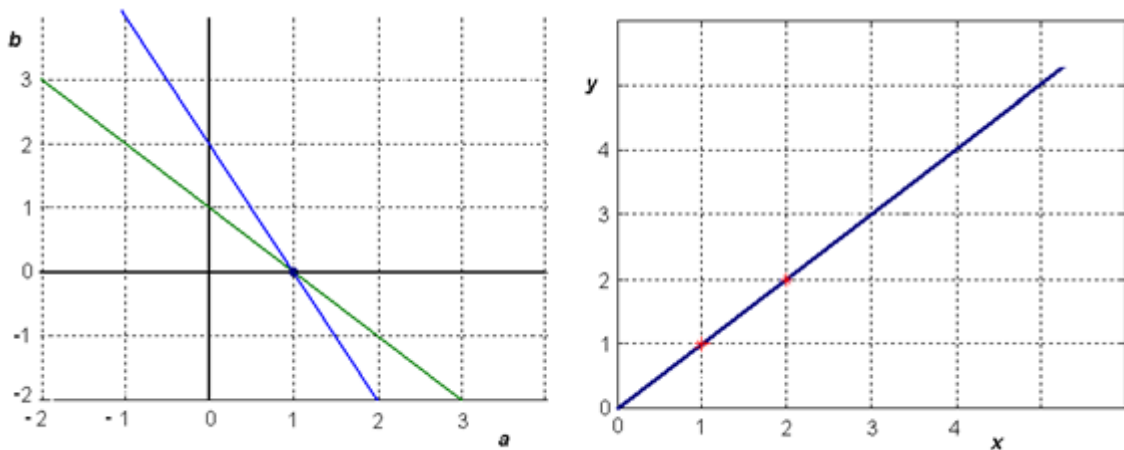


Figura 3.11 – (a) Plano $x - y$ (imagem) e (b) Plano $a - b$ (parâmetros).

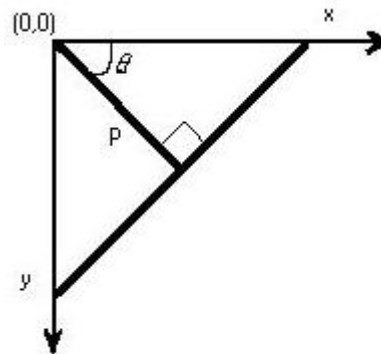


Figura 3.12 – representação da equação da reta (coordenada polar).

Um limiar pode ser utilizado quando se procura o(s) máximo(s) no acumulador, a fim de determinar um valor mínimo de pontos colineares. Se o valor do acumulador não for superior ao do limiar então será considerado um ruído.

As detecções de outras formas utilizando a transformada de Hough usam o mesmo princípio, há somente alteração no número de parâmetros da equação que será empregada, e em consequência na dimensão do acumulador.

Duda e Hart (1972) utilizaram coordenada polar para representação de uma linha. Sugeriram que linhas poderiam ser completamente parametrizadas usando o comprimento, ρ , e a orientação, θ , do vetor normal para a linha da imagem original (figura 3.12). Usando esta parametrização, todo o ponto (x,y) na linha satisfará a equação $\rho = x \cdot \cos(\theta) + y \cdot \sin(\theta)$.

Computa-se a transformada de Hough dos pixels da imagem, registrando os resultados em um histograma (matriz) bidimensional. Cada pixel (x,y) do espaço real produz uma linha senoidal no espaço de Hough. A linha no espaço de Hough parecerá como mostrado na figura 3.13, para representação de Duda e Hart (1972).

O histograma torna-se complicado quando a imagem contiver muitas linhas, com

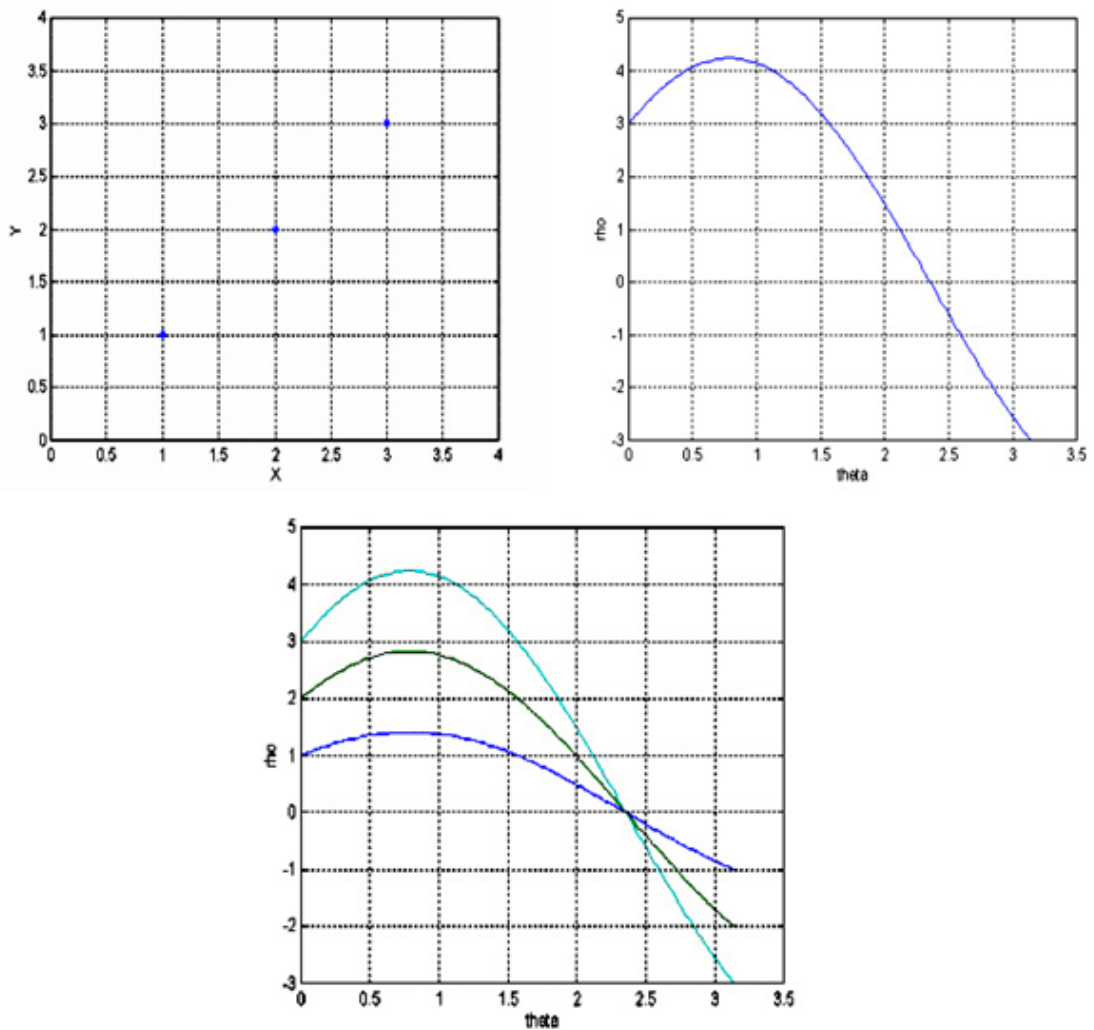


Figura 3.13 – Transformação do espaço $x - y$ original para o espaço parametrizado $\theta - \rho$.

máximos múltiplos, dos quais muitos não correspondem à linhas, mas a ruídos. Assim deve-se interativamente descobrir a maior linha da imagem, remover sua contribuição do histograma, e então repetir o processo para achar sucessivamente a maior das linhas restantes. A cada repetição acha-se o máximo global e obtém-se a equação da linha correspondente. Um simples pós-processamento pode ser usado para achar os pontos de começo e fim de cada linha. O comando em MATLAB correspondente é *hough*.

3.4.5 Detecção de Círculos e de Elipses

A detecção de objetos circulares em imagens digitais é um problema importante e recorrente em processamento de imagens e visão computacional, e tem muitas aplicações. Um algoritmo de detecção de círculo ideal deve ser executado com um conjunto fixo de parâmetros internos para todas as imagens, ou seja, não requer ajuste de parâmetros para imagens diferentes, ser muito rápido (em tempo real, se possível), detectar vários círculos pequenos e grandes, trabalhar com sintéticos, imagens naturais e ruidosas, têm alta taxa

de detecção e boa precisão e produzem poucas ou nenhuma detecção falsa [Akinlar e Topal \(2013\)](#).

Tradicionalmente, as técnicas de detecção de círculo mais populares são baseadas na famosa transformada de Hough ([seção 3.4.4](#)) para detecção de círculos. Essas técnicas primeiro calculam um mapa de arestas da imagem usando um detector de bordas tradicional como Canny, [seção 3.4.3](#), mapeiam os pixels das bordas no espaço circular de Hough tridimensional (x, y, r) e extraem círculos que contêm um certo número de bordas. As técnicas baseadas em transformada de Hough para detecção de círculos são muito lentas e exigem muita memória, mas também produzem muitas detecções falsas, especialmente na presença de ruído. Além disso, esses métodos possuem muitos parâmetros que devem ser predefinidos pelo usuário, o que limita bastante seu uso.

Para superar as limitações dos métodos clássicos, muitas variantes foram propostas para detecção de círculos em imagens com transformada de Hough, incluindo a probabilística, a randomizada, a difusa, etc. Todos esses métodos tentam corrigir diferentes deficiências nesse método, mas ainda exigem muita memória e são lentos para serem úteis em aplicações em tempo real [Akinlar e Topal \(2013\)](#).

O MATLAB contém um comando (*imfindcircles*) para detecção de círculos que satisfaz nossa proposta. Porém existem inúmeras literaturas que abrangem a detecção de círculos em imagem.

Outra detecção na imagem de interesse nesse trabalho são as elipses. A busca dos alvos que o manipulador robótico deve buscar são circulares, porém devido a inclinação da câmera em relação a imagem, esse alvo circular pode se apresentar como uma elipse. O comando correspondente em MATLAB para essa função é o *regionprops*.

3.5 Controle Servo Visual Baseado em Imagem - IVBS

No IBVS, são necessários conjuntos de características de imagem que correspondam às coordenadas específicas de um ponto desejado da imagem, \mathbf{p}^* , e ao ponto capturado na imagem após cada deslocamento, \mathbf{p} . A partir da comparação entre esses pontos, é gerado um sinal de erro a ser enviado ao manipulador robótico. No entanto, esse sinal é gerado no espaço da imagem e requer uma transformação para o espaço da tarefa, o que é feito através de uma matriz de interação de pontos ou o jacobiano da imagem, conforme proposto por [Chaumette \(2007\)](#).

O jacobiano da imagem \mathbf{J}_s se relaciona com \mathbf{p} desempenhando um papel crucial na lei de controle, sendo descrito por:

$$\dot{\mathbf{p}} = \begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \left(\frac{\partial \mathbf{p}}{\partial \mathbf{P}} \right) \frac{d\mathbf{P}}{dt} = \mathbf{J}_s \mathbf{v}_c \quad (3.15)$$

onde

$$\mathbf{J}_s = \left(\frac{\partial \mathbf{p}}{\partial \mathbf{P}} \right) \quad (3.16)$$

O jacobiano da imagem é composto pelas derivadas parciais da função de intensidade da imagem em relação às suas coordenadas espaciais [Chaumette \(2007\)](#). Ele é essencial para descrever como pequenas variações na posição dos pixels afetam a intensidade da imagem, sendo fundamental para a análise de movimentos, alinhamento de imagens, rastreamento de objetos e outras aplicações. Uma aplicação crucial que utiliza o jacobiano da imagem é a estimativa de movimento, ou fluxo óptico. O fluxo óptico descreve o padrão de movimento dos objetos, superfícies e bordas na cena visual.

A velocidade da câmera \mathbf{v}_c é:

$$\mathbf{v}_c = \frac{d\mathbf{P}}{dt} = [\mathbf{v}_t, \mathbf{v}_\Omega]^T \quad (3.17)$$

com componente de translação

$$\mathbf{v}_t(t) = [v_x(t), v_y(t), v_z(t)]^T \quad (3.18)$$

e de rotação

$$\mathbf{v}_\Omega(t) = [\omega_x(t), \omega_y(t), \omega_z(t)]^T \quad (3.19)$$

3.5.1 Matriz Jacobiano da Imagem

O jacobiano da imagem \mathbf{J}_s relaciona o movimento da câmera em relação a um referencial global com as variações das características no referencial da imagem. O sistema de coordenadas da imagem (u, v) de um ponto \mathbf{p} se relaciona com o sistema de coordenadas espaciais (X, Y, Z) do ponto \mathbf{P} real. A derivada do ponto \mathbf{P} pode ser expressa como:

$$\dot{\mathbf{P}} = -\mathbf{v}_\Omega \times \mathbf{P} - \mathbf{v}_t \quad (3.20)$$

Essa equação descreve a dinâmica do ponto \mathbf{P} em um sistema onde ele está sujeito a duas contribuições de movimento. O primeiro termo representa a influência da velocidade angular \mathbf{v}_Ω sobre a posição \mathbf{P} por meio de um produto vetorial. A posição \mathbf{P} está se movendo em um caminho circular devido à rotação em torno de um eixo, com a direção do movimento sendo perpendicular ao plano definido por \mathbf{v}_Ω e \mathbf{P} . Já o segundo termo representa uma velocidade adicional tangencial ao movimento \mathbf{v}_t , que pode ser uma velocidade de controle ou outra forma de movimento que também afeta a posição do ponto \mathbf{P} . A equação (3.20),

portanto, descreve a taxa de variação no tempo do vetor posição \mathbf{P} em resposta à rotação e a uma velocidade de controle.

Podemos escrever a equação (3.20) na forma escalar como:

$$\dot{X} = -v_x - \omega_y Z + \omega_z Y \quad (3.21)$$

$$\dot{Y} = -v_y - \omega_z X + \omega_x Z \quad (3.22)$$

$$\dot{Z} = -v_z - \omega_x Y + \omega_y X \quad (3.23)$$

A projeção em perspectiva equação (3.5) para coordenadas calibradas é:

$$u = \frac{f_x X}{\eta_x Z} + p_x, v = \frac{f_y Y}{\eta_y Z} + p_y \quad (3.24)$$

O controle servo-visual baseado em imagem (IBVS) utiliza essa projeção de pontos do espaço tridimensional para a imagem bidimensional como base para o controle do robô. Essas equações relacionam diretamente as coordenadas tridimensionais do ponto no espaço real (X, Y, Z) com as coordenadas bidimensionais projetadas na imagem (u, v) .

A equação (3.24) pode ser reorganizada como:

$$u - p_x = \frac{f_x X}{\eta_x Z}, v - p_y = \frac{f_y Y}{\eta_y Z} \quad (3.25)$$

substituindo os mapeamentos relacionados ao ponto central da imagem e a sua origem como $\bar{u} = u - p_x$ e $\bar{v} = v - p_y$:

$$\bar{u} = \frac{f_x X}{\eta_x Z}, \bar{v} = \frac{f_y Y}{\eta_y Z} \quad (3.26)$$

A derivada temporal da equação (3.24), usando a regra do quociente, é:

$$\dot{u} = \frac{f_x}{\eta_x} \left(\frac{\dot{X}Z - X\dot{Z}}{Z^2} \right), \dot{v} = \frac{f_y}{\eta_y} \left(\frac{\dot{Y}Z - Y\dot{Z}}{Z^2} \right) \quad (3.27)$$

Escrevendo a equação (3.26) como:

$$X = \bar{u} \frac{\eta_x}{f_x} Z, Y = \bar{v} \frac{\eta_y}{f_y} Z \quad (3.28)$$

substituindo essa equação (3.28) e as equações na forma escalar (3.21, 3.21, 3.23) da derivada do ponto \mathbf{P} na equação (3.27) obtemos a seguinte forma matricial:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} -\frac{f_x}{\eta_x Z} & 0 & \frac{\bar{u}}{Z} & \frac{\eta_y \bar{u} \bar{v}}{f_y} & -\left(\frac{f_x}{\eta_x} + \frac{\eta_x \bar{u}^2}{f_x}\right) & \frac{f_x \eta_y}{\eta_x f_y} \bar{v} \\ 0 & -\frac{f_y}{\eta_y Z} & \frac{\bar{v}}{Z} & \left(\frac{f_y}{\eta_y} + \frac{\eta_y \bar{v}^2}{f_y}\right) & -\frac{\eta_x \bar{u} \bar{v}}{f_x} & -\frac{f_y \eta_x}{\eta_y f_x} \bar{u} \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (3.29)$$

Portanto, pode-se inferir da equação (3.29) que \mathbf{J}_s é:

$$\mathbf{J}_s = \left(\frac{\partial \mathbf{p}}{\partial \mathbf{P}} \right) = \begin{bmatrix} -\frac{f_x}{\eta_x Z} & 0 & \frac{\bar{u}}{Z} & \frac{\eta_y \bar{u} \bar{v}}{f_y} & -\left(\frac{f_x}{\eta_x} + \frac{\eta_x \bar{u}^2}{f_x}\right) & \frac{f_x \eta_y}{\eta_x f_y} \bar{v} \\ 0 & -\frac{f_y}{\eta_y Z} & \frac{\bar{v}}{Z} & \left(\frac{f_y}{\eta_y} + \frac{\eta_y \bar{v}^2}{f_y}\right) & -\frac{\eta_x \bar{u} \bar{v}}{f_x} & -\frac{f_y \eta_x}{\eta_y f_x} \bar{u} \end{bmatrix} \quad (3.30)$$

Note que a matriz Jacobiano da imagem é função das coordenadas u e v e também da profundidade Z da imagem. Outra questão importante sobre a matriz \mathbf{J}_s é que sua dimensão é 2×6 para cada ponto $\mathbf{p} (u, v)^T$ da imagem.

Caso sejam consideradas correspondências entre pontos maiores ou iguais a três a matriz \mathbf{J}_s pode ser inversível e a equação (3.30) se torna:

$$\begin{bmatrix} \dot{p}_1 \\ \dot{p}_2 \\ \vdots \\ \dot{p}_n \end{bmatrix} = \begin{bmatrix} J_{s1} \\ J_{s2} \\ \vdots \\ J_{sn} \end{bmatrix} \mathbf{v}_c \quad (3.31)$$

O interesse desse trabalho é conseguir as velocidades necessárias para câmera a partir da velocidade da imagem. Isso é possível invertendo a matriz apresentada na equação (3.31):

$$\mathbf{v}_c = \begin{bmatrix} J_{s1} \\ J_{s2} \\ \vdots \\ J_{sn} \end{bmatrix}^{-1} \begin{bmatrix} \dot{p}_1 \\ \dot{p}_2 \\ \vdots \\ \dot{p}_n \end{bmatrix} \quad (3.32)$$

Para calcular o movimento necessário da câmera com base na velocidade dos pontos na imagem, utiliza-se um método linear simples de controle servo visual. Este método é descrito pela equação abaixo, proposta por Corke (2017):

$$\dot{\mathbf{p}} = \lambda (\mathbf{p}^* - \mathbf{p}), \quad (3.33)$$

onde $\dot{\mathbf{p}}$ representa a velocidade desse ponto no plano da imagem, \mathbf{p}^* é a localização desejada do ponto característico, \mathbf{p} é a localização atual desse ponto, e λ é um ganho positivo de controle que ajusta a taxa de convergência.

Essa abordagem direciona os pontos característicos no plano da imagem para as posições desejadas, gerando uma trajetória suave para a câmera. A ideia principal é que, ao final do movimento, os pontos característicos \mathbf{p} no plano da imagem coincidam com os pontos desejados \mathbf{p}^* , garantindo precisão no alinhamento visual. Esse método simples e eficaz é amplamente utilizado em controle servo visual baseado em imagem devido à sua robustez e à facilidade de implementação. Por fim a velocidade da câmera pode ser reescrita como:

$$\mathbf{v}_c = \lambda \begin{bmatrix} J_{s_1} \\ J_{s_2} \\ \vdots \\ J_{s_n} \end{bmatrix}^{-1} \begin{bmatrix} p_1^* - p_1 \\ p_2^* - p_2 \\ \vdots \\ p_n^* - p_n \end{bmatrix} \quad (3.34)$$

O controle IBVS difere do PBVS na forma como estima a posição do alvo. Enquanto o PBVS localiza o alvo visualmente e calcula sua posição e orientação, o IBVS realiza essa tarefa através da análise das diferenças entre imagens, buscando reduzir esse erro até alcançar a mínima possível.

Para implementar a metodologia PBVS, é essencial determinar tanto os parâmetros intrínsecos quanto os extrínsecos da câmera, conforme descrito na [equação \(3.9\)](#) da [seção 3.3.1.2](#). Esses parâmetros permitem estimar a posição e orientação de um alvo ou objeto no mundo real. No entanto, essa abordagem exige um poder de processamento considerável para resolver equações complexas, frequentemente por meio de métodos computacionais avançados. Adicionalmente, problemas relacionados à calibração inadequada podem introduzir erros significativos, afetando diretamente a precisão das estimativas e a robustez do sistema.

O método PBVS é mais suscetível à propagação de erros numéricos em comparação com o modelo IBVS. A [equação \(3.34\)](#) é uma abordagem simples e eficaz para que o robô localize seu alvo. No entanto, o maior desafio associado a este modelo (IBVS) reside na identificação de características do alvo na imagem, exigindo habilidades avançadas de aquisição e processamento de imagem (visão computacional). Definir pontos que conduzam o robô a uma determinada orientação e posição em relação ao alvo, e garantir que o sensor de visão possa detectar esses pontos, representa um desafio significativo.

3.6 Robótica

Nesta seção, serão abordadas as bases teóricas e matemáticas relacionadas ao manipulador robótico ABB IRB140, com foco na modelagem de sua cinemática e dinâmica para a

utilização do seu controlador programado no RAPID².

O cálculo do Jacobiano do manipulador robótico é essencial neste trabalho, pois conecta a cinemática do robô ao sistema de visão computacional. No contexto do controle servo visão baseado em imagem, o Jacobiano é utilizado para relacionar as velocidades das características detectadas pela câmera (no plano da imagem) com as velocidades cartesianas do efetuador do robô. Essas velocidades cartesianas são, então, convertidas em velocidades das juntas por meio do Jacobiano do robô, que também possibilita o controle do posicionamento e orientação do manipulador.

O modelo cinemático direto fornece a posição do efetuador no espaço, mas o modelo inverso, combinado com o Jacobiano, é indispensável para calcular as velocidades das juntas a partir de um vetor de velocidades cartesianas desejadas. Além disso, a integração entre o sensor de visão e o manipulador é matematicamente descrita pelo Jacobiano da câmera (\mathbf{J}_s), o qual relaciona a variação no plano da imagem ($\dot{\mathbf{p}}$) com a velocidade cartesiana do efetuador (\mathbf{v}_c), como mostrado na equação (3.15) e evidenciado a seguir:

$$\dot{\mathbf{p}} = \begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \mathbf{J}_s \mathbf{v}_c, \quad (3.35)$$

O Jacobiano do manipulador, por sua vez, é utilizado para calcular as velocidades angulares das juntas, a partir da relação:

$$\mathbf{v}_c = \mathbf{J}_m \dot{\boldsymbol{\theta}}, \quad (3.36)$$

onde \mathbf{J}_m é o Jacobiano do manipulador e $\dot{\boldsymbol{\theta}}$ é o vetor das velocidades das juntas.

Finalmente, para que o manipulador siga os pontos-alvo definidos pelo sistema de visão, o cálculo da inversa do Jacobiano do manipulador permite transformar a velocidade desejada no espaço cartesiano (\mathbf{v}_c) nas velocidades específicas de cada junta ($\dot{\boldsymbol{\theta}}$) mostrado na equação (3.37). Assim, o Jacobiano desempenha um papel central tanto na integração entre o sistema de visão e o manipulador quanto no controle preciso dos movimentos desse robô.

$$\dot{\boldsymbol{\theta}} = \mathbf{J}_m^{-1} \cdot \mathbf{J}_s^{-1} \cdot \dot{\mathbf{p}} \quad (3.37)$$

² RAPID é a linguagem de programação do RobotStudio (ABB) usada para programar robôs industriais. Ela permite controlar movimentos do manipulador tais como os movimentos lineares (MoveL), articulados (MoveJ), e circulares (MoveC), além de configurar ferramentas e sensores. Com suporte para estrutura de código, como loops e condicionais, o RAPID facilita a organização de programas em módulos e rotinas. A linguagem permite ainda comunicação com dispositivos externos e simulação no ambiente do RobotStudio, ajudando a identificar e corrigir problemas antes da execução.

3.6.1 Cinemática Direta: A Notação de Denavit-Hartenberg DH

Nesta subseção é apresentado o desenvolvimento das equações das cinemáticas diretas ou de configuração para as juntas do robô. O problema da cinemática direta é encontrar as relações entre as juntas individuais do robô manipulador e a posição e orientação do efetuador. As variáveis conjuntas são os ângulos entre os elos no caso de juntas de revolução ou rotacionais, e o elo de extensão no caso de juntas prismáticas ou deslizantes [Spong, Hutchinson e Vidyasagar \(2020\)](#).

Será desenvolvido um conjunto de convenções que fornecem um procedimento sistemático para realizar esta análise. É claro que é possível levar adiante a análise cinemática mesmo sem respeitar essas convenções, usando matrizes de transformações homogêneas. No entanto, a análise cinemática de um manipulador de múltiplos elos pode ser extremamente complexa e as convenções apresentadas na notação de DH simplificam consideravelmente a análise. Além disso, eles dão origem para uma linguagem universal com a qual os engenheiros de robôs possam se comunicar [Spong, Hutchinson e Vidyasagar \(2020\)](#).

A notação de DH é representada como um produto de quatro transformações homogêneas básicas **DH**:

$$\mathbf{DH} = \mathbf{Rot}_{Z,\theta_i} \mathbf{Tranz}_{Z,d_i} \mathbf{Tranz}_{X,l_i} \mathbf{Rot}_{X,\alpha_i} = \begin{bmatrix} C\theta_1 & -S\theta_1 & 0 & 0 \\ S\theta_1 & C\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & l_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_i & -S\alpha_i & 0 \\ 0 & S\alpha_i & C\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & l_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & l_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.38)$$

onde as transformações envolvem rotações (**Rot**) e translações (**Trans**) nos eixos Z e X , parametrizadas pelos ângulos θ_i e α_i , e pelas distâncias d_i e l_i . Sendo $C\theta_1 = \cos(\theta_1)$, $S\theta_1 = \sin(\theta_1)$, $C\alpha_1 = \cos(\alpha_1)$ e $S\alpha_1 = \sin(\alpha_1)$.

Ainda dentro da notação de DH deve-se assumir as seguintes convenções:

- O eixo x_i deve ser perpendicular ao eixo z_{i-1} ;
- O eixo x_i deve interceptar o eixo z_{i-1} .

Contudo, a aplicação da notação Denavit-Hartenberg (DH) padrão pode apresentar limitações em manipuladores robóticos com especificidades geométricas que não se adequam perfeitamente a modelos cinemáticos construídos especificamente para identificação de parâmetros geométricos em processos de calibração dos robôs. Essas limitações incluem casos em que os eixos de rotação são paralelos e consecutivos, situação em que podem surgir

descontinuidades em alguns dos parâmetros no processo de identificação, como d_i ou l_i .

Para lidar com essas peculiaridades, é recomendada a utilização de modificações na notação de Denavit-Hartenberg (MDH), sendo a notação de Hayati a mais comum (Hayati, 1986). A notação de Hayati, é particularmente útil em manipuladores com configurações complexas ou redundantes, permitindo a definição de sistemas de coordenadas que minimizam ambiguidades e simplificam a modelagem de cadeias cinemáticas que serão calibradas posteriormente. No entanto a notação de Hayati também apresenta singularidades quando eixos consecutivos são ortogonais.

Na notação MDH, a matriz de transformação homogênea é ajustada para considerar essas especificidades, como o alinhamento dos eixos de rotação e os pontos de referência dos elos, evitando ambiguidades que podem surgir na parametrização da cinemática direta. A matriz de transformação homogênea na notação MDH é expressa como:

$$\mathbf{MDH} = \mathbf{Rot}Z, \theta_i \cdot \mathbf{Trans}X, \alpha_i \cdot \mathbf{Rot}X, \alpha_i \cdot \mathbf{Rot}Y, \beta_i \quad (3.39)$$

3.6.2 Jacobiano Analítico de Manipuladores Robóticos

Matematicamente, as equações da cinemática direta definem uma função entre o espaço das posições e orientações cartesianas (ou, simplesmente, o **espaço cartesiano**) e o espaço das posições das juntas (ou, simplesmente, o **espaço das juntas**). As relações entre velocidades são, então, determinadas pelo Jacobiano dessa função Spong, Hutchinson e Vidyasagar (2020). O **Jacobiano** é uma função matricial, podendo ser imaginado como uma versão vetorial da derivada ordinária de uma função escalar. Trata-se de uma das quantidades mais importantes na análise e no controle do movimento de um robô. Ele aparece em basicamente todos os aspectos da manipulação de um robô: no planejamento e execução de trajetórias, na determinação de configurações singulares, na dedução das equações dinâmicas do movimento e na transformação de forças e torques do órgão terminal para as juntas do manipulador.

Para um manipulador com n membros, deve-se deduzir o Jacobiano que representa a transformação instantânea entre o vetor das velocidades das juntas (n componentes) e o vetor das velocidades lineares e angulares do órgão terminal (6 componentes, sendo 3 velocidades lineares e 3 velocidades angulares), ou de qualquer outro ponto do manipulador. Portanto, **o Jacobiano é uma matriz de dimensões $6 \times n$** .

As deduções das equações seguintes têm como base o **apêndice A**. Considere-se um manipulador com n variáveis das juntas, representadas pelo vetor $\mathbf{q} = [q_1 q_2 \cdots q_n]^T$ e seja a transformação do sistema do órgão terminal em relação ao sistema da base dada por

$$\mathbf{T}_0^n(\mathbf{q}) = \begin{bmatrix} \mathbf{R}_0^n(\mathbf{q}) & \mathbf{d}_0^n(\mathbf{q}) \\ 0 & 1 \end{bmatrix} \quad (3.40)$$

À medida que o robô se movimenta, tanto as variáveis das juntas, q_i , como a posição \mathbf{d}_0^n e a orientação do órgão terminal, \mathbf{R}_0^n , serão funções do tempo. O objetivo agora é relacionar as velocidades linear e angular do órgão terminal com o vetor das velocidades das juntas, $\dot{\mathbf{q}}(t)$.

Seja a velocidade linear do órgão terminal denotada por

$$\mathbf{v}_0^n = \dot{\mathbf{d}}_0^n \quad (3.41)$$

Deseja-se obter expressões nas formas

$$\mathbf{v}_0^n = \mathbf{J}_v \dot{\mathbf{q}} \quad (3.42)$$

$$\boldsymbol{\omega}_0^n = \mathbf{J}_\omega \dot{\mathbf{q}} \quad (3.43)$$

onde \mathbf{J}_v e \mathbf{J}_ω são matrizes $3 \times n$. Pode-se reunir as duas últimas equações como

$$\begin{bmatrix} \mathbf{v}_0^n \\ \boldsymbol{\omega}_0^n \end{bmatrix} = \mathbf{J}_0^n \dot{\mathbf{q}} \quad (3.44)$$

onde a matriz dada por

$$\mathbf{J}_0^n = \begin{bmatrix} \mathbf{J}_v \\ - \\ \mathbf{J}_\omega \end{bmatrix} \quad (3.45)$$

é o conhecido **Jacobiano do Manipulador**, uma matriz $6 \times n$, onde n é o número de membros do manipulador.

3.6.3 Dedução do Jacobiano (Spong, Hutchinson e Vidyasagar (2020))

Inicialmente, será determinada a parte inferior do Jacobiano da equação (3.45), \mathbf{J}_ω , referente à velocidade angular. As velocidades angulares podem ser somadas vetorialmente, desde que estejam expressas em relação a um mesmo sistema de coordenadas [apêndice A](#). Assim, pode-se determinar a velocidade angular do órgão terminal, em relação à base, expressando a velocidade angular de cada membro em relação à base e somando vetorialmente essas velocidades.

Logo, a velocidade angular do i -ésimo membro, se a junta for **rotativa**, em relação ao sistema $i - 1$, é dada por

$$\boldsymbol{\omega}_{i-1}^i = \dot{q}_i \mathbf{k} \quad (3.46)$$

Por outro lado, se a junta for **prismática**, então tal velocidade angular é nula:

$$\omega_{i-1}^i = \mathbf{0} \quad (3.47)$$

Portanto, a velocidade angular total do órgão terminal, em relação ao sistema da base, é dada por

$$\begin{aligned} \omega_0^n &= \rho_1 \dot{q}_1 \mathbf{k} + \rho_2 \dot{q}_2 \mathbf{R}_0^1 \mathbf{k} + \cdots + \rho_n \dot{q}_n \mathbf{R}_0^{n-1} \mathbf{k} \\ &= \sum_{i=1}^n \rho_i \dot{q}_i \mathbf{z}_{i-1} \end{aligned} \quad (3.48)$$

onde

$$\mathbf{z}_{i-1} = \mathbf{R}_0^{i-1} \mathbf{k} \quad (3.49)$$

denota o vetor unitário \mathbf{k} do sistema $i - 1$ expresso em relação ao sistema da base e onde:

$\rho_i = 1$ se a junta i é rotativa;

$\rho_i = 0$ se a junta i é prismática.

Assim, a metade inferior do Jacobiano da equação (3.45) é dada por

$$\mathbf{J}_\omega = [\rho_1 \mathbf{z}_0 \cdots \rho_n \mathbf{z}_{n-1}] \quad (3.50)$$

Será determinada agora a parte superior do Jacobiano da equação (3.45), \mathbf{J}_v , referente à velocidade linear. A velocidade linear do órgão terminal pode ser obtida a partir da derivação temporal do vetor posição, usando a regra da cadeia da derivação:

$$\dot{\mathbf{d}}_0^n = \sum_{i=1}^n \frac{\partial \mathbf{d}_0^n}{\partial q_i} \dot{q}_i \quad (3.51)$$

Assim, vê-se que a i -ésima coluna de \mathbf{J}_v é simplesmente

$$\frac{\partial \mathbf{d}_0^n}{\partial q_i} \quad (3.52)$$

Além disso, essa expressão é justamente a velocidade linear do órgão terminal que resulta se q_i for igual a 1 e os outros q_j forem nulos. Em outras palavras, a i -ésima coluna do Jacobiano é gerada mantendo-se todas as juntas fixas exceto a i -ésima, que é atuada com velocidade unitária. Dois casos são considerados a seguir.

Caso 1

Se a junta i é prismática, então \mathbf{R}_0^{j-1} é independente de $q_i = d_i$ para todo j , e

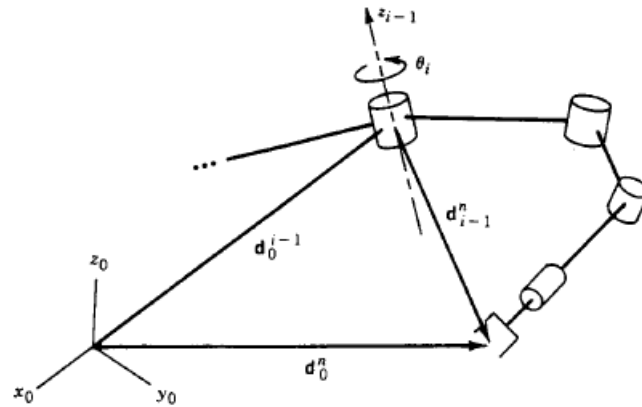


Figura 3.14 – Movimento do órgão terminal devido ao i -ésimo membro.

$$\mathbf{d}_{i-1}^i = d_i \mathbf{k} + \mathbf{R}_{i-1}^i a_i \mathbf{i} \quad (3.53)$$

Se todas as juntas forem fixadas, exceto a i -ésima, tem-se

$$\begin{aligned} \dot{\mathbf{d}}_0^n &= \mathbf{R}_0^{i-1} \dot{\mathbf{d}}_{i-1}^i \\ &= \dot{d}_i \mathbf{R}_0^{i-1} \mathbf{k} \\ &= \dot{d}_i \mathbf{z}_{i-1} \end{aligned} \quad (3.54)$$

Assim,

$$\frac{\partial \mathbf{d}_0^n}{\partial q_i} = \mathbf{z}_{i-1} \quad (3.55)$$

Caso 2

Se a junta i é rotativa, então \mathbf{o}_k denota o vetor d_0^k da origem O_0 à origem O_k para qualquer k , e pode-se então escrever

$$\mathbf{d}_0^n = \mathbf{d}_0^{i-1} + \mathbf{R}_0^{i-1} \mathbf{d}_{i-1}^n \quad (3.56)$$

ou, na nova notação:

$$\mathbf{o}_n - \mathbf{o}_{i-1} = \mathbf{R}_0^{i-1} \mathbf{d}_{i-1}^n \quad (3.57)$$

Com relação à [figura 3.14](#), que ilustra o movimento do órgão terminal devido ao membro i , observe-se que tanto d_0^{i-1} como \mathbf{R}_0^{i-1} são constantes se apenas a i -ésima junta for atuada.

Portanto, da [equação \(3.56\)](#):

$$\dot{\mathbf{d}}_0^n = \mathbf{R}_0^{i-1} \dot{\mathbf{d}}_{i-1}^n \quad (3.58)$$

Tendo em conta que o movimento do membro i é uma rotação q_i em torno de \mathbf{z}_{i-1} , tem-se

$$\dot{\mathbf{d}}_{i-1}^n = \dot{q}_i \mathbf{k} \times \mathbf{d}_{i-1}^n \quad (3.59)$$

e assim

$$\begin{aligned} \dot{\mathbf{d}}_0^n &= \mathbf{R}_0^{i-1} (\dot{q}_i \mathbf{k} \times \mathbf{d}_{i-1}^n) \\ &= \dot{q}_i \mathbf{R}_0^{i-1} \mathbf{k} \times \mathbf{R}_0^{i-1} \mathbf{d}_{i-1}^n \\ &= \dot{q}_i \mathbf{z}_{i-1} \times (\mathbf{o}_n - \mathbf{o}_{i-1}) \end{aligned} \quad (3.60)$$

Portanto,

$$\frac{\partial \mathbf{d}_0^n}{\partial q_i} = \mathbf{z}_{i-1} \times (\mathbf{o}_n - \mathbf{o}_{i-1}) \quad (3.61)$$

e a parte superior do Jacobiano, \mathbf{J}_v é

$$\mathbf{J}_v = [\mathbf{J}_{v1} \cdots \mathbf{J}_{vn}] \quad (3.62)$$

onde a i -ésima coluna é

$$\mathbf{J}_{vi} = \mathbf{z}_{i-1} \times (\mathbf{o}_n - \mathbf{o}_{i-1}) \quad (3.63)$$

se a junta for rotativa e

$$\mathbf{J}_{vi} = \mathbf{z}_{i-1} \quad (3.64)$$

se a junta for prismática.

Reunindo as metades superior e inferior do Jacobiano, foi mostrado que o Jacobiano para um manipulador de n membros tem a forma

$$\mathbf{J} = [\mathbf{J}_1 \mathbf{J}_2 \cdots \mathbf{J}_n] \quad (3.65)$$

onde a i -ésima coluna é dada por

$$\mathbf{J}_i = \begin{bmatrix} \mathbf{z}_{i-1} \times (\mathbf{o}_n - \mathbf{o}_{i-1}) \\ \mathbf{z}_{i-1} \end{bmatrix} \quad (3.66)$$

se a junta i for rotativa e

$$\mathbf{J}_i = \begin{bmatrix} \mathbf{z}_{i-1} \\ 0 \end{bmatrix} \quad (3.67)$$

se a junta i for prismática.

As fórmulas acima tornam simples a determinação do Jacobiano de qualquer manipulador, pois todas as quantidades necessárias já estão disponíveis a partir da cinemática direta. Na verdade, as únicas quantidades necessárias para calcular o Jacobiano são os vetores unitários \mathbf{z}_i e os vetores que localizam as origens $\mathbf{O}_1, \mathbf{O}_2, \dots, \mathbf{O}_n$, em relação à origem \mathbf{O}_0 . Ora, é fácil verificar que \mathbf{z}_i é dado pelos três primeiros elementos da terceira coluna da matriz \mathbf{H}_0^i , enquanto que \mathbf{o}_i é dado pelos três primeiros elementos da quarta coluna de \mathbf{H}_0^i . Portanto, apenas as terceira e quarta colunas das matrizes de transformação homogênea são necessárias para a construção do Jacobiano.

O procedimento acima funciona não apenas para calcular a velocidade do órgão terminal, mas também para determinar a velocidade de qualquer ponto do manipulador. Isso será muito importante para a determinação das velocidades dos centros de massa dos vários membros do manipulador, a fim de deduzir as equações dinâmicas do movimento.

3.7 Calibração de Robôs Industriais

A calibração de robôs industriais é um processo crítico para melhorar a precisão dos manipuladores em diversas aplicações. Embora os robôs apresentem alta repetibilidade, sua exatidão é frequentemente limitada devido a erros geométricos e não geométricos que afetam diretamente o posicionamento do efetuador. Pesquisas mostram que erros geométricos representam cerca de 90% do erro total de posicionamento, sendo causados por imperfeições de fabricação, montagem, desgaste natural e outras fontes estruturais (Hollerbach; Wampler, 1996).

Além disso, erros não geométricos, como variações térmicas, folgas e deformações, também podem impactar o desempenho, mas em menor proporção (Zhuang; Roth, 1994). Esses erros são especialmente relevantes em robôs utilizados em aplicações de alta precisão, como na indústria aeroespacial e automotiva (Angelini *et al.*, 2021).

A calibração pode ser classificada em três níveis principais, conforme as fontes de erro abordadas:

Nível 1: Calibração em nível de junta - Concentra-se na determinação de posições zero no espaço articular, assumindo que os parâmetros geométricos são ideais. Este nível é mais básico e geralmente adequado para ajustes iniciais (Driels; Pathre, 1991).

Nível 2: Calibração geométrica - Considera erros cinemáticos no espaço cartesiano, incluindo parâmetros estruturais como comprimentos de elos e orientações de eixos. Este

nível é amplamente utilizado na prática industrial devido à sua abrangência e aplicabilidade (Bernard; Mangin, 2004).

Nível 3: Calibração não geométrica - Inclui fatores como deformações, mudanças de temperatura, fricção e outras variáveis ambientais que impactam o desempenho do robô. Este nível é mais complexo devido à variabilidade desses fatores (Nubiola; Bonev, 2013).

O processo de calibração geralmente compreende quatro etapas fundamentais, descritas a seguir:

- a) **Modelagem:** Desenvolvimento de um modelo matemático que relacione os movimentos das juntas à posição e orientação do efetuador. Esse modelo pode ser baseado em notações como Denavit-Hartenberg (DH) ou Modified Denavit-Hartenberg (MDH), sendo esta última a modificação proposta por Hayati (Hayati, 1986). A notação DH é amplamente utilizada pela sua simplicidade, mas pode apresentar limitações em robôs com configurações geométricas complexas, como eixos paralelos ou quase perpendiculares. A MDH ajusta essas limitações ao permitir maior flexibilidade e evitar singularidades em tais configurações.
- b) **Medição:** Consiste em coletar dados sobre o posicionamento real do robô utilizando instrumentos de alta precisão, como Laser-Trackers, sistemas de visão computacional, braços de medição coordenada ou sensores redundantes (Schweikard; Bova *et al.*, 1992). Esses dispositivos permitem identificar discrepâncias entre o modelo teórico e a realidade física do robô.
- c) **Identificação:** Com base nos dados medidos, os parâmetros de erro são determinados utilizando métodos matemáticos e computacionais, como regressão linear, algoritmos de otimização ou técnicas de identificação paramétrica (Shirinzadeh *et al.*, 2004). Essa etapa é crucial para ajustar o modelo cinemático de forma precisa.
- d) **Compensação:** As correções calculadas são aplicadas ao controlador do robô ou ao seu modelo matemático, permitindo que o manipulador compense os erros identificados e alcance a precisão desejada (Everett *et al.*, 1987).

Cada uma dessas etapas desempenha um papel essencial no processo de calibração. A modelagem fornece a base teórica, enquanto a medição garante a coleta de dados confiáveis. A identificação permite traduzir esses dados em parâmetros úteis, e a compensação aplica esses ajustes na prática.

A escolha dos métodos e ferramentas de calibração depende das características do robô e dos requisitos específicos da aplicação. Por exemplo, a integração de sensores externos, como câmeras e acelerômetros, tem ganhado destaque por oferecer medições em tempo real e reduzir a dependência de equipamentos de alto custo. Essas tecnologias permitem que a calibração seja mais eficiente e acessível, particularmente em ambientes industriais dinâmicos (Bender *et al.*, 2013).

A norma [ISO9283 \(1998\)](#) é frequentemente usada como referência para avaliar o desempenho posicional dos robôs, classificando critérios como precisão, repetibilidade e erros de orientação. Essa padronização é essencial para validar os resultados da calibração e garantir que os robôs atendam aos requisitos industriais.

A calibração de robôs industriais evoluiu consideravelmente com a adoção de tecnologias modernas, como o laser tracking. Esse tipo de equipamento utiliza feixes de laser para medir com alta precisão a posição de alvos montados no robô, permitindo a determinação exata de sua posição e orientação em relação ao espaço de trabalho. Essa tecnologia é amplamente utilizada devido à sua precisão submilimétrica e sua capacidade de realizar medições em tempo real, mesmo em ambientes dinâmicos. Além disso, o laser tracking se destaca pela facilidade de integração com softwares de análise e controle, permitindo a automação de processos de calibração e a redução significativa do tempo necessário para ajustes manuais. Essas características tornam o laser tracker uma solução indispensável em aplicações industriais que exigem alta precisão, como na montagem de peças aeronáuticas e no controle de qualidade em linhas de produção.

A calibração de robôs industriais tem avançado significativamente com o uso de tecnologias modernas, garantindo maior precisão e eficiência nos processos. Além do rastreamento a laser (laser tracking), amplamente utilizado por sua alta precisão e medição em tempo real com equipamentos como o Radian Laser Tracker da API Metrology e o Vantage Laser Tracker da FARO, outras tecnologias têm se destacado. Braços de medição portáteis, como o FARO Arm e o Hexagon Romer, permitem medições diretas e flexíveis com sondas táteis ou lasers. Sistemas de medição óptica 3D, como os da Creafom MetraSCAN 3D, utilizam câmeras de alta resolução para capturar informações tridimensionais, sendo ideais para volumes de trabalho maiores. Tecnologias baseadas em infravermelho, como o Vicon Motion Systems, rastreiam marcadores reflexivos para medir movimentos dinâmicos com precisão. Sistemas inerciais de navegação (INS), frequentemente acoplados a câmeras, também oferecem dados sobre aceleração, velocidade angular e posição relativa, sendo úteis em calibrações específicas. Por fim, dispositivos de nivelamento de alta precisão, como os da Wyler e Niveltronic, são empregados para ajustar inclinações e garantir posicionamento milimétrico. Essas tecnologias complementares permitem adaptar o processo de calibração às necessidades específicas, sejam elas ambientes dinâmicos, robôs de alta velocidade ou operações em grandes áreas.

Os equipamentos utilizados na calibração de robôs industriais apresentados anteriormente, como rastreadores a laser, braços de medição portáteis e sistemas ópticos ou infravermelhos, são altamente precisos, mas também apresentam custos elevados, o que pode limitar sua aplicação em determinados contextos. Nesse sentido, o sensor proposto busca oferecer uma alternativa mais acessível, mantendo resultados satisfatórios em termos de precisão e eficiência, contribuindo para a viabilidade de calibração em ambientes com restrições orçamentárias.

Dessa forma, a calibração de robôs industriais evoluiu para integrar avanços tecnológicos e atender às demandas crescentes por precisão e eficiência. Métodos baseados em notações avançadas e sensores modernos têm possibilitado um aprimoramento contínuo no desempenho dos manipuladores, contribuindo significativamente para a automação de processos industriais complexos.

4 MEDIÇÃO DE POSIÇÃO E ORIENTAÇÃO DE ROBÔS POR VISÃO E CONTROLE DE MOVIMENTO

Aqui será apresentado todo desenvolvimento na criação do sensor para medição de posição e orientação do robô. Primeiramente serão apresentados todos os equipamentos utilizados na montagem do sensor, incluindo a câmera utilizada com laser acoplado e também todas as características necessárias do manipulador robótico utilizado no trabalho. Posteriormente será apresentada a metodologia utilizada para a criação do sensor, incluindo o procedimento de montagem física e o algoritmo para obter a profundidade da imagem Z em relação posição do centro óptico da câmera com utilização de laser. Logo em seguida todo o design do algoritmo Servo-Visual Baseado em Imagem (IBVS) gerando as velocidade necessárias para movimentação do manipulador robótico IRB140. E por fim, a apresentação da sistemática robótica necessária para movimentação de seus eixos com o sensor de visão acoplado em seu efetuador mudando seu posicionando gradativamente para atingir o alvo.

4.1 Equipamentos

O sensor é composto por uma câmera, uma lente e um laser. A câmera é do modelo acA2500, fabricado pela empresa Basler que dispõe de uma interface USB 3.0, um sensor ON Semiconductor MT9P031 mono CMOS e consegue entregar 14 frames por segundo com uma resolução de 5MP. Acoplada na câmera, utilizou-se uma lente do modelo CFFL F1.4 f16mm 2/3", que possui uma distância focal fixa de 16mm e um intervalo de abertura de F1.4 - F16. A [figura 4.1](#) em conjunto com a [tabela 4.1](#), a [tabela 4.2](#) e a [tabela 4.3](#) apresentam todos os componentes que formam o sensor de visão enquanto a [tabela 4.4](#) apresenta o computador e os softwares utilizados.

O manipulador robótico ABB IRB 140, utilizado neste projeto, é um robô industrial multifuncional de 6 eixos, projetado para operar dentro de um volume de trabalho determinado por suas dimensões construtivas [Robotics \(2003\)](#), conforme ilustrado na [figura 4.2](#).

4.2 Modelagem do Sensor para Obtenção da Profundidade da Imagem

A abordagem utilizada neste trabalho é baseada na triangulação descrita no artigo [Forsman et al. \(2001\)](#). Nesse método, o processo de cálculo da profundidade exige apenas um ponto de referência para realizar o equacionamento. A simplificação da metodologia

Figura 4.1 – Equipamentos utilizados para construção do sensor: a) Câmera acA2500 - 14 μ m; b) Laser LDM115 e gerador de linha óptico LGO115 e c) Lente CFFL F1.4 f16mm 2/3”.



Tabela 4.1 – Especificações da Câmera.

Especificações da Câmera	
Resolução (HxV pixels)	2592 x 1944
Tipo do Sensor	ON Semiconductor MT9P031 Progressive scan CMOS Rolling shutter
Tamanho do Sensor óptico	1/2500”
Tamanho Efetivo na Diagonal	7.2mm
Tamanho do Pixel (HxV)	2.2 μ m x 2.2 μ m
Taxa de Quadros Padrão	14fps
Mono/Color	Mono
Interface	USB 3.0, Taxa máx. 5Gbit/s
Alimentação	5VDC, via USB
Dimensões	29.3mm x 29mm x 29mm (sem lentes ou conectores)
Peso	<80g

Tabela 4.2 – Especificações da Lente.

Especificações da Lente	
Distância Focal	16.0mm
Abertura	F1.4 - F16.0
Tipo de Abertura	Manual
Círculo de Imagem Máximo	2/3”
Peso	74g

foi feita utilizando um feixe de laser projetado no espaço tridimensional, que se mostrou mais prático para detecção de linha gerada por esse próprio laser durante a aquisição e o processamento de imagem.

No sistema desenvolvido, o laser projeta um plano de luz no espaço, formando uma linha no plano do objeto (figura 4.4). A localização exata desse ponto na imagem é identificada

Tabela 4.3 – Especificações do Laser.

Especificações do Laser	
Comprimento de Onda	633nm
Dimensões	37mm de comprimento e 11mm de diâmetro
Classe do Laser - CDRH	IIIa
Cor	Vermelho
Precisão de Apontamento	<25mrad
Corrente de Operação	<70mA
Alimentação	3.5 - 5VDC
Peso	9.5g

Tabela 4.4 – Hardware e Software Utilizados

Especificações do Hardware	
Computador	Notebook HP
Modelo	HP 246 G6
Processador	Intel Core I3 - 6006U Dual Core 2.0 GHz
Memória RAM	4GB DDR4
Placa de Vídeo	Integrada, Intel HD Graphics 520
Especificações dos Softwares	
Programa Operacional	Windows 10
Software Matemático	Matlab 2023b
Software Robótico	RobotStudio 2023
Software da Câmera	Basler pylon Camera Software Suite (versão 7.0)

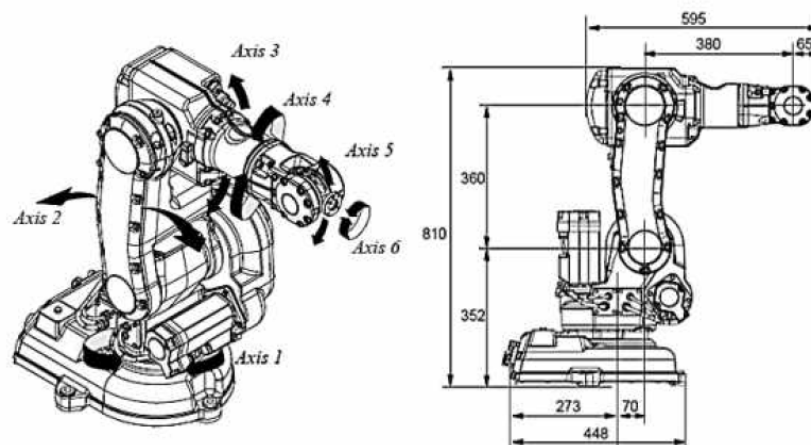


Figura 4.2 – Desenho técnico do robô industrial ABB IRB 140 em milímetros (Robotics, 2003).

pela interseção da linha gerada pelo laser com uma linha horizontal previamente definida, que passa pelo ponto principal da câmera no sistema de referência da câmera.

Essa abordagem oferece vantagens significativas em relação a métodos tradicionais:

1. **Detecção Simples e Confiável:** O ponto gerado pelo laser é fácil de identificar na imagem devido ao contraste significativo entre o feixe de luz e o fundo.

2. **Redução da Complexidade Computacional:** Como apenas um ponto é necessário para o cálculo, a quantidade de dados processados é significativamente reduzida.
3. **Maior Robustez ao Ruído:** O uso do laser minimiza a interferência de iluminação ambiente, garantindo maior precisão mesmo em condições variáveis.

A posição do ponto detectado na imagem é então usada, junto com as equações de projeção em perspectiva, para calcular a profundidade do ponto no espaço tridimensional.

Para o modelo de triangulação escolhido, o laser precisa estar inclinado em relação a câmera, por essa razão, desenvolveu-se um mecanismo de ajuste manual para garantir a movimentação angular do dispositivo. Esse mecanismo consiste em um fuso, que ao ser apertado, permite que o laser fique fixo na posição em que se encontra e ao ser afrouxado, permite a alteração da sua inclinação. Nesse modelo, a câmera não deve se movimentar, por isso ela foi fixada diretamente na caixa.

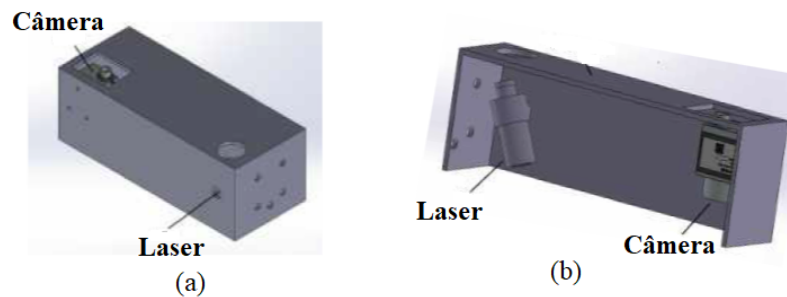


Figura 4.3 – Sensor de visão: a) Vista isométrica; b) Vista interna.

Sistemas de medição baseados em triangulação consistem em configurações de dois receptores passivos ou de um transmissor e um receptor que, juntamente com o local da medição, formam um triângulo.

Na [figura 4.4](#), f é a distância focal da câmera; b , a distância da fonte de luz com relação a câmera e β , o ângulo do feixe de luz com respeito ao eixo x do *frame* da câmera. As coordenadas do ponto medido, com respeito ao sistema global de referência da câmera, são dadas por X e Z (assume-se a coordenada Y como zero) [Forsman et al. \(2001\)](#). A partir do triângulo formado, tem-se:

$$\tan \beta = \frac{Z}{b - X} \quad (4.1)$$

Da equação da câmera perspectiva [equação \(3.1\)](#) adaptada para as variáveis em estudo:

$$(u,v)^T = \left(f \frac{X}{Z}, f \frac{Y}{Z} \right)^T \quad (4.2)$$

Substituindo na equação de triangulação:

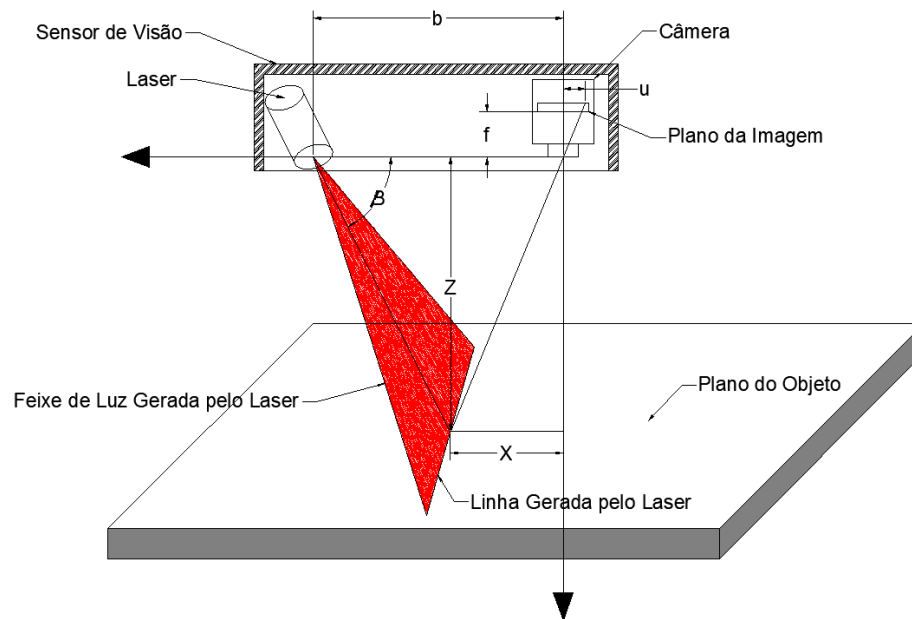


Figura 4.4 – Sensor de Medição baseado em triangulação. Fonte: Elaboração própria.

$$\tan \beta = \frac{Z}{b - \frac{uZ}{f}} \quad (4.3)$$

Isolando Z :

$$Z = \frac{b}{\frac{1}{\tan \beta} + \frac{u}{f}} \quad (4.4)$$

Sendo u a distância horizontal em pixel do centro focal até o ponto de projeção da luz laser formado na imagem.

Com essa adaptação, o sistema se torna mais eficiente, prático e robusto, especialmente para aplicações industriais que demandam detecção precisa e rápida com baixo custo computacional. Por fim, vale ressaltar que a variabilidade do tamanho da linha laser, dependente da distância do alvo, foi integrada ao modelo matemático, permitindo ajustes dinâmicos na detecção e cálculo da profundidade. Essa flexibilidade foi essencial para lidar com diferentes cenários durante os testes experimentais e validação do sistema.

A escolha dos equipamentos teve impacto direto no desempenho e na reprodutibilidade do sistema. A câmera Basler, por exemplo, garantiu a precisão necessária para identificar os pontos característicos na imagem com alta resolução, o que foi fundamental para minimizar erros no controle baseado em visão (IBVS). A estabilidade do laser LDM115 contribuiu para reduzir interferências externas e aumentou a confiabilidade da detecção do ponto de interseção na imagem. Além disso, o hardware computacional assegurou a

execução em tempo real, mesmo com a complexidade dos algoritmos de triangulação e controle.

4.3 Modelagem do Sensor para o Controle Servo-Visual Baseado em Imagem IBVS

Um parâmetro importante para esta dissertação é o tipo de alvo utilizado e foi apresentado na [tabela 2.3](#). O tipo de alvo será definido aqui com base na bibliografia apresentada anteriormente na [seção 3.5](#) para obtenção do IVBS e cuja característica principal está no inverso da matriz jacobiana da imagem. A matriz jacobiana da imagem [equação \(3.30\)](#) tem dimensão 2×6 . Com a utilização de alvo com número de pontos, n , na imagem igual ou superior a três obtemos uma matriz jacobiana da imagem com dimensão $2n \times 6$ sendo $n \geq 3$. Isso faz com que o Jacobiano da imagem seja facilmente inversível. Com utilização de n pontos como alvo as equações do modelo serão:

$$\begin{bmatrix} \dot{u}_1 \\ \dot{v}_1 \\ \dot{u}_2 \\ \dot{v}_2 \\ \vdots \\ \dot{u}_n \\ \dot{v}_n \end{bmatrix} = \begin{bmatrix} \mathbf{J}_s(u_1, v_1, Z_1) \\ \mathbf{J}_s(u_2, v_2, Z_2) \\ \vdots \\ \mathbf{J}_s(u_n, v_n, Z_n) \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (4.5)$$

sendo \mathbf{J}_s a matriz jacobiana da imagem [equação \(3.30\)](#).

Isolando o vetor de velocidade da câmera em função do vetor de velocidade em pixels, tem-se:

$$\begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \left(\begin{bmatrix} \mathbf{J}_s(u_1, v_1, Z_1) \\ \mathbf{J}_s(u_2, v_2, Z_2) \\ \vdots \\ \mathbf{J}_s(u_n, v_n, Z_n) \end{bmatrix} \right)^{-1} \begin{bmatrix} \dot{u}_1 \\ \dot{v}_1 \\ \dot{u}_2 \\ \dot{v}_2 \\ \vdots \\ \dot{u}_n \\ \dot{v}_n \end{bmatrix} \quad (4.6)$$

A velocidade desejada das coordenadas em pixel, em função das coordenadas pixel atuais (u_i, v_i) e das coordenadas que se desejam alcançar (u_i^*, v_i^*) pode ser escrita como:

$$\begin{bmatrix} \dot{u}_i \\ \dot{v}_i \end{bmatrix} = \lambda \left\{ \begin{bmatrix} u_i^* \\ v_i^* \end{bmatrix} - \begin{bmatrix} u_i \\ v_i \end{bmatrix} \right\} \quad (4.7)$$

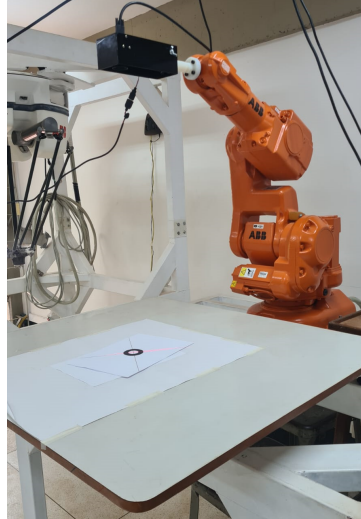


Figura 4.5 – Robô IRB140 com sensor de visão acoplado.

, em que λ é um escalador positivo. A equação (4.7) se baseia no método linear simples mostrado na equação (3.33) para determinar a velocidade da câmera.

Para simplificar, pode-se escrever a equação (4.7) da seguinte forma:

$$\begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \lambda \begin{bmatrix} \mathbf{J}_s(u_1, v_1, Z_1) \\ \mathbf{J}_s(u_2, v_2, Z_2) \\ \mathbf{J}_s(u_3, v_3, Z_3) \\ \vdots \\ \mathbf{J}_s(u_n, v_n, Z_n) \end{bmatrix}^{-1} \begin{bmatrix} p_1^* - p_1 \\ p_2^* - p_2 \\ p_3^* - p_3 \\ \vdots \\ p_n^* - p_n \end{bmatrix} \quad (4.8)$$

obtendo-se, assim, a velocidade da câmera necessária para que haja o deslocamento entre os pontos \mathbf{p}_i e \mathbf{p}_i^* .

Maiores informações para o modelamento do sensor de visão serão apresentadas no apêndice B com o algoritmo criado em *MATLAB*.

4.4 Manipulador ABB IRB140

Para obter as velocidades geradas pelo modelo IVBS ao efetuador final, que nesse caso, se trata da câmera acoplada ao manipulador robótico (figura 4.5) é necessário obter as velocidades de juntas do IRB140.

O fato do modelo obtido com IVBS utilizar somente velocidades para alcançar o alvo ausenta a necessidade de executar a cinemática inversa da posição para o robô em cada iteração. Somente é necessário obter as velocidades das juntas para alcançar as velocidades do sensor de visão no efetuador do manipulador até que o modelo IVBS diminua gradativamente o sinal de erro até alcançar o alvo.

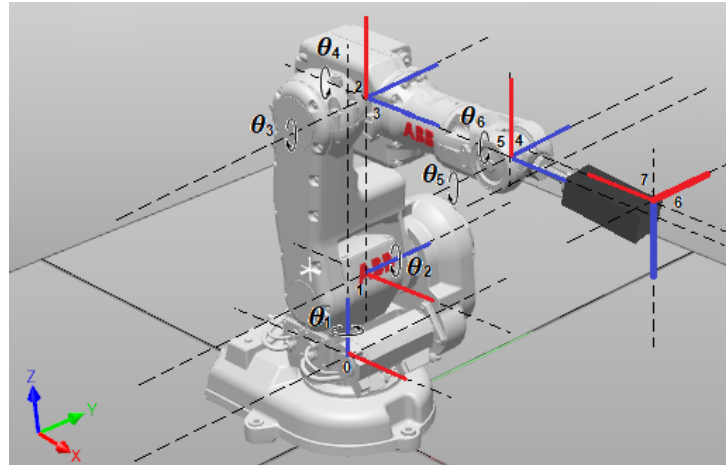


Figura 4.6 – Alocação de eixos da notação Denavit-Hartenberg para o robô IRB140 com sensor de visão acoplado.

As equações a seguir são baseadas na [seção 3.6](#).

As velocidades das juntas necessárias que moverão o efetuador do manipulador para as velocidades da câmera gerado pelo IVBS é obtido com o Jacobiano do robô. Com base na [equação \(3.66\)](#) o Jacobiano analítico do robô IRB140 é definido por:

$$\mathbf{J} = \begin{bmatrix} \mathbf{z}_0 \times (\mathbf{o}_6 - \mathbf{o}_0) & \mathbf{z}_1 \times (\mathbf{o}_6 - \mathbf{o}_1) & \mathbf{z}_2 \times (\mathbf{o}_6 - \mathbf{o}_2) \\ \mathbf{z}_0 & \mathbf{z}_1 & \mathbf{z}_2 \\ \mathbf{z}_3 \times (\mathbf{o}_6 - \mathbf{o}_3) & \mathbf{z}_4 \times (\mathbf{o}_6 - \mathbf{o}_4) & \mathbf{z}_5 \times (\mathbf{o}_6 - \mathbf{o}_5) \\ \mathbf{z}_3 & \mathbf{z}_4 & \mathbf{z}_5 \end{bmatrix} \quad (4.9)$$

O vetor \mathbf{o}_i é dado pelos 3 primeiros elementos da última coluna da [equação \(4.10\)](#):

$${}^0\mathbf{T}_1 \cdot {}^1\mathbf{T}_2 \dots {}^{i-1}\mathbf{T}_i = {}^0\mathbf{T}_i \quad (4.10)$$

com ${}^{i-1}\mathbf{T}_i$ sendo a matriz homogênea da cinemática direta do manipulador robótico da junta $i - 1$ a junta i .

E o vetor \mathbf{z}_i é dado pelos 3 primeiros elementos da terceira coluna da [equação \(4.10\)](#).

Os parâmetros da notação de Denavit-Hartenberg ([seção 3.6.1](#)) para o manipulador robótico IRB140 estão apresentados na [tabela 4.5](#). Esses valores foram obtidos com base na [figura 4.6](#), utilizando os dados construtivos do manipulador ilustrados na [figura 4.2](#) e o sensor [figura 4.2](#). Além disso, considera-se que o sensor possui dimensões construtivas específicas, com uma distância da câmera de 280 mm em relação à base que o conecta ao eixo 6 e um deslocamento de 15 mm na direção do sei eixo. As equações de transformação de cinemática direta entre as juntas do manipulador IRB140 estão detalhadas no Apêndice [A](#), permitindo uma descrição precisa da relação posicional entre os elos.

Tabela 4.5 – Parâmetros de Denavit-Hartenberg para o robô IRB140

Junta	$\alpha_i [^\circ]$	$l_i [mm]$	$d_i [mm]$	$\theta_i [^\circ]$	Limites [°]
1	-90	70	352	θ_1	± 180
2	0	360	0	$\theta_2 - 90$	-90 a +110
3	-90	0	0	θ_3	-230 a +50
4	90	0	380	θ_4	± 200
5	-90	0	0	θ_5	± 120
6	0	15	345	$\theta_6 - 90$	± 400
sensor	0	0	0	-90	-

5 EXPERIMENTOS E RESULTADOS

Esta seção apresenta os experimentos realizados e os resultados obtidos ao longo da pesquisa. Os experimentos foram cuidadosamente planejados e executados para validar as hipóteses formuladas e avaliar a eficácia das metodologias propostas. Inicialmente, descrevem-se os procedimentos experimentais, incluindo a configuração dos equipamentos, as condições de teste e os critérios de avaliação. Em seguida, são apresentados os dados coletados, acompanhados de análises gráficas detalhadas que sustentam as conclusões. Os resultados são discutidos à luz das expectativas teóricas e comparados com estudos prévios, destacando as contribuições e implicações práticas da pesquisa.

5.1 Metodologia Experimental

A lógica de programação que governa a interação entre os algoritmos MATLAB e RAPID opera da seguinte maneira: o *RAPID* transmite os dados na forma de *string* referentes à posição e orientação do efetuador do robô (no caso específico, o sensor de visão mencionado), enquanto o algoritmo do *MATLAB* recebe e converte esses dados em linguagem matemática. Em seguida, esse algoritmo realiza os cálculos envolvendo o modelo de triangulação e do IVBS, determina as velocidades da câmera e executa o cálculo do Jacobiano analítico inverso do manipulador IRB140 para obter as velocidades das juntas. Posteriormente, essas velocidades são convertidas em dados do tipo *string* e transmitidas de volta para o *RAPID*. O *RAPID*, então, recebe esses dados, os converte em um formato adequado para a movimentação do robô e inicia o processo de movimentação. Após a conclusão da movimentação, o *RAPID* novamente transmite os dados referentes à posição e orientação, agora na forma de *string*, de volta para o *MATLAB*.

Essa interação é repetida até que a imagem capturada pela câmera e o alvo pré-determinado se tornem coincidentes, reduzindo o erro a zero. O diagrama de fluxo mostrado na [figura 5.1](#) ilustra de forma esquemática as interações e a comunicação entre os programas.

5.2 Planejamento para coleta de dados de interesse

O planejamento experimental foi desenvolvido para atender aos objetivos propostos. O experimento tem como objetivo principal avaliar a repetibilidade do manipulador robótico ABB IRB 140, além de demonstrar sua capacidade de atingir consistentemente um alvo definido no espaço.

5.2.1 Configuração do Alvo e Condições Experimentais

Para os experimentos, o alvo foi configurado como dois círculos concêntricos:

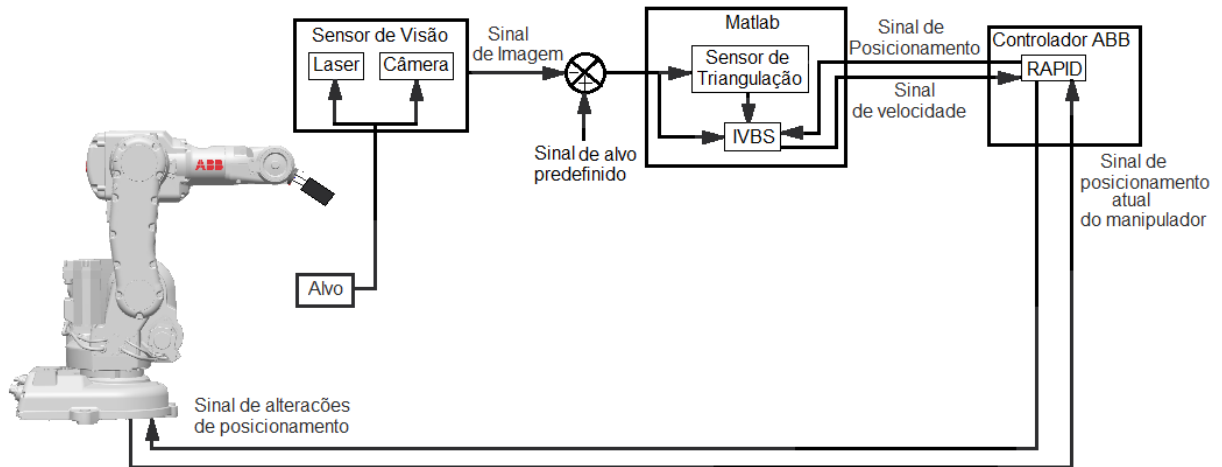


Figura 5.1 – Fluxograma do sistema.

- O círculo externo apresenta fundo preto, proporcionando contraste para detecção pelo algoritmo MATLAB;
- O círculo interno possui fundo branco com uma cruz central formada pela interseção de duas retas, definindo o centro do alvo.

O objetivo do experimento é garantir que o centro do círculo interno, representado pelo ponto de interseção das retas, seja alinhado ao ponto principal da câmera. Adicionalmente, o feixe de laser acoplado ao manipulador deve atravessar esse mesmo ponto, validando o alinhamento e a manutenção de uma profundidade Z constante entre a câmera e o alvo.

5.2.2 Aquisição de Dados

Os dados foram coletados ao longo de múltiplas repetições para analisar a capacidade do manipulador robótico em atingir o alvo com consistência. Para cada repetição, foram registrados:

- As posições finais das juntas do manipulador robótico, relativas à sua configuração inicial;
- As posições cartesianas finais da câmera, no sistema de coordenadas do manipulador;
- As orientações finais da câmera, expressas em ângulos (roll, pitch, yaw);
- A profundidade Z da imagem, calculada como a distância entre a câmera e o ponto central do alvo.

5.2.3 Critérios de Avaliação

Os dados coletados foram analisados para verificar:

- **Precisão:** A consistência na distância entre a câmera e o alvo é garantida pelo alinhamento preciso entre o centro do alvo e o ponto principal da câmera, combinado com

o feixe de laser atravessando esse ponto de interseção. Esse alinhamento assegura que o sistema mantenha a mesma profundidade em cada posicionamento;

- **Repetibilidade:** Estudo da variabilidade nas posições finais alcançadas em diferentes repetições.

Métricas estatísticas, como média, desvio padrão, mediana e moda, foram calculadas para os seguintes parâmetros:

- Posições cartesianas finais (x,y,z) ;
- Orientações finais (roll, pitch, yaw);
- Profundidade Z da imagem.

5.2.4 Utilização dos Resultados para Calibração Robótica

Os resultados obtidos neste experimento podem ser aplicados no processo de calibração de manipuladores robóticos. A repetibilidade da profundidade Z , em conjunto com o alinhamento consistente da câmera ao ponto principal do alvo, permite:

- Ajustar modelos cinemáticos do manipulador, corrigindo discrepâncias entre as posições previstas e as alcançadas;
- Refinar a precisão no alinhamento da câmera com o alvo, especialmente para aplicações que exigem posicionamento robusto e confiável.

A repetibilidade do robô ao atingir o alvo é fundamental para validar ajustes no modelo cinemático e garantir alta precisão em futuras aplicações. Essas informações podem ser usadas para ajustar parâmetros do modelo do manipulador, considerando variações nos comprimentos dos elos, offsets das juntas ou características específicas do sistema de visão.

5.3 Resultados

O experimento foi desenvolvido com o objetivo de avaliar a repetibilidade com precisão do manipulador robótico ABB IRB140 utilizando um sistema de visão computacional baseado em servo visão. O procedimento experimental foi implementado com cinco pontos característicos definidos na imagem, conforme descrito a seguir:

5.3.1 Definição do Alvo e Pontos Característicos

O alvo utilizado foi composto por dois círculos concêntricos: o círculo externo maior (com fundo preto) e o círculo interno menor (com fundo branco), contendo uma cruz no centro, como mostrado na [figura 5.2](#). Os cinco pontos característicos definidos foram:

- O centro do alvo.
- As quatro interseções entre as retas da cruz e a circunferência do círculo maior.

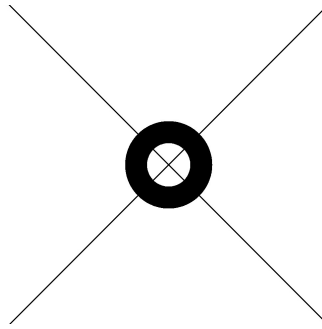


Figura 5.2 – Alvo utilizado para obtenção de pontos característicos.

Antes de iniciar cada experimento, foi alinhado manualmente com o ponto principal da câmera. Esse alinhamento foi essencial para garantir que o manipulador atingisse consistentemente essa posição e também que a imagem em interesse estivesse no campo de visão da câmera. Além disso, serviu como base para determinar as posições desejadas dos pontos característicos a serem utilizados no Jacobiano da imagem, conforme descrito na [seção 4.3](#).

5.3.2 Execução dos Experimentos

Foram realizados cinco experimentos distintos, cada um com 12 repetições. Em cada repetição, os seguintes passos foram seguidos:

Posição inicial: O manipulador inicia seu movimento em posições previamente delimitadas, dentro de um volume de trabalho mostrado na [figura 5.3](#). As posições das juntas foram selecionadas de forma distinta para cada experimento. A [tabela 5.1](#) apresenta as posições iniciais das juntas do robô ABB IRB 140 escolhidas para cada teste, enquanto a [tabela 5.2](#) detalha as posições e orientações iniciais da câmera acoplada ao efetuador resultantes dessas configurações.

Tabela 5.1 – Posições iniciais das juntas do manipulador para os 5 experimentos (em graus).

Exp.	Junta 1[°]	Junta 2[°]	Junta 3[°]	Junta 4[°]	Junta 5[°]	Junta 6[°]
1	1,0000	1,5000	-30,0000	1,5000	30,0000	-1,0000
2	-5,7000	0,9000	-27,1000	1,0000	30,6000	0,8000
3	11,9000	6,0000	-46,2000	23,6000	47,3000	17,9000
4	4,0000	0,1000	-42,7000	0,7000	33,2000	-0,7000
5	-1,0000	-6,0000	-30,6000	0,0000	32,8000	-4,0000

Tabela 5.2 – Posições e orientações iniciais da câmera para os 5 experimentos.

Exp.	X[mm]	Y[mm]	Z[mm]	Roll[°]	Pitch[°]	Yaw[°]
1	757,5590	32,7176	884,3310	0,0048	0,0261	0,0048
2	758,4880	-57,5356	853,3130	0,0087	0,0762	0,0087
3	733,2370	65,4175	928,8620	0,0078	0,0763	0,0078
4	687,7290	65,4488	1025,5100	0,0095	-0,1638	0,0095
5	681,9100	3,0687	958,4160	-0,0711	-0,0649	-0,0711

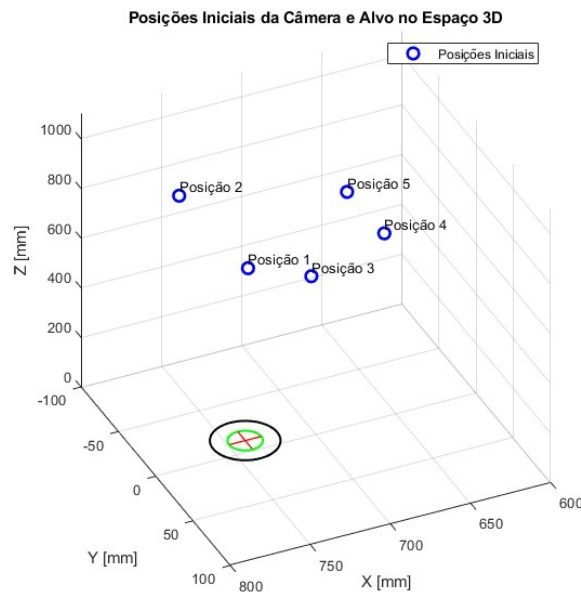


Figura 5.3 – Demonstração Esquemática de Posições Iniciais da Câmera.

Controle baseado em servo visão: A câmera capturou imagens do alvo a cada iteração. As diferenças entre as posições atuais e desejadas dos cinco pontos característicos foram usadas para calcular as velocidades necessárias e corrigir o posicionamento da câmera.

Convergência: A posição final foi alcançada quando o centro do alvo coincidiu com o ponto principal da câmera e o feixe de laser atravessou o centro do alvo, garantindo consistência na profundidade Z . Essa convergência é garantida por meio de movimentações nos eixos tridimensionais, acompanhadas de ajustes nas orientações da câmera ao longo do movimento do robô. Antes de cada experimento, a postura da posição final foi previamente determinada com base nas características do alvo, de modo a assegurar a precisão da convergência ao término de cada repetição.

Durante cada repetição, o controle baseado em servo visão foi realizado iterativamente, ajustando a posição e orientação do efetuador até alcançar a convergência. O número de iterações dentro de cada repetição foi variável, dependendo de fatores como as condições de iluminação no laboratório, a precisão dos dados recebidos do controlador RAPID, e as características específicas do movimento realizado pelo manipulador.

Assim, todos os experimentos consistiram de 12 repetições, as figuras apresentadas (5.4) no texto mostram as iterações da primeira repetição do experimento 1 como exemplo típico para ilustrar o comportamento do sistema. As iterações das primeiras repetições dos experimentos 2 a 5, que apresentam comportamento semelhante, estão documentadas no [apêndice D](#). O número exato de iterações variou em cada repetição, pois o algoritmo continuava ajustando o posicionamento até atingir os critérios de convergência previamente estabelecidos. Esse processo reflete a robustez do controle, capaz de lidar com variações nas

condições experimentais.

5.3.3 Coleta de Resultados

Durante os experimentos, os seguintes resultados foram coletados com base no primeiro experimento, sendo os resultados dos demais experimentos semelhantes e detalhados no [apêndice D](#):

Imagens capturadas - A [figura 5.4](#) apresenta as imagens capturadas pela câmera ao longo do movimento durante a primeira repetição do experimento 1. Essas imagens documentam o alinhamento progressivo entre o ponto central da câmera e o centro do alvo, além do ajuste do laser para sua posição final no alvo desejado. Resultados similares foram obtidos nos demais experimentos.

Posições cartesianas finais da câmera - As coordenadas absolutas X , Y e Z da câmera em relação à base do manipulador foram registradas ao término de cada repetição. Esses valores são apresentados nas [figuras 5.5](#), [5.6](#) e [5.7](#) para o experimento 1, ilustrando a precisão e consistência espacial alcançadas. Resultados equivalentes foram observados nos experimentos 2 a 5.

Profundidade estimada Z - A profundidade Z , ou distância absoluta entre a câmera e o centro do alvo, foi estimada para cada repetição do experimento 1 conforme a [seção 4.2](#). A [figura 5.8](#) apresenta o histograma das profundidades estimadas, destacando a uniformidade dos valores obtidos. Os experimentos restantes apresentaram comportamentos similares.

Orientações finais da câmera - Os ângulos de orientação da câmera (roll, pitch e yaw) foram registrados ao final de cada repetição do experimento 1 e estão representados nas [figuras 5.9](#), [5.10](#) e [5.11](#). Esses gráficos evidenciam a estabilidade nas orientações finais alcançadas. As orientações obtidas nos demais experimentos foram consistentes com essas.

Posições finais das juntas do robô - Os ângulos das seis juntas do manipulador foram registrados após cada repetição do experimento 1. Esses valores possibilitaram a análise da consistência dos movimentos do robô, conforme ilustrado nas [figuras 5.12](#) a [5.17](#). Resultados semelhantes foram obtidos nas demais análises realizadas nos experimentos 2 a 5.

Análise Estatística - As [tabelas 5.3](#) à [5.14](#), apresentam os resultados da análise do afastamento em relação à média para cada parâmetro avaliado nos experimentos. Para cada dado, foi calculada a diferença absoluta entre os valores das doze repetições e a média correspondente. Com base nesses cálculos, são reportados: a média dos afastamentos, o maior afastamento observado e o menor afastamento registrado.

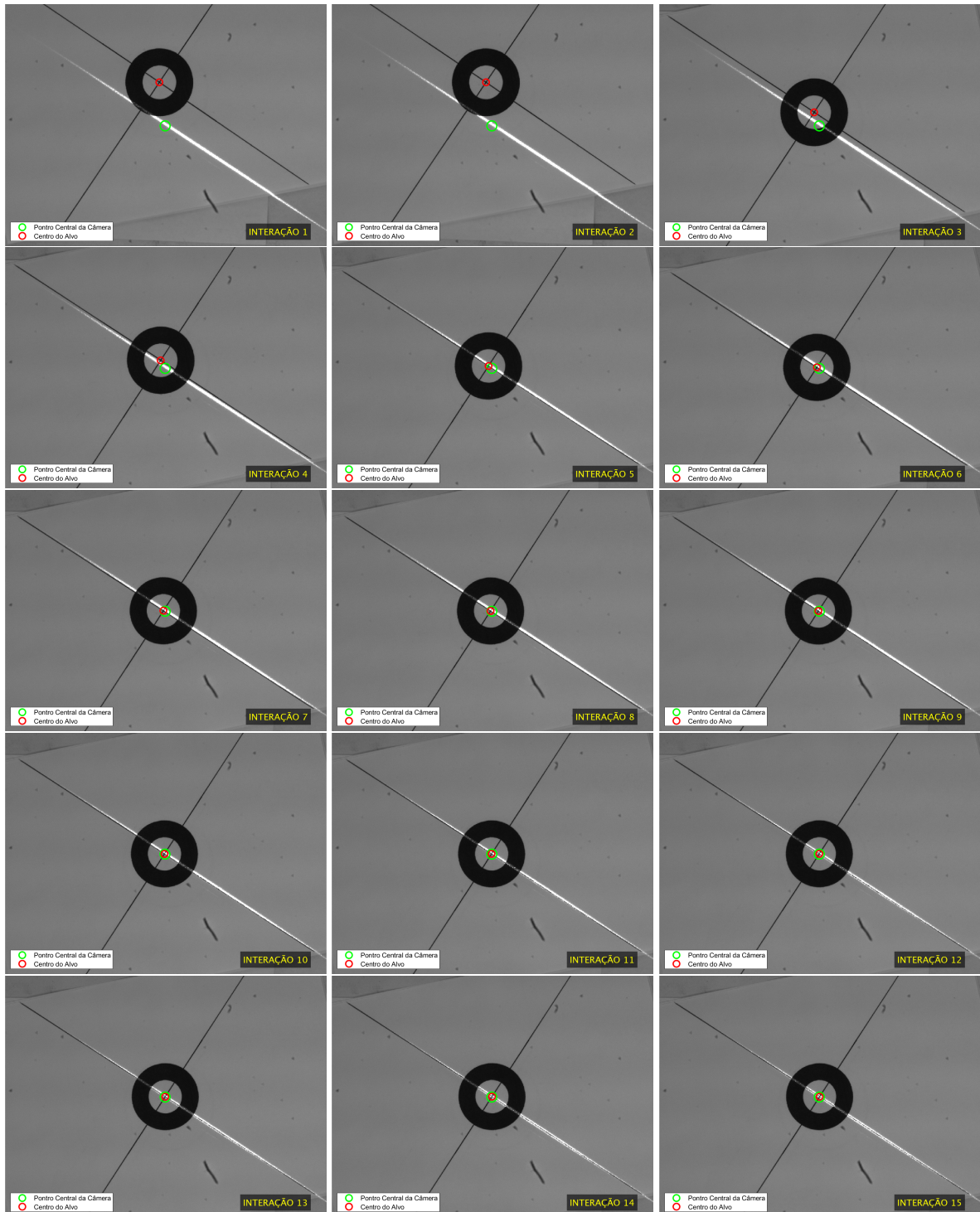


Figura 5.4 – Imagens capturadas pela câmera no decorrer do movimento - Experimento 1.

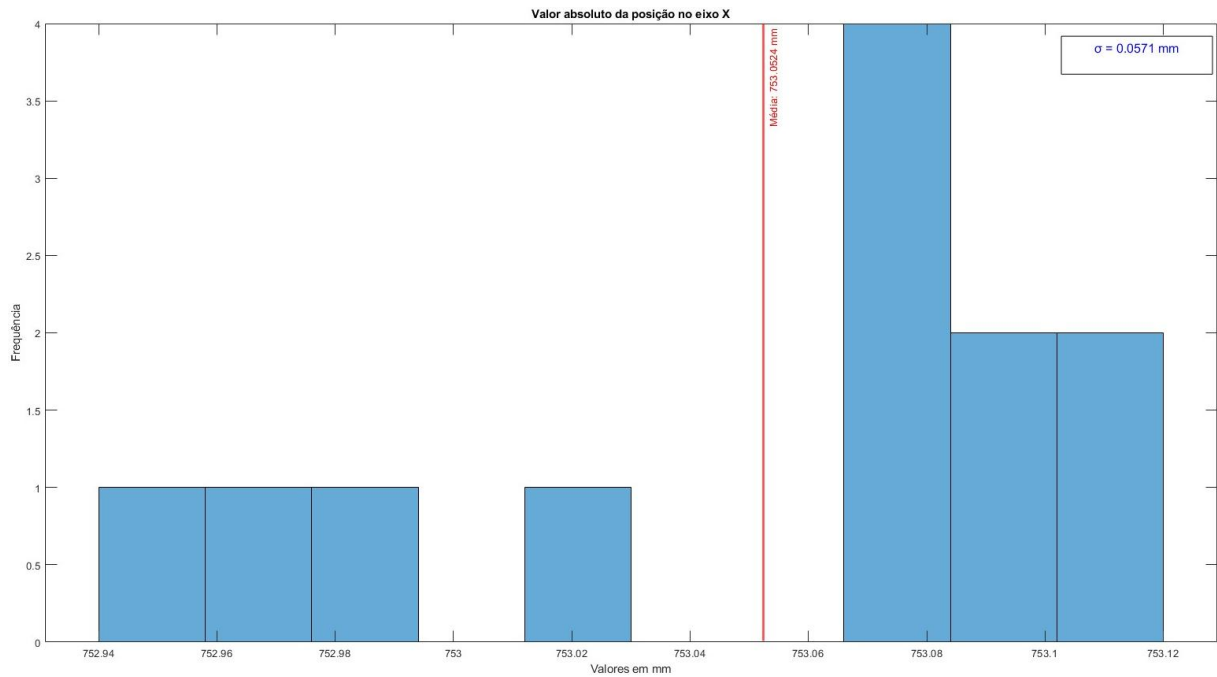


Figura 5.5 – Valor absoluto da posição no eixo x - Experimento 1.

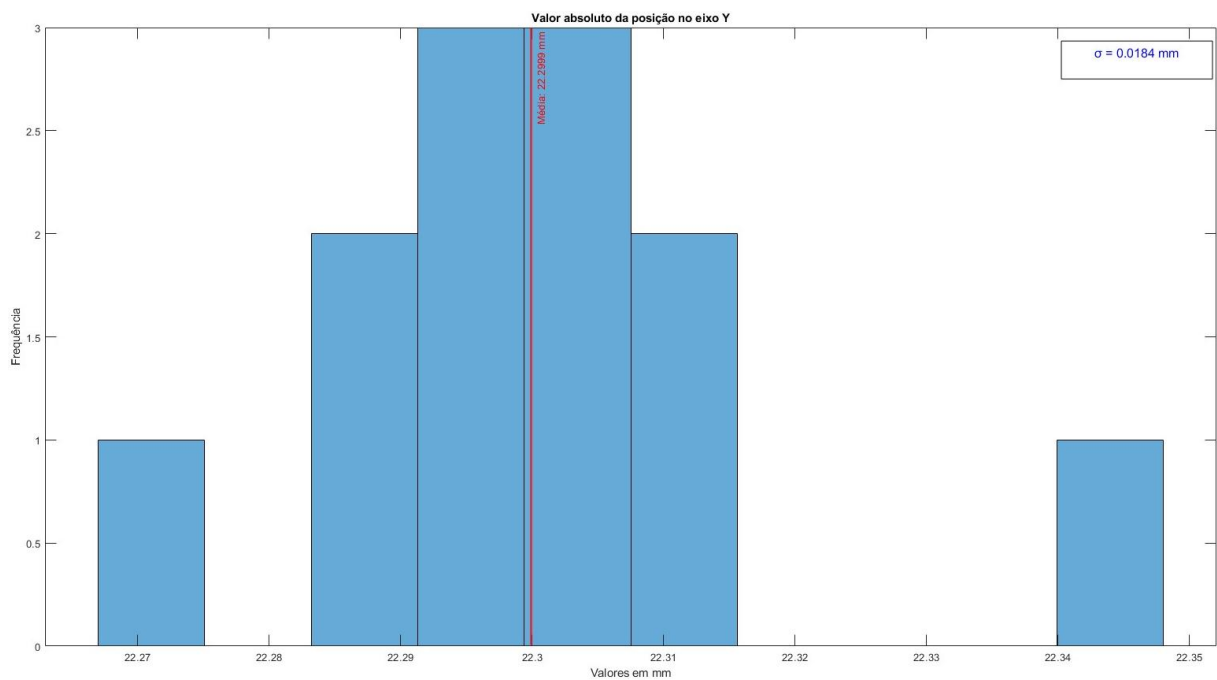


Figura 5.6 – Valor absoluto da posição no eixo y - Experimento 1.

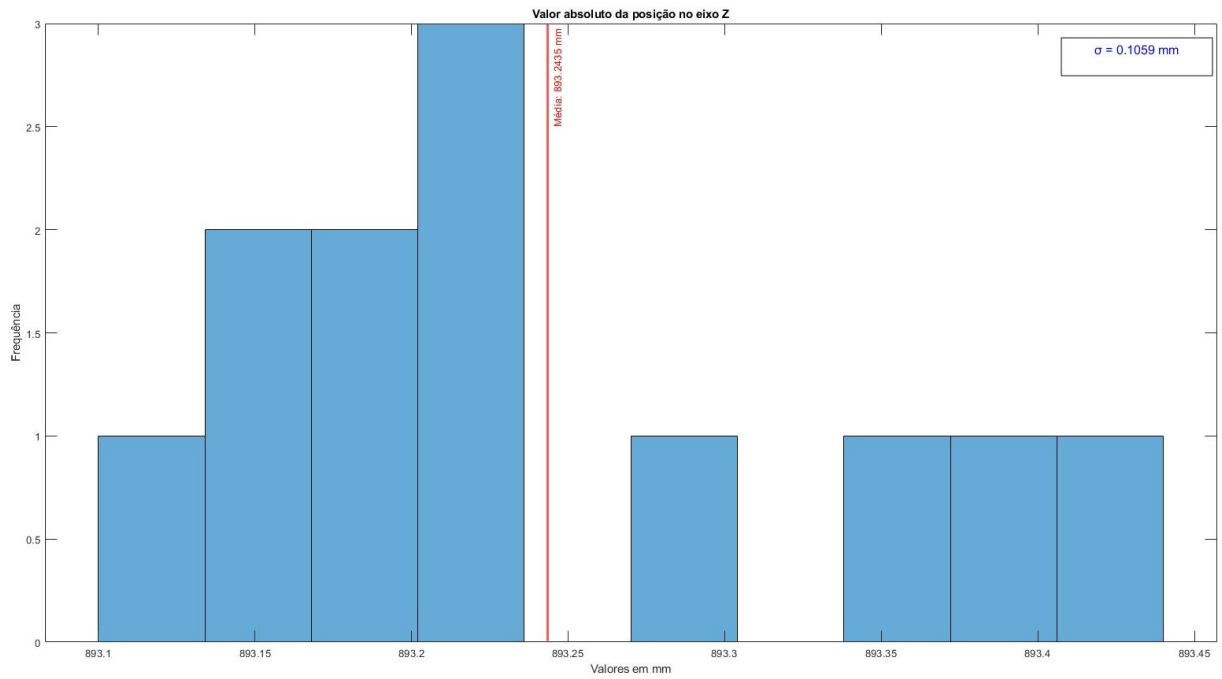


Figura 5.7 – Valor absoluto da posição no eixo z - Experimento 1.

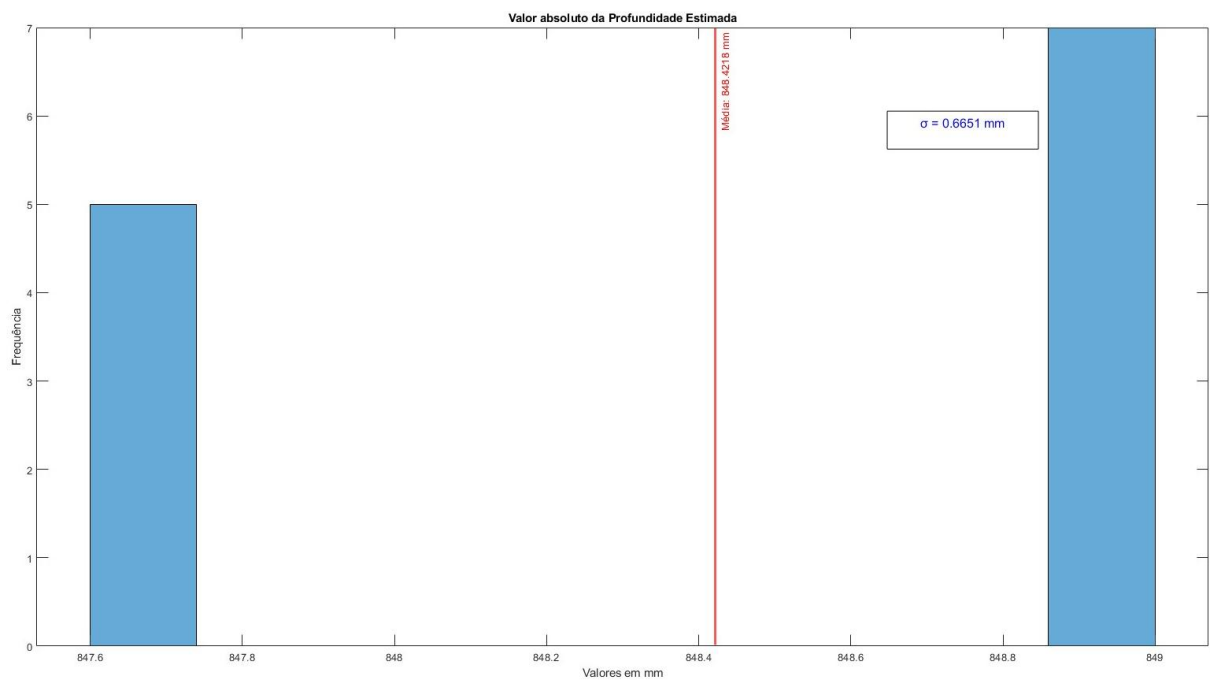


Figura 5.8 – Valor absoluto da Profundidade Estimada - Experimento 1.

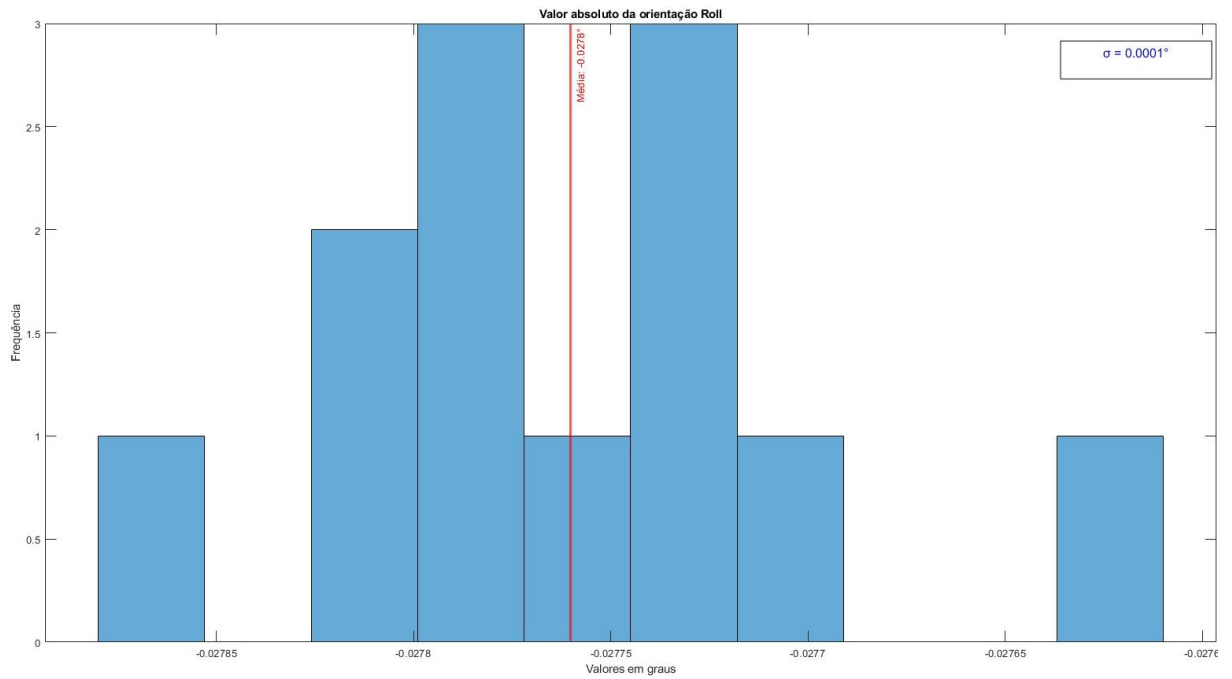


Figura 5.9 – Valor absoluto da orientação Roll - Experimento 1.

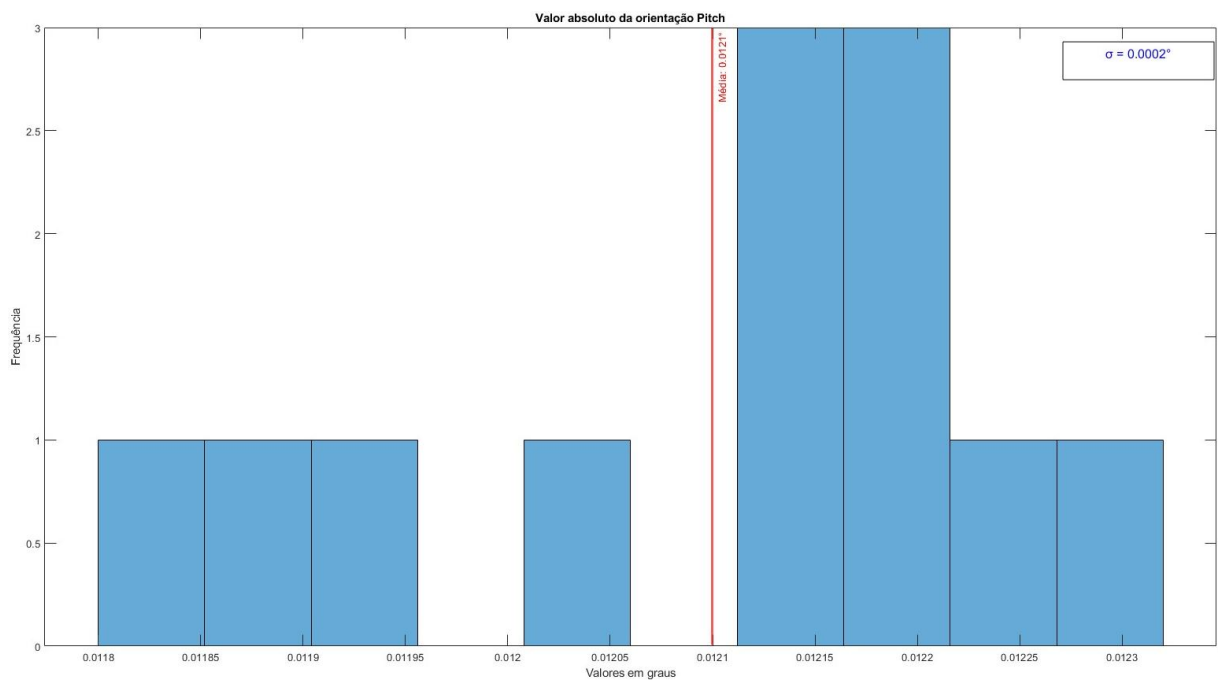


Figura 5.10 – Valor absoluto da orientação Pitch - Experimento 1.

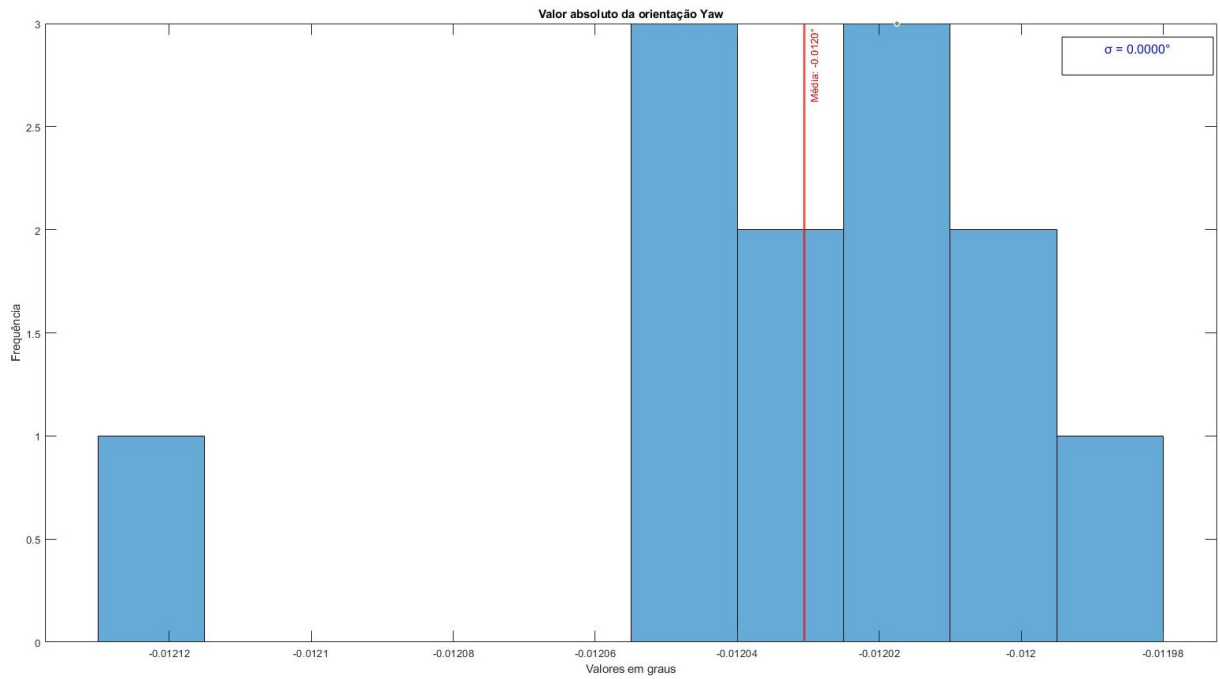


Figura 5.11 – Valor absoluto da orientação Yaw - Experimento 1.

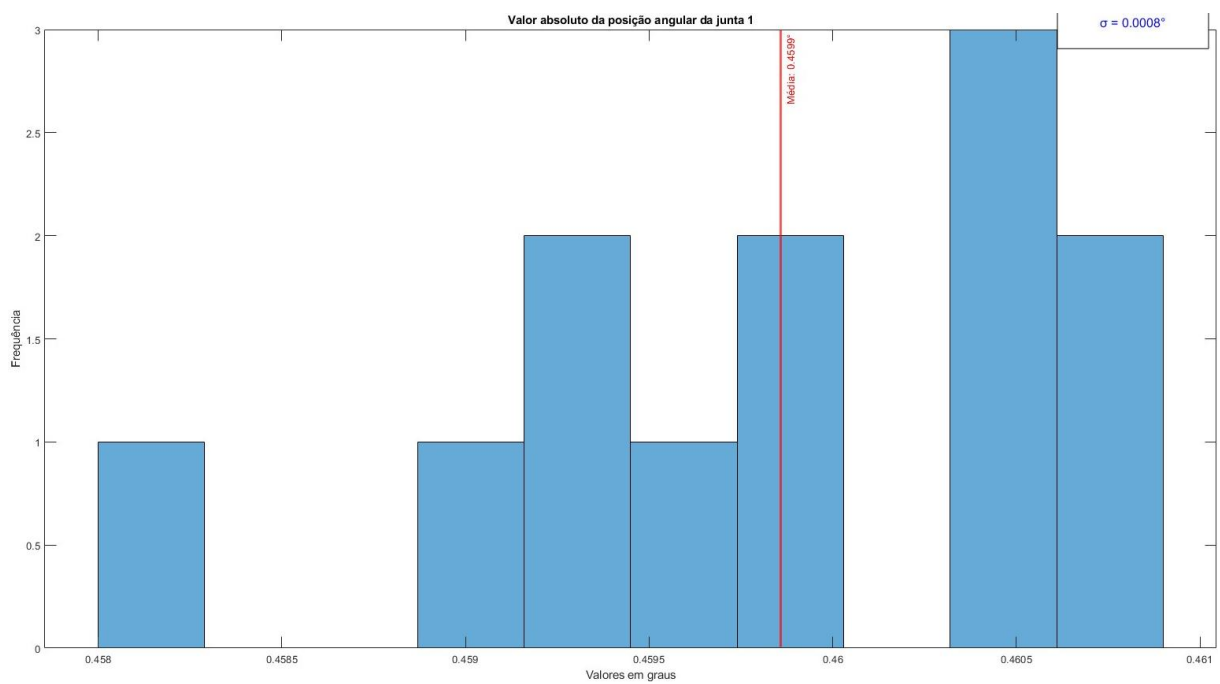


Figura 5.12 – Valor absoluto da rotação da Junta 1 - Experimento 1.

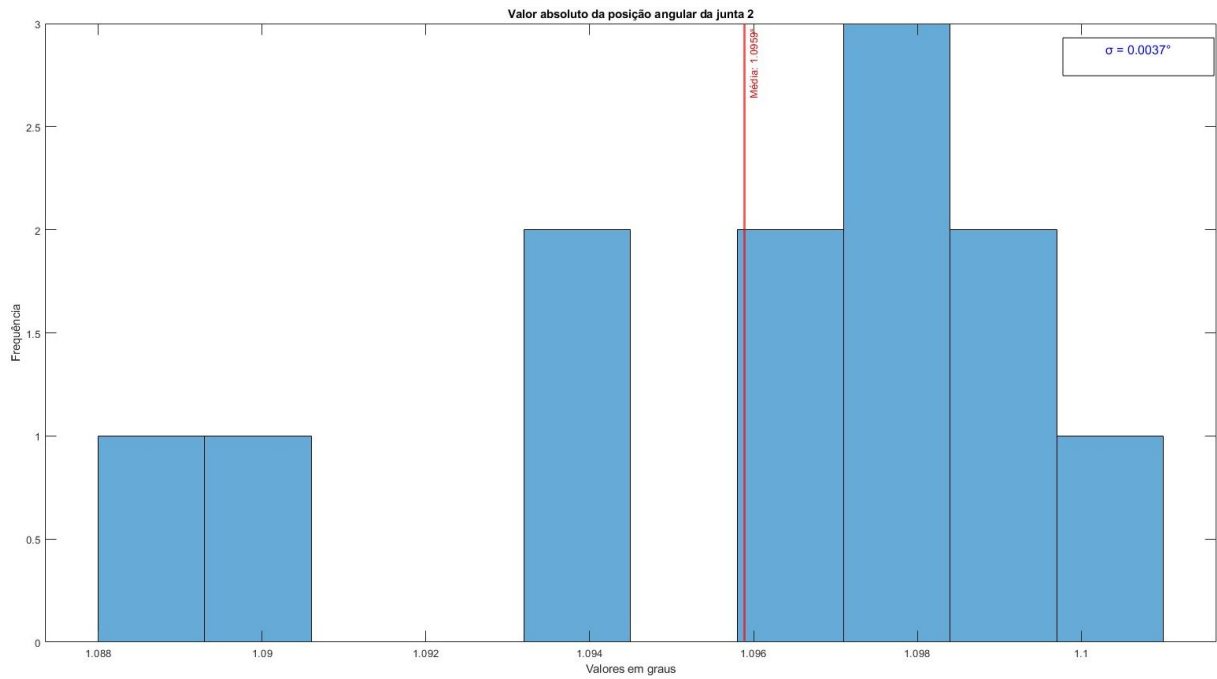


Figura 5.13 – Valor absoluto da rotação da Junta 2 - Experimento 1.

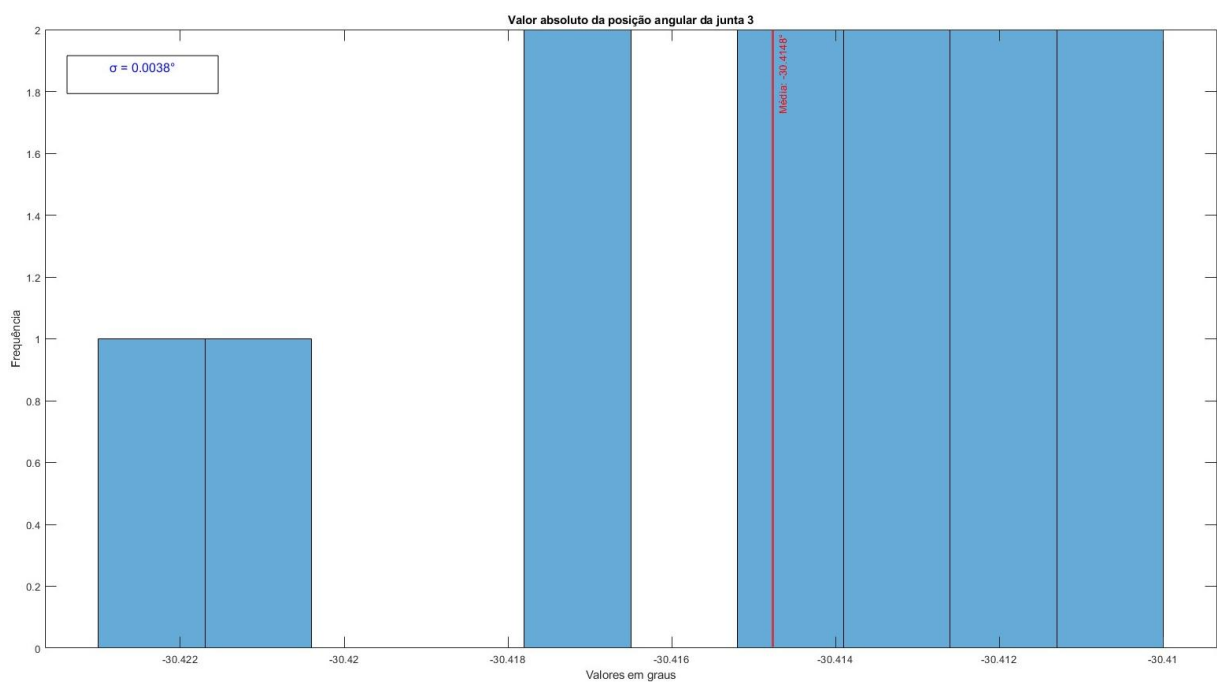


Figura 5.14 – Valor absoluto da rotação da Junta 3 - Experimento 1.

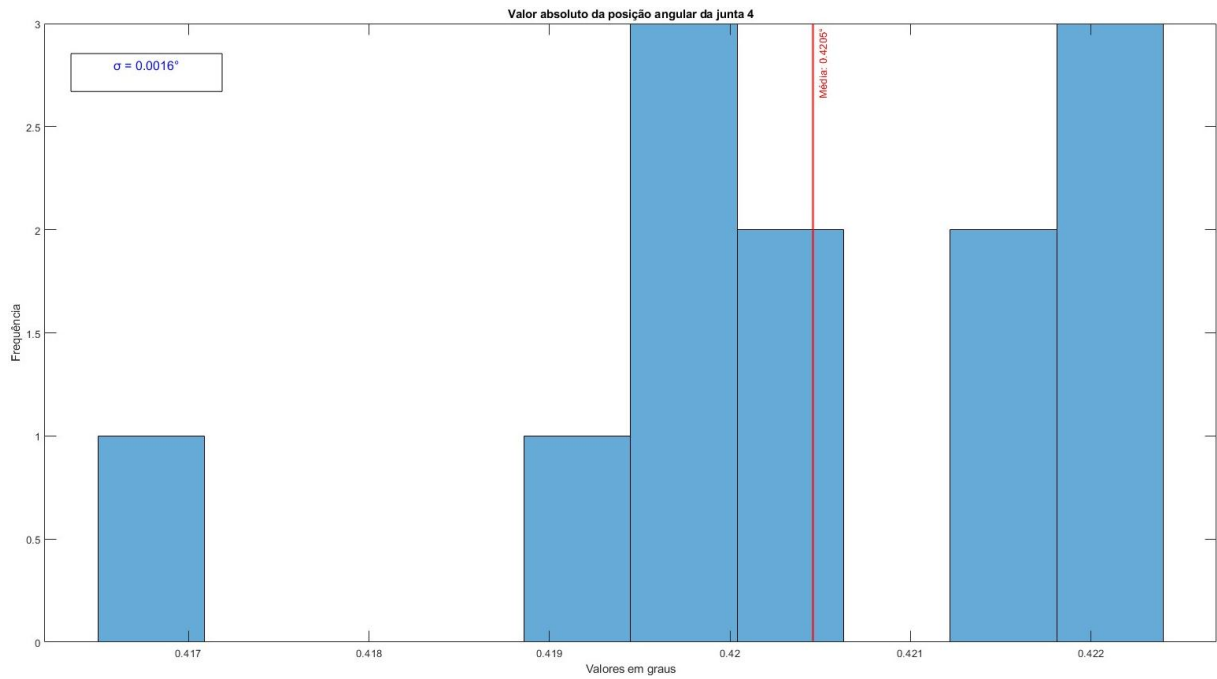


Figura 5.15 – Valor absoluto da rotação da Junta 4 - Experimento 1.

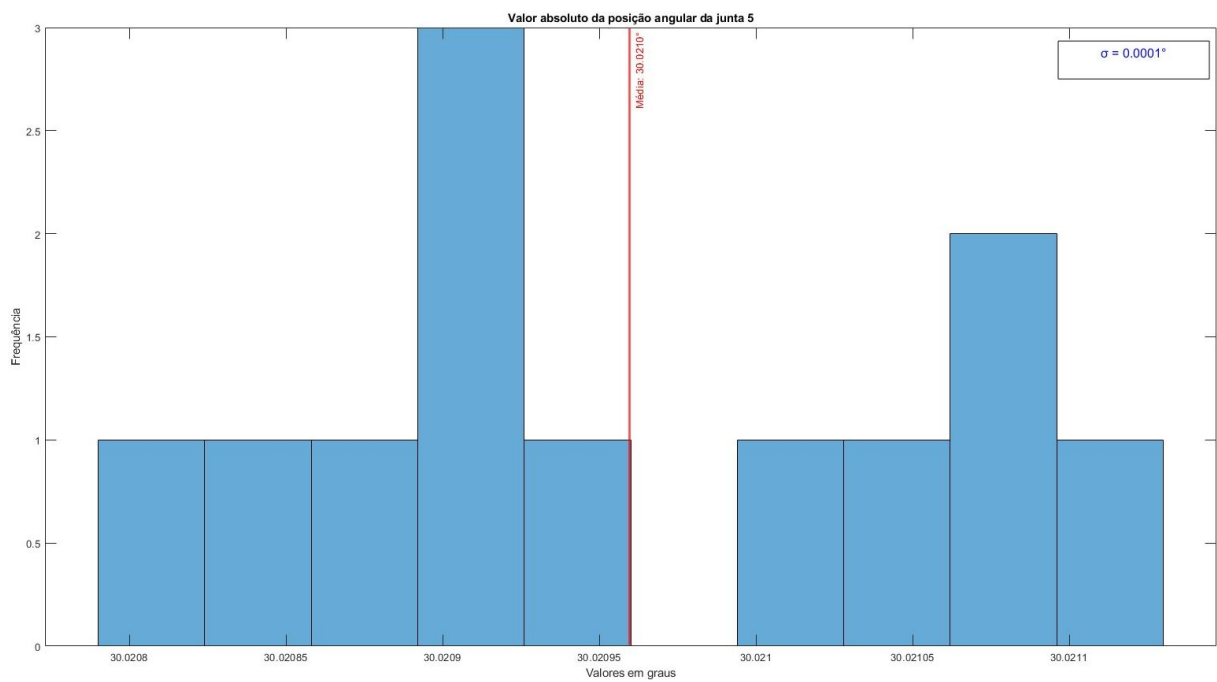


Figura 5.16 – Valor absoluto da rotação da Junta 5 - Experimento 1.

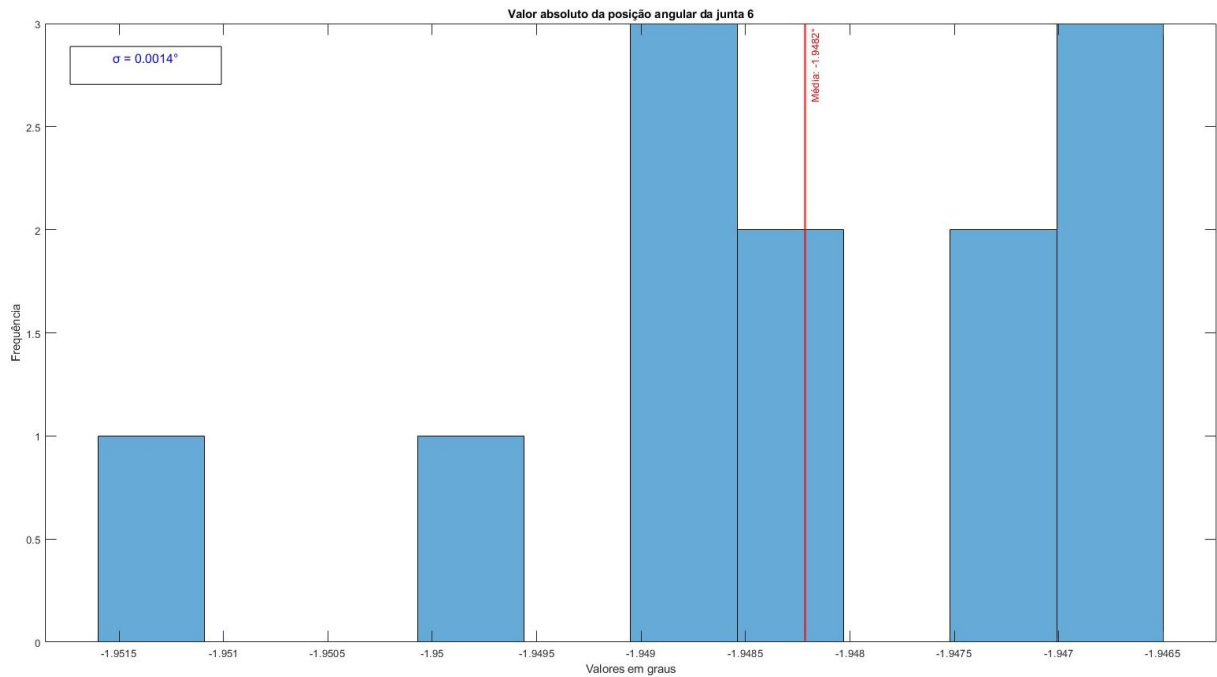


Figura 5.17 – Valor absoluto da rotação da Junta 6 - Experimento 1.

Tabela 5.3 – Erro entre as posições finais e iniciais da posição X.

Estatísticas	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-3} mm]	48,000	64,000	106,458	81,278	44,861
Maior Valor [10^{-3} mm]	103,500	120,000	224,583	262,833	134,583
Desvio Padrão [10^{-3} mm]	57,100	80,000	132,100	117,900	64,200

Tabela 5.4 – Erro entre as posições finais e iniciais da posição Y.

Estatísticas	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-3} mm]	12,150	15,333	19,583	10,256	128,333
Maior Valor [10^{-3} mm]	43,200	42,000	47,500	20,067	385,000
Desvio Padrão [10^{-3} mm]	18,400	21,300	28,400	32,300	187,000

Tabela 5.5 – Erro entre as posições finais e iniciais da posição Z.

Estatísticas	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-3} mm]	81,222	114,375	127,208	166,500	64,444
Maior Valor [10^{-3} mm]	181,333	236,250	254,417	536,500	193,333
Desvio Padrão [10^{-3} mm]	105,900	142,100	159,500	230,300	96,100

Tabela 5.6 – Erro entre as orientações finais e iniciais da orientação Roll.

	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-3} graus]	0,047	0,046	0,052	0,045	0,450
Maior Valor [10^{-3} graus]	0,137	0,158	0,121	0,096	1,350
Desvio Padrão [10^{-3} graus]	0,100	0,100	0,160	0,190	0,600

Tabela 5.7 – Erro entre as orientações finais e iniciais da orientação Pitch.

	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-3} graus]	0,121	0,165	0,225	0,229	0,0898
Maior Valor [10^{-3} graus]	0,273	0,310	0,450	0,742	0,270
Desvio Padrão [10^{-3} graus]	0,200	0,200	0,300	0,320	0,100

Tabela 5.8 – Erro entre as orientações finais e iniciais da orientação Yaw.

	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-3} graus]	0,0237	0,0256	0,0354	0,0240	0,250
Maior Valor [10^{-3} graus]	0,0912	0,0787	0,0596	0,0620	0,750
Desvio Padrão [10^{-3} graus]	0,0001	0,00013	0,0050	0,00020	0,300

Tabela 5.9 – Erro entre as posições finais e iniciais da Junta 1.

	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-6} graus]	636,389	651,667	1067,500	1005,420	1799,440
Maior Valor [10^{-6} graus]	1691,670	2068,330	3304,170	2897,500	5398,330
Desvio Padrão [10^{-6} graus]	800,000	900,000	1500,000	1600,000	2600,000

Tabela 5.10 – Erro entre as posições finais e iniciais da Junta 2.

	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-6} graus]	2888,889	4083,333	5000,000	4791,667	2277,778
Maior Valor [10^{-6} graus]	7258,333	7000,000	10000,000	15525,000	6166,667
Desvio Padrão [10^{-6} graus]	37,000	5100,000	5900,000	6900,000	3100,000

Tabela 5.11 – Erro entre as posições finais e iniciais da Junta 3.

	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-6} graus]	3033,333	4344,444	4937,500	5416,667	2333,333
Maior Valor [10^{-6} graus]	7475,000	7933,333	9875,000	17550,000	6250,000
Desvio Padrão [10^{-6} graus]	3800,000	5200,000	6200,000	7700,000	3100,000

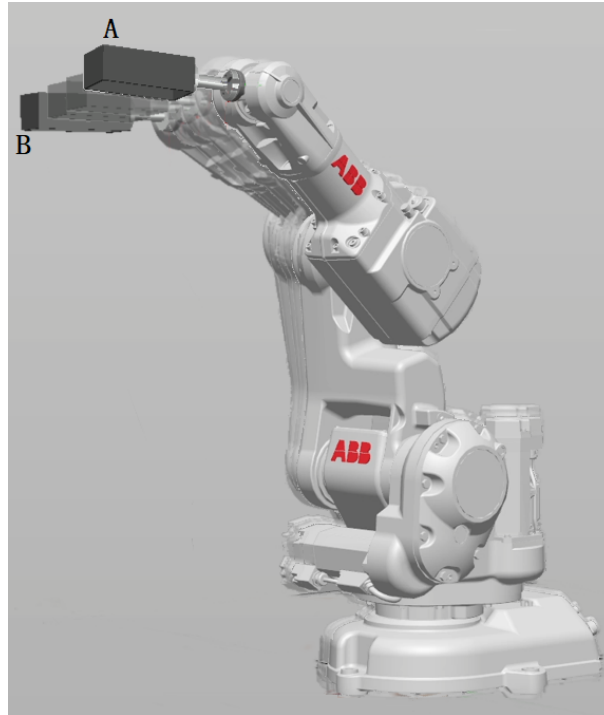


Figura 5.18 – Exemplo de trajetória do manipulador no espaço com posição e orientação inicial A para final B - 1ª Repetição do Experimento 4

Tabela 5.12 – Erro entre as posições finais e iniciais da Junta 4.

	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-6} graus]	1196,389	1312,500	1347,500	2291,667	3644,444
Maior Valor [10^{-6} graus]	3638,333	4375,000	4170,833	6050,000	10933,333
Desvio Padrão [10^{-6} graus]	1600,000	2000,000	1900,000	2900,000	5100,000

Tabela 5.13 – Erro entre as posições finais e iniciais da Junta 5.

	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-6} graus]	87,833	653,889	652,500	575,000	54,444
Maior Valor [10^{-6} graus]	153,000	1063,333	1812,500	1950,000	151,667
Desvio Padrão [10^{-6} graus]	100,000	800,000	900,000	800,000	100,000

Tabela 5.14 – Erro entre as posições finais e iniciais da Junta 6.

	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Média [10^{-6} graus]	1069,583	1180,000	729,167	2013,889	3150,000
Maior Valor [10^{-6} graus]	3102,500	3687,500	1475,000	5416,667	10150,000
Menor Valor [10^{-6} graus]	42,500	1600,000	900,000	2600,000	4500,000

5.4 Limitações do Método Experimental

Apesar dos resultados satisfatórios obtidos, o método experimental utilizado apresenta algumas limitações que podem impactar a precisão e a aplicabilidade do sistema. Essas limitações foram analisadas e possíveis soluções ou justificativas foram propostas, conforme descrito a seguir:

Iluminação da Cena - A iluminação da cena representa uma das principais limitações para sistemas de visão computacional. Alterações na intensidade ou direção da luz podem introduzir ruídos e dificultar a detecção do ponto gerado pelo laser. Para minimizar os efeitos da iluminação ambiente, o experimento foi realizado em um laboratório controlado, conforme descrito na metodologia. Adicionalmente, a implementação de filtros adaptativos ou algoritmos de compensação de iluminação, como o método descrito por [Chen, Zhang e Liu \(2018\)](#), pode ser considerada em aplicações futuras para ambientes não controlados.

Alvo Não Plano - O método de triangulação com laser utilizado depende de uma superfície plana para gerar uma linha no plano do objeto. Em alvos tridimensionais, a profundidade pode não ser estimada corretamente. Uma solução possível para lidar com alvos tridimensionais seria a utilização de um sistema de visão estéreo com duas câmeras. Esse método é amplamente descrito na literatura, como em [Hartley e Zisserman \(2004\)](#) [Hartley e Zisserman \(2004\)](#), e permite estimar a profundidade de alvos complexos ao combinar informações obtidas de diferentes perspectivas. Alternativamente, sensores de profundidade ativos, como os baseados em luz estruturada, podem ser empregados para aumentar a precisão em cenários tridimensionais.

Imagem Fora do Campo de Visão da Câmera - Outra limitação ocorre quando o ponto de interesse se encontra fora do campo de visão da câmera. Isso inviabiliza a detecção e, conseqüentemente, o controle do manipulador. Uma solução proposta para essa limitação é o desenvolvimento de um algoritmo de busca que permita reposicionar o manipulador de forma a trazer o ponto de interesse para o campo de visão da câmera. Essa proposta será discutida com mais detalhes no [seção 6.2](#) deste trabalho.

Nitidez da Imagem - A nitidez da imagem é garantida apenas dentro de uma faixa de distâncias da câmera em relação ao alvo. Quando o objeto está muito próximo ou muito distante, a qualidade da imagem é prejudicada, dificultando a detecção precisa do ponto. Para mitigar essa limitação, o robô foi configurado para operar apenas dentro do volume de trabalho onde a nitidez da imagem é garantida. Adicionalmente, o uso de lentes com foco ajustável, como discutido em trabalhos como o de [Cui, Chen e Wang \(2015\)](#), pode ser explorado em futuras implementações para aumentar a flexibilidade do sistema.

Outras Limitações Comuns em Sistemas IBVS - Entre as limitações gerais de sistemas de visão baseados em servo visual (IBVS), destaca-se a sensibilidade a erros de calibração da câmera, que podem impactar a precisão do controle. Garantir uma calibração

precisa da câmera, utilizando métodos como o de [Zhang \(2000\)](#), e revisar periodicamente os parâmetros do modelo para manter a confiabilidade do sistema. Adicionalmente, o uso de redundância de sensores pode melhorar a robustez do sistema em aplicações críticas.

As limitações mencionadas representam desafios comuns em sistemas de visão computacional e servo visual. Contudo, as soluções propostas, tanto implementadas quanto sugeridas, demonstram a viabilidade de superar tais desafios e ampliar a aplicabilidade do método desenvolvido.

5.5 Análise Geral

Os experimentos realizados forneceram dados cruciais sobre a capacidade do sistema de visão desenvolvido em manter a precisão ao longo das repetições, especialmente em relação à distância entre a câmera e o alvo, que foi determinada com base no alinhamento do ponto central do alvo com o ponto principal da câmera. A análise dos dados coletados demonstrou que o robô é capaz de alcançar o centro do alvo com um desvio mínimo, conforme evidenciado pelos valores de desvio padrão calculados, que foram muito pequenos para a maioria das medições. Esses resultados indicam que, mesmo com as variações nos movimentos do robô, o sistema manteve uma consistência notável, refletindo uma alta repetibilidade.

No que diz respeito às orientações do manipulador (Roll, Pitch e Yaw), os erros médios foram consistentemente baixos, com valores na ordem de 10^{-3} graus, indicando que o sistema não apenas mantém o posicionamento preciso, mas também a estabilidade angular necessária.

É importante destacar que dados experimentais consideraram a influência do erro de repetibilidade do robô de 0,03 mm especificado pelo fabricante do manipulador ABB IRB 140 ([ABB Robotics, 2020](#)). Esse erro de medição, inerente ao sistema, introduz uma contaminação de fundo nas leituras registradas pelo controlador desse manipulador, cujos dados foram extraídos e processados através do RAPID. Embora pequeno, esse erro pode impactar a precisão dos resultados, especialmente em aplicações de alta sensibilidade, como as descritas neste trabalho. Portanto, os desvios reportados devem ser interpretados levando em conta essa limitação, uma vez que ela reflete um ruído sistemático presente em todas as medições realizadas.

Insights sobre Repetibilidade e Robustez - Os dados coletados também destacam a capacidade do sistema de minimizar a variação entre as repetições, demonstrando sua robustez em diferentes cenários experimentais. Essa repetibilidade é fundamental para a calibração de robôs e para a execução de tarefas que requerem movimentos contínuos e consistentes, como montagem de componentes sensíveis ou inspeção visual de alta precisão.

A precisão do sistema de visão desenvolvido foi validada ao se garantir que o feixe de laser passasse pelo ponto central do alvo, mantendo uma profundidade constante Z

durante as interações. O alinhamento do ponto central da câmera e do alvo, combinado com a profundidade constante, foi essencial para assegurar medições consistentes em cada repetição, livres de grandes variações.

Os resultados obtidos durante os experimentos oferecem insights valiosos para o processo de calibração de manipuladores robóticos. A repetibilidade da profundidade Z , que foi consistentemente mantida em todos os experimentos, proporciona uma base sólida para ajustes nos modelos cinemáticos do robô. Esses ajustes podem corrigir discrepâncias entre as posições previstas e as posições alcançadas pelo robô, considerando variações em fatores como offsets das juntas, seus comprimentos ou até características específicas do sistema de visão utilizado.

A calibração pode ser especialmente útil em sistemas onde o robô precisa ser ajustado com precisão para interagir com alvos em diferentes cenários industriais. O alinhamento preciso entre o ponto central da câmera e o alvo, validado pelo feixe de laser, pode ser aplicado para melhorar a precisão em tarefas de manipulação, como montagem ou inspeção, onde a precisão é crítica.

De acordo com as hipóteses formuladas no início deste trabalho, os resultados experimentais confirmaram que a instalação do sensor de visão acoplado ao efetuador do robô não apenas melhorou o direcionamento do manipulador, como também demonstrou a viabilidade do controle servo-visual. A obtenção da distância entre o sensor de visão e o alvo, utilizando o algoritmo de triangulação (seção 4.2) para estimar a profundidade da imagem em relação ao centro óptico da câmera, mostrou-se eficaz em manter uma distância constante nos movimentos do robô.

Esses resultados refletem o atendimento pleno aos requisitos de alta precisão e repetibilidade, destacados nos objetivos gerais e específicos do trabalho. O sensor de visão ativa desenvolvido, integrado ao manipulador robótico, mostrou-se eficiente em realizar medições em tempo real e corrigir erros de posicionamento, validando sua viabilidade e robustez para aplicações industriais dinâmicas. Além disso, os algoritmos implementados permitiram um aumento expressivo na eficiência do sistema, excedendo significativamente a estimativa inicial de 30% de melhoria, demonstrando a eficácia dos métodos propostos para calibração e controle.

Portanto, as metas estabelecidas no início do trabalho, incluindo a garantia de alta exatidão, repetibilidade e custo reduzido, foram atendidas com sucesso. Esses resultados fornecem uma base sólida para avanços futuros, como o aumento da quantidade de pontos característicos e a implementação de técnicas complementares para lidar com desafios relacionados ao campo de visão e à nitidez.

Além disso, o desenvolvimento de algoritmos avançados que geram as velocidades e orientações necessárias para que o robô alcance o alvo de forma precisa e repetível, demonstrou que o controle servo-visual baseado em imagem é uma ferramenta eficaz para o

controle de manipuladores robóticos, especialmente em ambientes industriais dinâmicos.

Os resultados do experimento reafirmaram as hipóteses e os objetivos formulados no início do trabalho. A precisão nas medições e a repetibilidade observada mostram que o controle servo-visual é uma abordagem promissora para manipuladores robóticos com seis graus de liberdade, como o ABB IRB 140. A integração de visão computacional no controle do manipulador proporciona uma solução robusta para tarefas industriais, onde a necessidade de precisão e flexibilidade é essencial. O sistema demonstrou alta capacidade de adaptação, permitindo ajustes rápidos e precisos, o que é vantajoso para ambientes industriais dinâmicos.

A realização dos experimentos enfrentou diversos desafios, especialmente relacionados às limitações impostas pela linguagem de programação *RAPID* e às restrições mecânicas intrínsecas do manipulador IRB140. Um dos principais obstáculos foi o desencontro entre os dados gerados pelo sensor de visão e os comandos de movimentação robótica. A linguagem *RAPID* permite que o manipulador execute apenas movimentos baseados em posições predefinidas, restringindo a implementação de comandos diretos de velocidade.

Idealmente, o manipulador deveria aceitar entradas de velocidades para maior fluidez no controle servo-visual. Para contornar essa limitação, foi necessário desenvolver uma metodologia que convertesse as velocidades calculadas pelo sensor de visão em posições para todas as seis juntas do robô. Essas posições foram ajustadas simultaneamente em intervalos de tempo pré-definidos, permitindo aproximar o comportamento desejado.

Outro fator desafiador, comum a sistemas baseados em IBVS, é a dependência do alvo estar dentro do campo de visão da câmera. Caso o alvo saia do campo de visão, o robô é incapaz de calcular os comandos necessários para corrigir sua posição, interrompendo qualquer tentativa de realinhamento ou movimentação.

Além disso, a aproximação ou afastamento excessivo do alvo frequentemente resultou em perda de nitidez na imagem, dificultando a detecção precisa dos pontos característicos pelo sistema de visão. Esse problema foi intensificado pela limitação da abertura da lente, especificada na [tabela 4.2](#), permanecer fixa durante a execução dos experimentos, uma vez que o sistema não possui controle automático desse parâmetro. Como consequência, ajustes precisos da nitidez tornaram-se inviáveis durante o movimento, comprometendo temporariamente a estabilidade do algoritmo IBVS em certas condições. Outro desafio significativo foi a variação das condições de iluminação no ambiente experimental, que em alguns momentos prejudicou a identificação confiável dos pontos característicos do alvo. Tais variações exigiram ajustes manuais no sistema de iluminação e na calibração dos parâmetros da câmera para garantir o desempenho esperado.

6 CONCLUSÕES E PERSPECTIVAS DE TRABALHOS FUTUROS

Este trabalho teve como objetivo principal avaliar a repetibilidade do sistema de visão utilizado no controle do manipulador robótico ABB IRB 140, destacando sua precisão ao atingir um alvo definido no espaço. O estudo empregou o controle servo-visual baseado em imagem (IBVS), focando na capacidade do sistema de garantir a consistência necessária para aplicações de alta precisão.

6.1 Conclusões

Em síntese, este trabalho demonstrou que o manipulador robótico ABB IRB 140, utilizando o sistema de visão integrado, é capaz de atingir o alvo com alta precisão e repetibilidade. A calibração realizada, com base nos resultados obtidos, poderá ser aplicada para aprimorar o desempenho do manipulador em tarefas específicas, garantindo maior confiabilidade e robustez no contexto industrial. A abordagem proposta, centrada na visão computacional e controle servo-visual, representa um avanço significativo para o desenvolvimento de sistemas de controle de manipuladores robóticos, destacando-se como um sistema de medição para abordagens de calibração de robôs com utilização de um sensor de baixo custo e com técnicas de calibração em diversas pequenas regiões em um único processo de calibração.

A análise da repetibilidade das medições realizadas com o método proposto revelou resultados promissores. O valor médio da repetibilidade obtido foi de 0,037 mm, indicando um alto nível de consistência nas posições finais do manipulador em relação ao alvo. O maior valor registrado de repetibilidade foi de 0,06 mm, permanecendo dentro de limites aceitáveis para aplicações de precisão. Além disso, o desvio padrão das medições foi calculado como 0,012 mm, reforçando a estabilidade do método em diferentes execuções. Esses valores foram determinados a partir de uma análise estatística dos dados experimentais, considerando a média aritmética dos desvios absolutos em relação à posição de referência, o maior desvio registrado e a dispersão dos dados (desvio padrão). Tais resultados evidenciam a eficácia do método proposto em minimizar os erros de posicionamento final, atendendo aos critérios de precisão definidos para o sistema.

A solução proposta neste trabalho demonstrou um impacto econômico significativo em comparação com tecnologias tradicionais de calibração de robôs industriais. O sensor foi desenvolvido utilizando componentes acessíveis, como uma câmera de foco fixo (16 mm), um laser e ferramentas computacionais amplamente disponíveis, como o MATLAB, eliminando a necessidade de sistemas de calibração proprietários de alto custo. Atualmente, os calibradores

industriais de robôs disponíveis no mercado, como teodolitos e laser trackers, oferecem alta precisão de medição tanto em curtas quanto em longas distâncias. No entanto, esses equipamentos podem custar até cerca de U\$ 80.000 dólares, dependendo da complexidade e precisão desejada. Em contraste, a solução desenvolvida apresenta um custo estimado inferior a R\$ 5.000,00, representando uma economia de até 98,75%. Além do custo reduzido, o método proposto também se destacou pelo impacto operacional, possibilitando uma calibração ágil e flexível, especialmente em ambientes controlados, como laboratórios acadêmicos. Esses resultados reforçam o potencial da solução em democratizar o acesso à calibração de robôs industriais, especialmente para pequenas e médias empresas ou para aplicações experimentais em instituições de ensino e pesquisa.

Com base nesses resultados, é possível afirmar que a precisão e repetibilidade alcançadas neste estudo podem ser aplicadas diretamente para calibrar e melhorar os manipuladores robóticos em ambientes industriais, além de contribuir para a evolução do controle servo-visual em robótica.

6.2 Perspectivas de Trabalhos Futuros

O sensor é altamente versátil e pode ser empregado na calibração de robôs, proporcionando a oportunidade de criar novas programações de posicionamento no ambiente de trabalho. Isso pode ser feito tanto por meio da calibração atualizada do robô quanto através do IVBS para identificação de alvos específicos.

Um trabalho interessante para esse sensor seria a criação de um banco de dados para o jacobiano da imagem. A análise da equação do jacobiano da imagem é baseada em um ponto de pixel. Esse valor é inalterável. A ideia seria criar um banco de dados para diminuir o tempo e energia de processamento. Ao invés de fazer esse cálculo inúmeras vezes o algoritmo buscaria no banco de dados esse valor previamente calculado com endereço característico ao pixel.

Uma iniciativa adicional para aprimorar o IVBS seria desenvolver um algoritmo de busca para o manipulador, ativado sempre que o alvo não estivesse visível no campo de visão da câmera.

Uma tarefa desafiadora seria aplicar o modelo proposto nesta dissertação para permitir que o robô localize e interaja com alvos móveis. O principal desafio residiria na sincronização precisa do movimento do robô com o deslocamento do alvo, além da necessidade de operar simultaneamente com a imagem capturada pela câmera, garantindo que o processamento seja realizado dentro de um tempo mínimo compatível com a execução dos movimentos.

Referências

- ABB Robotics. **Product Specification Manual - IRB 140**. Västerås, Sweden, 2020. 36 p. Document ID: 3HAC041346-001 Revision: Q. Disponível em: <https://www.abb.com/robotics>. Citado na p. 99.
- ABDERRAHIM, M.; KHAMIS, A.; GARRIDO, S.; MORENO, L. Accuracy and calibration issues of industrial manipulators. **Industrial robotics: programming, simulation and application**, I-Tech Madrid, Spain, p. 131–146, 2004. Citado na p. 23.
- AI Magazine. Machine vision enabling the factories of the future. **AI Magazine**, 2022. Acessado em: 07 out. 2024. Disponível em: <https://www.aaai.org/ocs/index.php/aimagazine/article/view/11640>. Citado na p. 41.
- AJEWOLE, F.; KELKAR, A.; MOORE, D.; SHAO, E.; THIRTHA, M. Unlocking the industrial potential of robotics and automation. **McKinsey & Company**, 2023. Disponível em: <https://www.mckinsey.com/industries/industrials-and-electronics/our-insights/unlocking-the-industrial-potential-of-robotics-and-automation>. Citado na p. 22.
- AKINLAR, C.; TOPAL, C. Edcircles: A real-time circle detector with a false detection control. **Pattern Recognition**, Elsevier, v. 46, n. 3, p. 725–740, 2013. Citado na p. 57.
- ANGELINI, P.; PALMIERI, G.; ANGELETTI, S. *et al.* Systematic review on calibration methods for industrial robots. **Journal of Manufacturing Systems**, v. 61, p. 255–269, 2021. Citado na p. 69.
- ARIATEJA, D.; ARDIYANTO, I.; SOESANTI, I. A review of contrast enhancement techniques in digital image processing. *In: IEEE. 2018 4th International Conference on Science and Technology (ICST)*. [S.l.], 2018. p. 1–6. Citado na p. 38.
- AZIZIAN, M.; KHOSHNAM, M.; NAJMAEI, N.; PATEL, R. V. Visual servoing in medical robotics: a survey. part i: endoscopic and direct vision imaging – techniques and applications. **The International Journal of Medical Robotics and Computer Assisted Surgery**, v. 10, n. 3, p. 263–274, 2014. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/rcs.1531>. Citado nas pp. 33, 34 e 35.
- BENDER, D.; SCHIKORA, M.; STURM, J.; CREMERS, D. Graph-based bundle adjustment for ins-camera calibration. *In: Unmanned Aerial Vehicle in Geomatics (UAV-g)*. Rostock, Germany: [s.n.], 2013. p. 4–6. Citado na p. 70.
- BERNARD, A.; MANGIN, A. Calibration of robot kinematics. **Mechanics Based Design of Structures and Machines**, v. 32, n. 3, p. 393–412, 2004. Citado na p. 70.
- BIBER, P.; ANDREASSON, H.; DUCKETT, T.; SCHILLING, A. 3d modeling of indoor environments by a mobile robot with a laser scanner and panoramic camera. *In: IEEE*.

- 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**(IEEE Cat. No. 04CH37566). [S.l.], 2004. v. 4, p. 3430–3435. Citado na p. 20.
- BLOSS, R. Collaborative robots are rapidly providing major improvements in productivity, safety, programming ease, portability and cost while addressing many new applications. **Industrial Robot: An International Journal**, Emerald Group Publishing Limited, v. 43, n. 5, p. 463–468, 2016. Citado na p. 21.
- BROWN, S.; BESSANT, J. The manufacturing strategy-capabilities links in mass customisation and agile manufacturing—an exploratory study. **International Journal of Operations & Production Management**, MCB UP Ltd, v. 23, n. 7, p. 707–730, 2003. Citado nas pp. 19 e 22.
- CANNY, J. A computational approach to edge detection. **IEEE Transactions on pattern analysis and machine intelligence**, Ieee, n. 6, p. 679–698, 1986. Citado na p. 52.
- CERIANI, N. M.; ZANCHETTIN, A. M.; ROCCO, P.; STOLT, A.; ROBERTSSON, A. Reactive task adaptation based on hierarchical constraints classification for safe industrial robots. **IEEE/ASME Transactions on Mechatronics**, IEEE, v. 20, n. 6, p. 2935–2949, 2015. Citado na p. 21.
- CHAUMETTE, F. Potential problems of stability and convergence in image-based and position-based visual servoing. *In: The confluence of vision and control*. [S.l.]: Springer, 2007. p. 66–78. Citado nas pp. 57 e 58.
- CHEAH, C.-C.; HIRANO, M.; KAWAMURA, S.; ARIMOTO, S. Approximate jacobian control for robots with uncertain kinematics and dynamics. **IEEE transactions on robotics and automation**, IEEE, v. 19, n. 4, p. 692–702, 2003. Citado nas pp. 33, 34 e 35.
- CHEAH, C.-C.; LIU, C.; SLOTINE, J.-J. E.; . Adaptive tracking control for robots with unknown kinematic and dynamic properties. **The International Journal of Robotics Research**, SAGE Publications, v. 25, n. 3, p. 283–296, 2006. Citado nas pp. 32, 33, 34 e 35.
- CHEN, F.; BROWN, G. M.; SONG, M. Overview of 3-d shape measurement using optical methods. **Optical engineering**, SPIE, v. 39, n. 1, p. 10–22, 2000. Citado na p. 20.
- CHEN, T.; HUANG, Z.; MOTES, J.; GENG, J.; TA, Q. M.; DINKEL, H.; ABDUL-RASHID, H.; MYERS, J.; MUN, Y.-J.; LIN, W.-c. *et al.* Insights from an industrial collaborative assembly project: Lessons in research and collaboration. **arXiv preprint arXiv:2205.14340**, 2022. Disponível em: <https://arxiv.org/abs/2205.14340>. Citado na p. 22.
- CHEN, X.; SMITH, R. An overview of vision-based robot calibration techniques. **Journal of Robotic Systems**, v. 17, n. 6, p. 345–356, 2000. Citado na p. 23.

- CHEN, Y.; ZHANG, P.; LIU, X. Illumination compensation for robust object detection under varying lighting conditions. **IEEE Transactions on Image Processing**, IEEE, v. 27, n. 10, p. 4852–4866, 2018. Citado na p. 98.
- COPPE. **Aquisição e digitalização de imagem em câmera CCD**. 2024. Acesso em: janeiro de 2024. URL não disponível atualmente. Disponível em: <http://www.materia.coppe.ufrj.br/sarra/artigos/artigo10297/>. Citado nas pp. 9 e 39.
- CORKE, P. **Robotics, vision and control: fundamental algorithms in MATLAB®**. Londres: Springer, 2017. Citado na p. 60.
- CUI, J.; CHEN, X.; WANG, Y. Adaptive focus control for high-resolution images in industrial vision systems. **Optics and Lasers in Engineering**, Elsevier, v. 69, p. 47–54, 2015. Citado na p. 98.
- DOMBRE, E.; KHALIL, W. **Modeling, Identification and Control of Robots**. [S.l.]: Butterworth-Heinemann, 2007. Citado na p. 23.
- DOMBRE, E.; KHALIL, W. Modeling, performance analysis and control of robot manipulators. **(No Title)**, 2007. Citado na p. 23.
- DRIELS, M. R.; PATHRE, R. V. Using robot calibration to improve the accuracy of point-to-point robotic motion. **Journal of Robotic Systems**, v. 8, n. 4, p. 519–543, 1991. Citado na p. 69.
- DUDA, R. O.; HART, P. E. Use of the hough transformation to detect lines and curves in pictures. **Communications of the ACM**, ACM New York, NY, USA, v. 15, n. 1, p. 11–15, 1972. Citado na p. 55.
- EVERETT, H. R. *et al.* Model-based error correction in robotic manipulator trajectory following. **The International Journal of Robotics Research**, v. 6, n. 4, p. 46–56, 1987. Citado na p. 70.
- FORSMAN, P. *et al.* **Three-dimensional localization and mapping of static environments by means of mobile perception**. [S.l.]: Helsinki University of Technology, 2001. Citado nas pp. 73 e 76.
- FRY, J.; PUSATERI, M. A system and method for auto-correction of first order lens distortion. *In*: IEEE. **2010 IEEE 39th Applied Imagery Pattern Recognition Workshop (AIPR)**. [S.l.], 2010. p. 1–4. Citado nas pp. 9 e 48.
- FU, K. S.; GONZALEZ, R. C.; LEE, C. G.; FREEMAN, H. **Robotics: control, sensing, vision, and intelligence**. [S.l.]: McGraw-Hill New York, 1987. v. 1. Citado na p. 36.
- GONZALEZ, R. C.; WOODS, R. E.; EDDINS, S. L. **Digital image processing using MATLAB**. [S.l.]: Pearson Education India, 2004. Citado na p. 39.

-
- Google DeepMind. Shaping the future of advanced robotics with sara-rt. **Google DeepMind**, 2023. Acessado em: 07 out. 2024. Disponível em: <https://deepmind.com/research/publications/sara-rt-advancements>. Citado na p. 41.
- GROOVER, M. P. **Automation, production systems, and computer-integrated manufacturing**. [S.l.]: Pearson Education India, 2016. Citado na p. 22.
- HARTLEY, R.; ZISSERMAN, A. **Multiple view geometry in computer vision**. [S.l.]: Cambridge university press, 2003. Citado nas pp. 9, 44, 46, 47, 48 e 49.
- HARTLEY, R.; ZISSERMAN, A. **Multiple View Geometry in Computer Vision**. 2nd. ed. Cambridge, UK: Cambridge University Press, 2004. ISBN 9780521540513. Citado na p. 98.
- HAYATI, S. Robot arm geometric link parameter estimation. *In: Proceedings of IEEE Conference on Robotics and Automation*. [S.l.: s.n.], 1986. p. 601–608. Citado nas pp. 64 e 70.
- HEIKKILÄ, J. **Camera Calibration: Theory and Practice**. 2nd edition. ed. [S.l.]: Springer, 2018. ISBN 978-3-319-45678-9. Citado na p. 46.
- HOLLERBACH, J. M.; WAMPLER, C. W. Calibration of robot manipulators. **The International Journal of Robotics Research**, v. 15, n. 1, p. 555–571, 1996. Citado na p. 69.
- IBARGUREN, A.; MAURTUA, I.; PÉREZ, M. A.; SIERRA, B. Multiple target tracking based on particle filtering for safety in industrial robotic cells. **Robotics and Autonomous Systems**, Elsevier, v. 72, p. 105–113, 2015. Citado na p. 21.
- IFR. **Global Robotics Race: Korea, Singapore and Germany in the Lead**. 2022. News. Acesso em: outubro de 2024. Disponível em: <https://ifr.org/ifr-press-releases/news/global-robotics-race-korea-singapore-and-germany-in-the-lead>. Citado na p. 19.
- ISO9283. **ISO 9283: Manipulating industrial robots — Performance criteria and related test methods**. 1998. Citado na p. 71.
- JENSEN, T.; NIELSEN, O. Adaptive vision-based control for industrial robots using image features. **International Journal of Robotics Research**, v. 40, n. 5-6, p. 712–729, 2021. Citado na p. 22.
- JESUS, E. O.; JR, R. C. A utilização de filtros gaussianos na análise de imagens digitais. **Proceeding Series of the Brazilian Society of Computational and Applied Mathematics**, v. 3, n. 1, 2015. Citado na p. 51.
- KAEHLER, A.; BRADSKI, G. **Learning OpenCV 3: computer vision in C++ with the OpenCV library**. [S.l.]: "O'Reilly Media, Inc.", 2016. Citado na p. 44.

- KELLY, R. Robust asymptotically stable visual servoing of planar robots. **IEEE Transactions on Robotics and Automation**, IEEE, v. 12, n. 5, p. 759–766, 1996. Citado nas pp. 33, 34 e 35.
- KELLY, R.; CARELLI, R.; NASISI, O.; KUCHEN, B.; REYES, F. Stable visual servoing of camera-in-hand robotic systems. **IEEE/ASME transactions on mechatronics**, IEEE, v. 5, n. 1, p. 39–48, 2000. Citado nas pp. 33, 34 e 35.
- KIM, B.-K.; CHUNG, S.-W.; SONG, M.-K.; SONG, W.-J. Correcting radial lens distortion with advanced outlier elimination. *In: IEEE. 2010 International Conference on Audio, Language and Image Processing*. [S.l.], 2010. p. 1693–1699. Citado nas pp. 9, 48 e 49.
- KIM, S.-H.; PARK, J.-W. A robust calibration method for camera parameters in robotics. **IEEE Transactions on Robotics**, v. 38, n. 4, p. 1230–1242, 2022. Citado na p. 46.
- KJELLANDER, J.; NILSSON, H. Integrated calibration techniques for robotic systems. **Journal of Automation and Systems Engineering**, v. 3, n. 2, p. 129–140, 2009. Citado na p. 24.
- KJELLANDER, J. A.; RAHAYEM, M. An integrated platform for 3d measurement with geometric reverse engineering. **Computer-Aided Design and Applications**, Taylor & Francis, v. 6, n. 6, p. 877–887, 2009. Citado nas pp. 20 e 23.
- KOFMAN, J.; WU, X.; LUU, T. J.; VERMA, S. Teleoperation of a robot manipulator using a vision-based human-robot interface. **IEEE transactions on industrial electronics**, IEEE, v. 52, n. 5, p. 1206–1219, 2005. Citado nas pp. 33, 34 e 35.
- KOOTBALLY, Z. Industrial robot capability models for agile manufacturing. **Industrial Robot: An International Journal**, Emerald Group Publishing Limited, v. 43, n. 5, p. 481–494, 2016. Citado na p. 20.
- KRISHNA, R. Computer vision: Foundations and applications. **Reference Book**, v. 213, 2017. Citado na p. 53.
- LARSSON, S. r.; KJELLANDER, J. An industrial robot and a laser scanner as a flexible solution towards an automatic system for reverse engineering of unknown objects. *In: Engineering Systems Design and Analysis*. [S.l.: s.n.], 2004. v. 4174, p. 341–350. Citado nas pp. 20 e 22.
- LITWILLER, D. **CCD vs. CMOS: Facts and Fiction reprinted from Jan. 2001 Issue of Photonics Spectra**. [S.l.]: Laurin Publishing Co. Inc.(4 pages), 2001. 154–158 p. Citado na p. 37.
- LIU, C.; ZHU, J.; YANG, X.; YE, S. Autocar-body on-line laser inspection system with machine vision. **Chin. J. Sci. Instrum**, v. 25, p. 671–672, 2004. Citado na p. 20.

- LIU, Y.-H.; WANG, H.; WANG, C.; LAM, K. K. Uncalibrated visual servoing of robots using a depth-independent interaction matrix. **IEEE Transactions on Robotics**, IEEE, v. 22, n. 4, p. 804–817, 2006. Citado nas pp. 33, 34 e 35.
- MA, H.; ZHOU, L. Optimization techniques in camera calibration for industrial applications. **International Journal of Computer Vision**, Springer, v. 129, p. 789–804, 2021. Citado na p. 46.
- MA, J.; FAN, X.; YANG, S. X.; ZHANG, X.; ZHU, X. Contrast limited adaptive histogram equalization-based fusion in yiq and hsi color spaces for underwater image enhancement. **International Journal of Pattern Recognition and Artificial Intelligence**, World Scientific, v. 32, n. 07, p. 1854018, 2018. Citado na p. 38.
- MAIA, G.; PORFÍRIO, V. A. O processo de detecção de bordas de canny: fundamentos, algoritmos e avaliação experimental. 2002. Citado na p. 51.
- MICHALOS, G.; MAKRIS, S.; SPILIOPOULOS, J.; MISIOS, I.; TSAROUCHI, P.; CHRYSOLOURIS, G. Robo-partner: Seamless human-robot cooperation for intelligent, flexible and safe operations in the assembly factories of the future. **Procedia CIRP**, Elsevier, v. 23, p. 71–76, 2014. Citado na p. 21.
- MILLER, W. T. Real-time application of neural networks for sensor-based control of robots with vision. **IEEE Transactions on Systems, Man, and Cybernetics**, IEEE, v. 19, n. 4, p. 825–831, 1989. Citado nas pp. 33, 34 e 35.
- MORVAN, Y. Acquisition, compression and rendering of depth and texture for multi-view video. 2009. Citado nas pp. 9, 47 e 48.
- NUBIOLA, A.; BONEV, I. A. Experimental comparison of calibration methods for industrial robots. **Mechanism and Machine Theory**, v. 70, p. 1–13, 2013. Citado na p. 70.
- QIN, J.; LIU, Y.; GROSVENOR, R. A categorical framework of manufacturing for industry 4.0 and beyond. **Procedia cirp**, Elsevier, v. 52, p. 173–178, 2016. Citado na p. 19.
- RAINER, D.; ALEXANDER, H. Industrie 4.0: hit or hype. **Industrial Electronics Magazine**, v. 8, n. 2, p. 56–58, 2014. Citado na p. 20.
- RAMIREZ, D.; ORTEGA, P. Vision-based calibration and control in industrial robotics: A review. **Robotics and Computer-Integrated Manufacturing**, v. 64, p. 101–110, 2020. Citado na p. 22.
- ROBOTICS. A. product manual irb 140. In: N. M2000. **ABB Robotics Products AB publication**. [S.l.], 2003. p. 7564–1. Citado nas pp. 9, 73 e 75.
- ROTH, Z.; MOORING, B.; RAVANI, B. An overview of robot calibration. **IEEE Journal on Robotics and Automation**, IEEE, v. 3, n. 5, p. 377–385, 1987. Citado na p. 23.

-
- SANNEMAN, L.; FOURIE, C.; SHAH, J. A. The state of industrial robotics: Emerging technologies, challenges, and key research directions. **arXiv preprint arXiv:2010.14537**, 2020. Disponível em: <https://arxiv.org/abs/2010.14537>. Citado na p. 22.
- SCHRÖER, K. Theory of kinematic modelling and numerical procedures for robot calibration. **Robot Calibration**, Chapman & Hall London, v. 157196, p. 44, 1993. Citado na p. 23.
- SCHWEIKARD, A.; BOVA, F. J. *et al.* Robot calibration for neurosurgery. **IEEE Transactions on Robotics and Automation**, v. 9, n. 6, p. 722–731, 1992. Citado na p. 70.
- SHIAKOLAS, P.; CONRAD, K.; YIH, T. On the accuracy, repeatability, and degree of influence of kinematics parameters for industrial robots. **International journal of modelling and simulation**, Taylor & Francis, v. 22, n. 4, p. 245–254, 2002. Citado na p. 23.
- SHIRINZADEH, B. *et al.* Design of a robust calibration system for robot manipulators. **Robotics and Autonomous Systems**, v. 47, p. 157–169, 2004. Citado na p. 70.
- SPONG, M. W.; HUTCHINSON, S.; VIDYASAGAR, M. **Robot modeling and control**. [S.l.]: John Wiley & Sons, 2020. Citado nas pp. 17, 63, 64, 65 e 114.
- SS, B. T. M. Vision-based control of robotic manipulator for citrus harvesting. **Computers and electronics in agriculture**, Elsevier, v. 102, p. 146–158, 2014. Citado nas pp. 32, 33, 34 e 35.
- SUN, C.-C.; RUAN, S.-J.; SHIE, M.-C.; PAI, T.-W. Dynamic contrast enhancement based on histogram specification. **IEEE Transactions on Consumer Electronics**, v. 51, n. 4, p. 1300–1305, 2005. Citado na p. 38.
- SZELISKI, R. **Computer vision: algorithms and applications**. [S.l.]: Springer Nature, 2010. 111–123 p. Citado na p. 50.
- TOQUICA, J. S.; MOTTA, J. M. S. A novel approach for robot calibration based on measurement sub-regions with comparative validation. **The International Journal of Advanced Manufacturing Technology**, Springer, p. 1–14, 2024. Citado nas pp. 42 e 43.
- VIANNA, M. E. Calibração de sistemas de visão computacional para aplicação em automação e robótica. **Belo Horizonte**, 2009. Citado nas pp. 9, 40 e 41.
- WABNER, M.; RENTZSCH, H.; IHLENFELDT, S.; OTTO, A. Technological robot—machine tool collaboration for agile production. **Frontiers in Robotics and AI**, Frontiers Media SA, v. 9, p. 1027173, 2022. Disponível em: <https://www.frontiersin.org/articles/10.3389/frobt.2022.1027173/full>. Citado na p. 22.
- WANG, H.; LI, Y. Machine learning-based camera calibration techniques for autonomous robotics. **Artificial Intelligence and Robotics**, Taylor & Francis, v. 15, p. 95–110, 2023. Citado na p. 46.

- WANG, H.; LIU, Y.-H.; ZHOU, D.; . Adaptive visual servoing using point and line features with an uncalibrated eye-in-hand camera. **IEEE Transactions on Robotics**, IEEE, v. 24, n. 4, p. 843–857, 2008. Citado nas pp. 33 e 34.
- WANG, S.; WAN, J.; LI, D.; ZHANG, C. Implementing smart factory of industrie 4.0: an outlook. **International journal of distributed sensor networks**, SAGE Publications Sage UK: London, England, v. 12, n. 1, p. 3159805, 2016. Citado na p. 20.
- YANG, R.; CHENG, S.; YANG, W.; CHEN, Y. Robust and accurate surface measurement using structured light. **IEEE Transactions on Instrumentation and Measurement**, IEEE, v. 57, n. 6, p. 1275–1280, 2008. Citado na p. 20.
- YIHONG, G.; HONGJIANG, Z.; CHUAN, C. H. An image database system with fast image indexing capability based on color histograms. *In: IEEE. Proceedings of TENCON'94-1994 IEEE Region 10's 9th Annual International Conference on: 'Frontiers of Computer Technology'*. [S.l.], 1994. p. 407–411. Citado na p. 36.
- ZHANG, W.; LI, J. Advances in camera calibration for visual perception in robotics. **Journal of Robotics and Autonomous Systems**, Elsevier, v. 98, p. 45–58, 2020. Citado na p. 46.
- ZHANG, Y.; LI, Q.; WANG, J. Computer vision in robotics: Emerging trends for autonomous control. **Automation and Robotics Journal**, v. 37, n. 3, p. 301–320, 2023. Citado na p. 22.
- ZHANG, Z. A flexible new technique for camera calibration. **IEEE Transactions on pattern analysis and machine intelligence**, IEEE, v. 22, n. 11, p. 1330–1334, 2000. Citado na p. 99.
- ZHAO, L.; WAN, Y. An edge-guided exact histogram specification method for enhancing extremely dark images. *In: 2017 International Conference on Computer Technology, Electronics and Communication (ICCTEC)*. [S.l.: s.n.], 2017. p. 612–616. Citado na p. 38.
- ZHAO, Q.; WAN, X. Precise camera intrinsic parameter estimation for robotic applications. **Robotics and Automation Letters**, v. 4, n. 3, p. 2284–2291, 2019. Citado na p. 46.
- ZHOU, Z.; MA, L.; LIU, X.; CAO, Z.; YU, J. Simultaneously calibration of multi hand–eye robot system based on graph. **IEEE Transactions on Industrial Electronics**, IEEE, v. 71, n. 5, p. 5010–5020, 2023. Citado na p. 40.
- ZHUANG, H.; ROTH, Z. S. A self-calibration technique for stereovision systems and its applications to 3-d vision metrology. **IEEE Transactions on Robotics and Automation**, v. 10, n. 2, p. 197–206, 1994. Citado na p. 69.
- ZIOU, D.; TABBONE, S. *et al.* Edge detection techniques-an overview. **Pattern Recognition and Image Analysis C/C of Raspoznavaniye Obrazov I Analiz Izobrazhenii**, NAUKA/INTERPERIODICA PUBLISHING, v. 8, p. 537–559, 1998. Citado na p. 51.

ZOU, S.; YU, H.; ZHANG, L. Recent advances in robot vision systems for industrial applications. **IEEE Transactions on Industrial Electronics**, v. 69, n. 7, p. 7125–7138, 2022. Citado na p. [22](#).

Apêndices

Apêndice A – Cinemática Robótica

A.1 Cinemática da Velocidade e da Aceleração

Aqui serão apresentadas as propriedades e deduções das relações entre as velocidades lineares e angulares do órgão terminal (ou de qualquer outro ponto do manipulador) e as velocidades das juntas baseados em [Spong, Hutchinson e Vidyasagar \(2020\)](#). Também serão discutidas a relação entre as acelerações das juntas e do órgão terminal (ou de qualquer outro ponto do manipulador).

Considere-se uma matriz de rotação variante no tempo $\mathbf{R} = \mathbf{R}(t)$. Tendo em vista a ortogonalidade de \mathbf{R} , pode-se escrever

$$\mathbf{R}(t)\mathbf{R}^T(t) = \mathbf{I} \quad (\text{A.1})$$

Derivando em relação ao tempo:

$$\dot{\mathbf{R}}(t)\mathbf{R}^T(t) + \mathbf{R}(t)\dot{\mathbf{R}}^T(t) = \mathbf{0} \quad (\text{A.2})$$

Definindo

$$\mathbf{S}(t) = \dot{\mathbf{R}}(t)\mathbf{R}^T(t) \quad (\text{A.3})$$

pode-se verificar facilmente que a matriz \mathbf{S} é anti-simétrica, pois

$$\mathbf{S}(t) + \mathbf{S}^T(t) = \mathbf{0} \quad (\text{A.4})$$

Pós-multiplicando a [equação \(A.3\)](#) por $\mathbf{R}(t)$, e levando em conta a [equação \(A.2\)](#) e a [equação \(A.3\)](#), chega-se a

$$\dot{\mathbf{R}}(t) = \mathbf{S}(t)\mathbf{R}(t) \quad (\text{A.5})$$

que permite expressar a derivada temporal da matriz de rotação em função dela mesma e da matriz anti-simétrica $\mathbf{S}(t)$.

Essa matriz anti-simétrica $\mathbf{S}(t)$ tem uma interpretação física interessante. Considere-se um vetor constante \mathbf{p}' e o vetor função do tempo $\mathbf{p}(t) = \mathbf{R}(t)\mathbf{p}'$. Derivando em relação ao tempo o vetor $\mathbf{p}(t)$:

$$\dot{\mathbf{p}}(t) = \dot{\mathbf{R}}(t)\mathbf{p}' \quad (\text{A.6})$$

ou, tendo em vista a equação (A.3):

$$\dot{\mathbf{p}}(t) = \mathbf{S}(t) \mathbf{R}(t) \mathbf{p}' \quad (\text{A.7})$$

Denotando o vetor velocidade angular instantânea do sistema $\mathbf{R}(t)$ com relação ao sistema inercial por $\omega(t)$, sabe-se da mecânica que

$$\dot{\mathbf{p}}(t) = \omega(t) \times \mathbf{R}(t) \mathbf{p}' \quad (\text{A.8})$$

Portanto, observando as duas últimas equações, verifica-se que a matriz $\mathbf{S}(t)$ descreve o produto matricial entre o vetor $\omega(t)$ e o vetor $\mathbf{R}(t)\mathbf{p}'$. A matriz $\mathbf{S}(t)$ representa o vetor $\omega(t) = [\omega_x \ \omega_y \ \omega_z]^T$ na forma

$$\mathbf{S} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad (\text{A.9})$$

o que justifica que $\mathbf{S}(t) = \mathbf{S}(\omega(t))$. No importante caso particular dos vetores unitários \mathbf{i} , \mathbf{j} e \mathbf{k} , tem-se

$$\mathbf{S}(\mathbf{i}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \quad (\text{A.10})$$

$$\mathbf{S}(\mathbf{j}) = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \quad (\text{A.11})$$

$$\mathbf{S}(\mathbf{k}) = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (\text{A.12})$$

Também pode-se provar que, se \mathbf{R} é uma matriz de rotação, então

$$\mathbf{R} \mathbf{S}(\omega) \mathbf{R}^T = \mathbf{S}(\mathbf{R}\omega) \quad (\text{A.13})$$

A matriz \mathbf{S} apresenta algumas propriedades interessantes que estão relacionadas a seguir.

(1) Linearidade

$$\mathbf{S}(\alpha \mathbf{a} + \beta \mathbf{b}) = \alpha \mathbf{S}(\mathbf{a}) + \beta \mathbf{S}(\mathbf{b}) \quad (\text{A.14})$$

onde α e β são escalares e \mathbf{a} e \mathbf{b} são vetores.

(2) Para qualquer vetor \mathbf{p}

$$\mathbf{S}(\mathbf{a})\mathbf{p} = \mathbf{a} \times \mathbf{p} \quad (\text{A.15})$$

isto é, o produto matricial da matriz anti-simétrica associada ao vetor \mathbf{a} , $\mathbf{S}(\mathbf{a})$, pelo vetor \mathbf{p} , é igual ao produto vetorial do vetor \mathbf{a} pelo vetor \mathbf{p} .

(3) Seja a matriz ortogonal 3x3 \mathbf{R} e sejam \mathbf{a} e \mathbf{b} dois vetores no espaço 3D. Então

$$\mathbf{R}(\mathbf{a} \times \mathbf{b}) = \mathbf{R}\mathbf{a} \times \mathbf{R}\mathbf{b} \quad (\text{A.16})$$

ou seja, se primeiro forem girados \mathbf{a} e \mathbf{b} usando a matriz de transformação \mathbf{R} e depois formado o produto vetorial dos vetores girados $\mathbf{R}\mathbf{a}$ e $\mathbf{R}\mathbf{b}$, o resultado é o mesmo que o obtido primeiro formando o produto vetorial $\mathbf{a} \times \mathbf{b}$ e depois girando o vetor produto.

(4)

$$\mathbf{R}\mathbf{S}(\mathbf{a})\mathbf{R}^T = \mathbf{S}(\mathbf{R}\mathbf{a}) \quad (\text{A.17})$$

(5) Se $\mathbf{R} = \mathbf{R}(\theta)$ é uma matriz de rotação função apenas da variável θ , então

$$\frac{d\mathbf{R}}{d\theta} = \mathbf{S}\mathbf{R}(\theta) \quad (\text{A.18})$$

Seja $\mathbf{R}(t)$ uma matriz de rotação ortogonal 3x3, dependente do tempo. De acordo com a equação (A.5) e com a justificativa de que $\mathbf{S}(t) = \mathbf{S}(\omega(t))$, pode-se escrever

$$\dot{\mathbf{R}} = \mathbf{S}(\omega(t))\mathbf{R}(t) = \mathbf{S}\mathbf{R}(t) \quad (\text{A.19})$$

Velocidade

Considere-se, inicialmente, o caso da **rotação pura**. Seja um vetor \mathbf{p}_1 , definido no sistema móvel $O_1x_1y_1z_1$, o qual gira em relação ao sistema fixo $O_0x_0y_0z_0$. Então, o vetor dado é transformado para o sistema fixo através da relação

$$\mathbf{p}_0 = \mathbf{R}(t)\mathbf{p}_1 \quad (\text{A.20})$$

Para achar a velocidade em relação ao sistema fixo, basta derivar a equação (A.20) em relação ao tempo:

$$\begin{aligned} \dot{\mathbf{p}}_0 &= \mathbf{S}(\omega)\mathbf{R}(t)\mathbf{p}_1 \\ &= \mathbf{S}(\omega)\mathbf{p}_0 \\ &= \omega \times \mathbf{p}_0 \end{aligned} \quad (\text{A.21})$$

que é a familiar expressão da velocidade no caso da rotação pura.

Seja, agora, o caso geral de **translação e rotação**. Nesse caso, a matriz de transformação homogênea é dada por

$$\mathbf{H}_0^1(t) = \begin{bmatrix} \mathbf{R}_0^1(t) & \mathbf{d}_0^1(t) \\ \mathbf{0} & 1 \end{bmatrix} \quad (\text{A.22})$$

Por comodidade, serão omitidos o argumento t e os superescritos e subscritos que aparecem na matriz e no vetor da expressão acima. Assim, o vetor posição, em relação ao sistema da base, é dado por

$$\mathbf{p}_0 = \mathbf{d} + \mathbf{R}\mathbf{p}_1 \quad (\text{A.23})$$

Derivando em relação ao tempo, obtem-se o vetor **velocidade**:

$$\begin{aligned} \dot{\mathbf{p}}_0 &= \dot{\mathbf{R}}\mathbf{p}_1 + \dot{\mathbf{d}} \\ &= \mathbf{S}(\omega) \mathbf{R}\mathbf{p}_1 + \dot{\mathbf{d}} \\ &= \omega \times \mathbf{r} + \mathbf{v} \end{aligned} \quad (\text{A.24})$$

considerando que \mathbf{p}_1 é constante e que foi usada a equação (A.19) e adotada a notação $\mathbf{d} = \mathbf{v}$.

Na equação (A.24) \mathbf{v} é a velocidade linear da origem do sistema móvel em relação ao sistema fixo; ω é a velocidade angular do sistema móvel em relação ao sistema fixo e $\mathbf{r} = \mathbf{R}\mathbf{p}_1$ é o vetor posição \mathbf{p}_1 em relação ao sistema fixo.

Se o vetor \mathbf{p}_1 estiver se movimentando em relação ao sistema móvel, então deve-se adicionar ao termo \mathbf{v} o termo

$$R(t) \dot{\mathbf{p}}_1 \quad (\text{A.25})$$

que é a taxa de variação de \mathbf{p}_1 expressa no sistema $O_0x_0y_0z_0$.

Aceleração

A equação (A.24) pode ser colocada na forma

$$\begin{aligned} \dot{\mathbf{p}}_0 - \dot{\mathbf{d}} &= \dot{\mathbf{R}}\mathbf{p}_1 \\ &= \omega \times \mathbf{R}\mathbf{p}_1 \end{aligned} \quad (\text{A.26})$$

Derivando em relação ao tempo:

$$\begin{aligned} \ddot{\mathbf{p}}_0 - \ddot{\mathbf{d}} &= \dot{\omega} \times \mathbf{R}\mathbf{p}_1 + \omega \times (\dot{\mathbf{R}}\mathbf{p}_1) \\ &= \dot{\omega} \times \mathbf{r} + \omega \times (\omega \times \mathbf{r}) \end{aligned} \quad (\text{A.27})$$

A equação (A.27) pode ser escrita como

$$\ddot{\mathbf{p}}_0 = \dot{\omega} \times \mathbf{r} + \omega \times (\omega \times \mathbf{r}) + \mathbf{a} \quad (\text{A.28})$$

onde \mathbf{a} é a aceleração linear. O termo $\omega \times (\omega \times \mathbf{r})$ denomina-se aceleração centrípeta da partícula e está sempre dirigido para o eixo de rotação, sendo perpendicular a esse eixo. O termo é $\dot{\omega} \times \mathbf{r}$ chamado aceleração transversal.

Se o vetor \mathbf{p}_1 estiver variando com relação ao sistema móvel, a equação (A.28) deve ser modificada para

$$\ddot{\mathbf{p}}_0 = \dot{\omega} \times \mathbf{r} + \omega \times (\omega \times \mathbf{r}) + 2\omega \times \mathbf{R}\dot{\mathbf{p}}_1 + \mathbf{a} \quad (\text{A.29})$$

onde $\mathbf{a} = \mathbf{R}\ddot{\mathbf{p}}_1 + \ddot{\mathbf{d}}$. O termo $2\omega \times \mathbf{R}\dot{\mathbf{p}}_1$ é conhecido como **aceleração de Coriolis**.

Adição de Velocidades Angulares

Muitas vezes tem-se interesse em achar a velocidade angular resultante devida à rotação relativa de vários sistemas de coordenadas. Considere-se, inicialmente, a composição das velocidades angulares de apenas dois sistemas de coordenadas móveis, $O_1x_1y_1z_1$ e $O_2x_2y_2z_2$, em relação a um sistema fixo $O_0x_0y_0z_0$. Seja um ponto \mathbf{p} representado nos respectivos sistemas pelas relações

$$\begin{aligned} \mathbf{p}_0 &= \mathbf{R}_0^1 \mathbf{p}_1 + \mathbf{d}_0^1 \\ \mathbf{p}_1 &= \mathbf{R}_1^2 \mathbf{p}_2 + \mathbf{d}_1^2 \\ \mathbf{p}_0 &= \mathbf{R}_0^2 \mathbf{p}_2 + \mathbf{d}_0^2 \end{aligned} \quad (\text{A.30})$$

onde

$$\mathbf{R}_0^2 = \mathbf{R}_0^1 \mathbf{R}_1^2 \quad (\text{A.31})$$

e

$$\mathbf{d}_0^2 = \mathbf{d}_0^1 + \mathbf{R}_0^1 \mathbf{d}_1^2 \quad (\text{A.32})$$

Derivando a equação (A.31) em relação ao tempo:

$$\dot{\mathbf{R}}_0^2 = \dot{\mathbf{R}}_0^1 \mathbf{R}_1^2 + \mathbf{R}_0^1 \dot{\mathbf{R}}_1^2 \quad (\text{A.33})$$

O termo $\dot{\mathbf{R}}_0^2$ da expressão acima pode ser escrito como

$$\dot{\mathbf{R}}_0^2 = \mathbf{S}(\omega_0^2) \mathbf{R}_0^2 \quad (\text{A.34})$$

O primeiro termo do lado direito da equação (A.33) é simplesmente

$$\dot{\mathbf{R}}_0^1 \mathbf{R}_1^2 = \mathbf{S}(\omega_0^1) \mathbf{R}_0^1 \mathbf{R}_1^2 = \mathbf{S}(\omega_0^1) \mathbf{R}_0^2 \quad (\text{A.35})$$

Quanto ao segundo termo do lado direito da equação (A.33), usando a equação (A.16), obtém-se

$$\begin{aligned} \mathbf{R}_0^1 \dot{\mathbf{R}}_1^2 &= \mathbf{R}_0^1 \mathbf{S}(\omega_1^2) \mathbf{R}_1^2 \\ &= \mathbf{R}_0^1 \mathbf{S}(\omega_1^2) \mathbf{R}_0^{1T} \mathbf{R}_0^1 \mathbf{R}_1^2 = \mathbf{S}(\mathbf{R}_0^1 \omega_1^2) \mathbf{R}_0^1 \mathbf{R}_1^2 \\ &= \mathbf{S}(\mathbf{R}_0^1 \omega_1^2) \mathbf{R}_0^2 \end{aligned} \quad (\text{A.36})$$

Combinando as expressões acima

$$\mathbf{S}(\omega_0^2) \mathbf{R}_0^2 = \left[\mathbf{S}(\omega_0^1) + \mathbf{S}(\mathbf{R}_0^1 \omega_1^2) \right] \mathbf{R}_0^1 \mathbf{R}_1^2 \quad (\text{A.37})$$

Tendo em vista que $\mathbf{S}(\mathbf{a}) + \mathbf{S}(\mathbf{b}) = \mathbf{S}(\mathbf{a} + \mathbf{b})$, vê-se então que

$$\omega_0^2 = \omega_0^1 + \mathbf{R}_0^1 \omega_1^2 \quad (\text{A.38})$$

Em outras palavras, as velocidades angulares podem ser somadas, desde que estejam expressas em relação ao mesmo sistema de coordenadas, no caso o sistema $O_0x_0y_0z_0$.

A equação (A.38) pode ser estendida para qualquer número de sistemas de coordenadas:

$$\omega_0^n = \omega_0^1 + \mathbf{R}_0^1 \omega_1^2 + \mathbf{R}_0^2 \omega_2^3 + \mathbf{R}_0^3 \omega_3^4 + \cdots + \mathbf{R}_0^{n-1} \omega_{n-1}^n \quad (\text{A.39})$$

Apêndice B – Códigos de programação

Os códigos de programação desenvolvidos para este trabalho, incluindo os algoritmos de programação em MATLAB e de programação utilizada para o robô ABB IRB 140 em RAPID , são disponibilizados para consulta no repositório online. Eles podem ser acessados através do seguinte link:

- https://drive.google.com/drive/folders/1O13wg-gNa3vT3Ifwyccm81g05fVsbsxT?usp=drive_link

Apêndice C – Dados Experimentais

Todos os Dados Experimentais podem ser acessados no link a seguir:

- <https://drive.google.com/drive/folders/1ml5j6pJKCcYeNiuHAYMx9GcQVbDfPKmp?usp=sharing>

Apêndice D – Resultados Experimentais

Resultados do 2º ao 5º experimento:

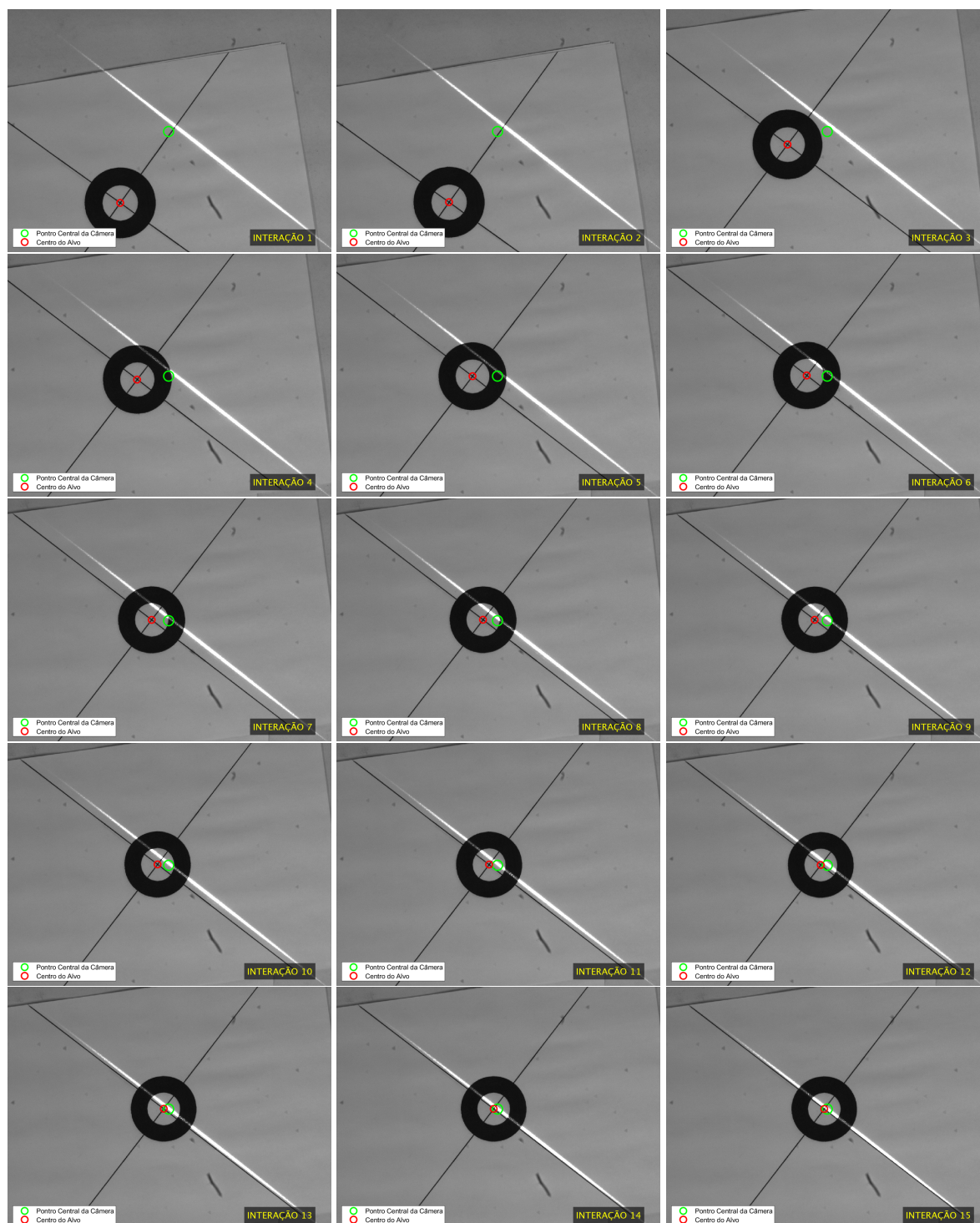


Figura D.1 – Imagens capturadas pela câmera no decorrer do movimento (1ª a 15ª interação) - Experimento 2.

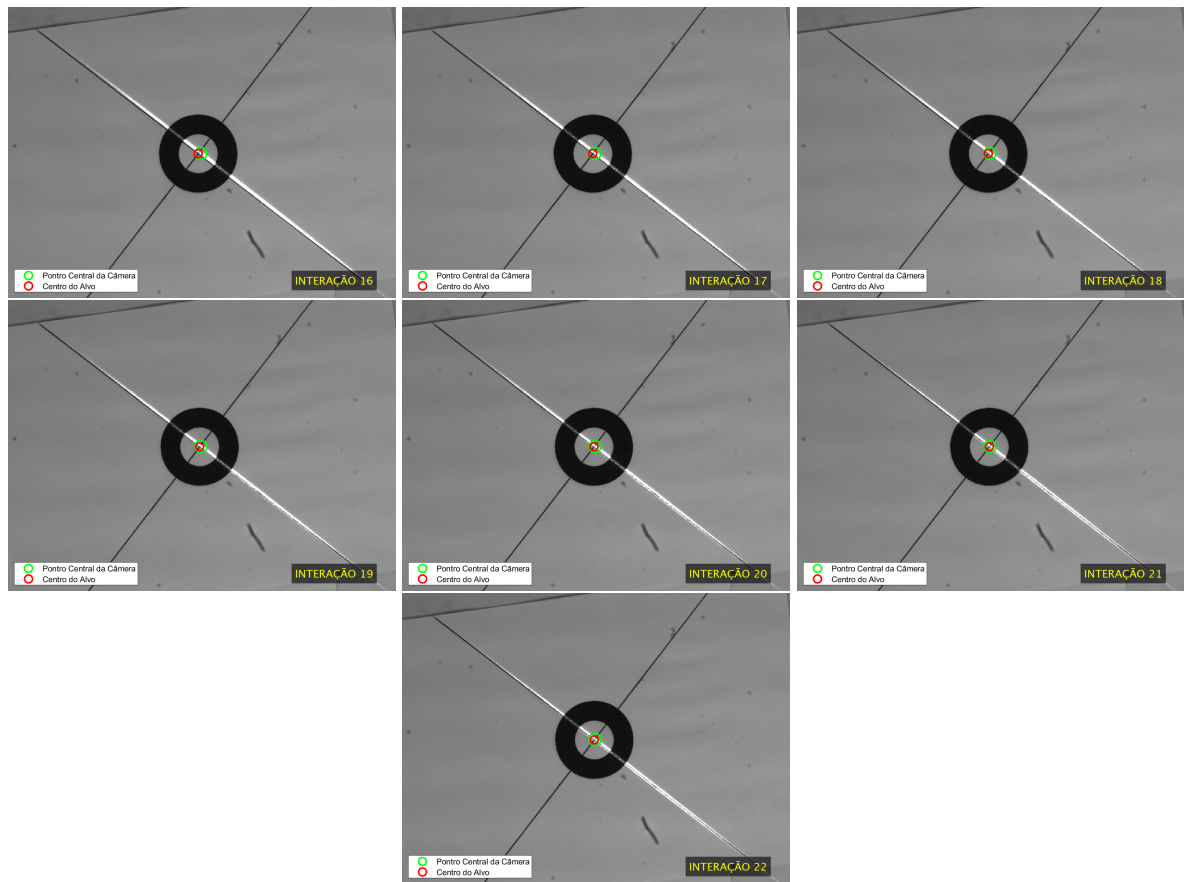


Figura D.2 – Imagens capturadas pela câmera no decorrer do movimento (16ª a 22ª interação) - Experimento 2.

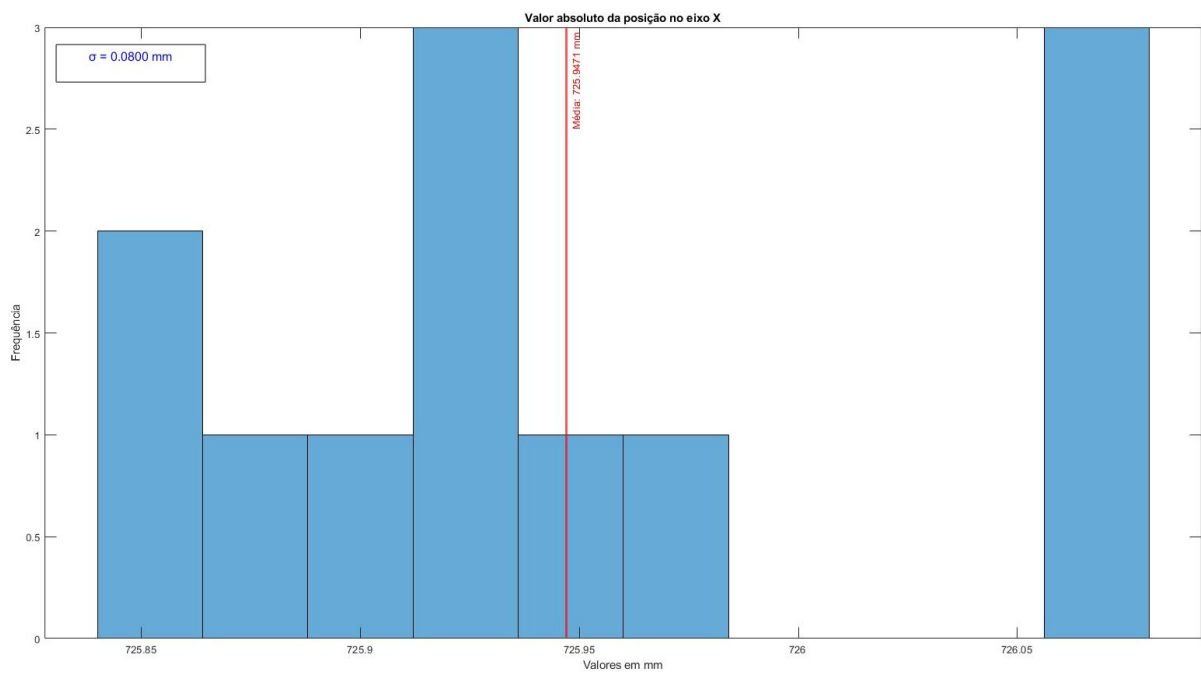


Figura D.3 – Valor absoluto da posição no eixo x - Experimento 2.

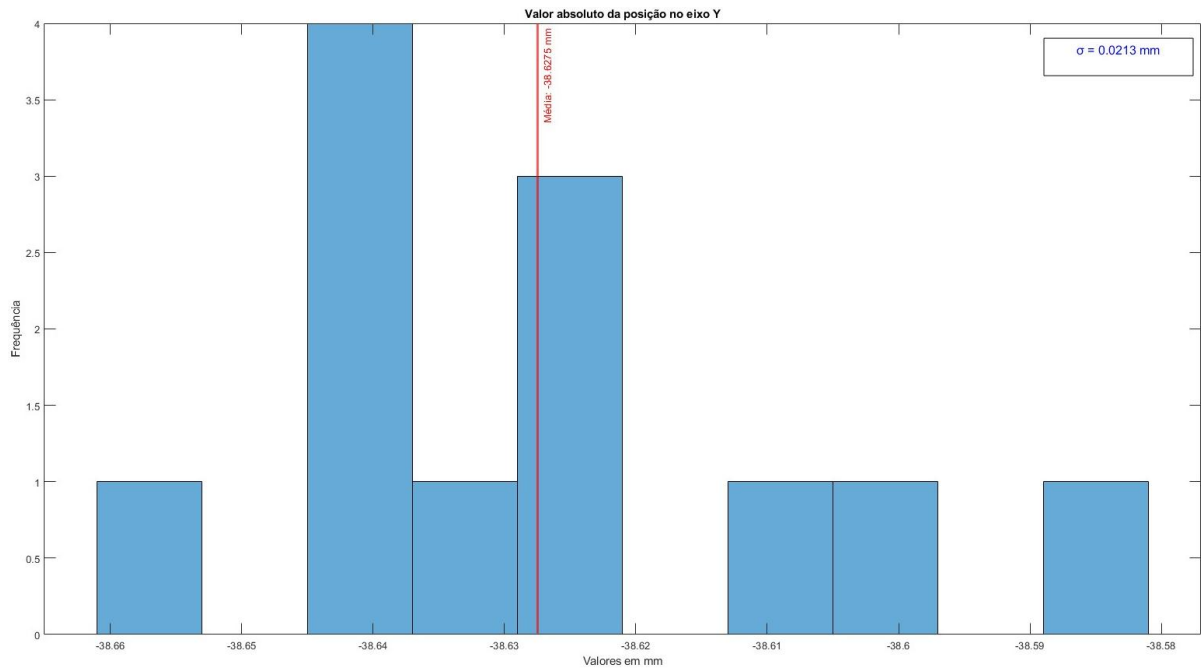


Figura D.4 – Valor absoluto da posição no eixo y - Experimento 2.

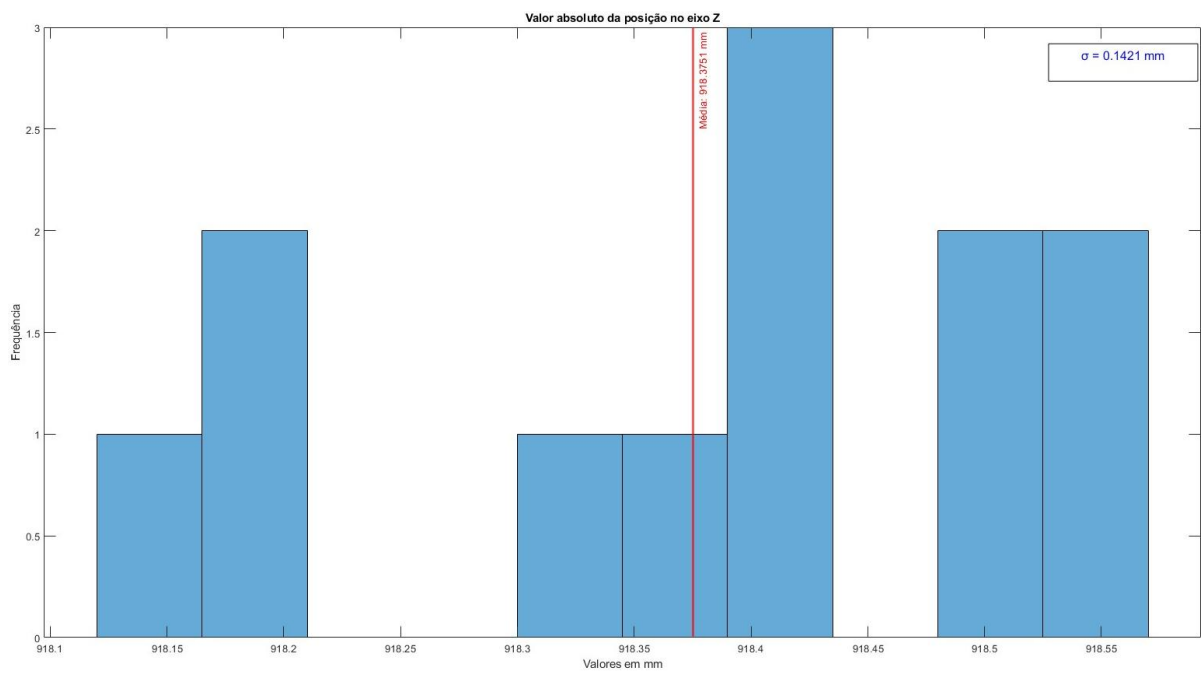


Figura D.5 – Valor absoluto da posição no eixo z - Experimento 2.

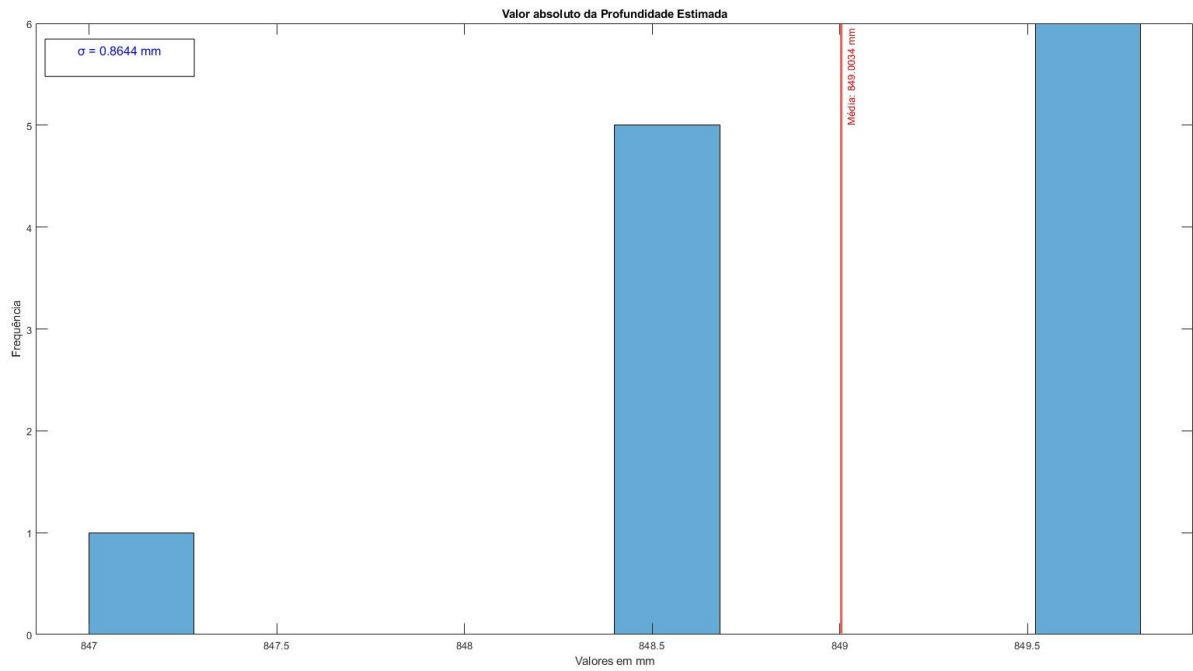


Figura D.6 – Valor absoluto da Profundidade Estimada - Experimento 2.

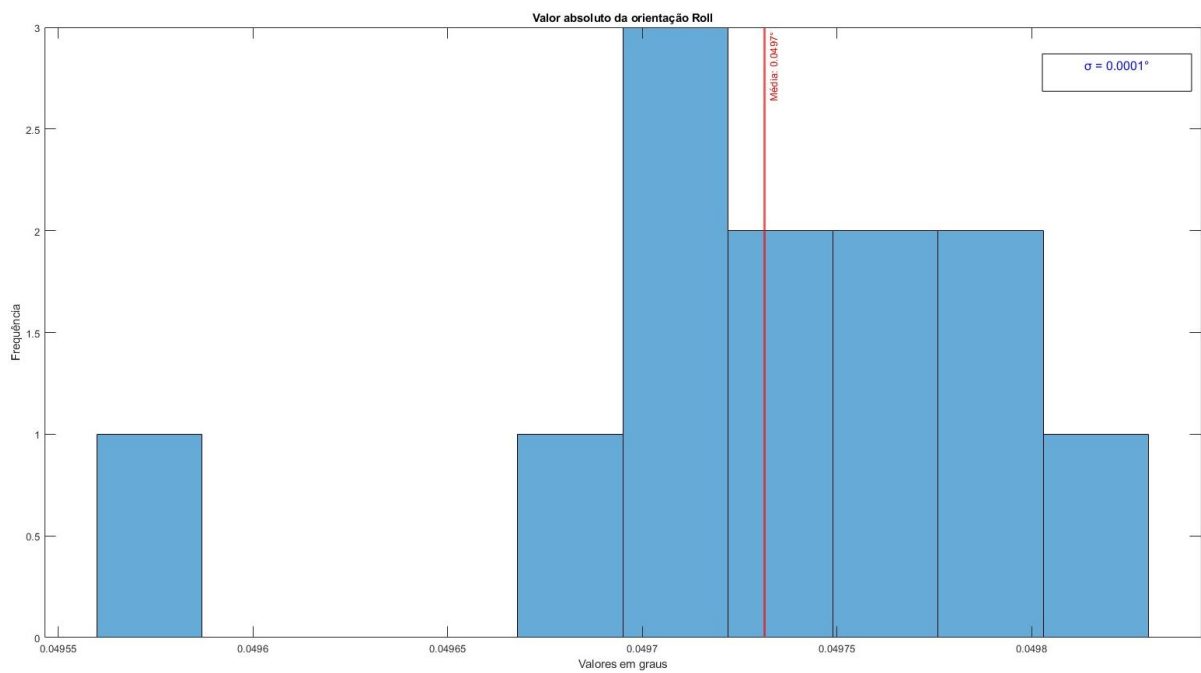


Figura D.7 – Valor absoluto da orientação Roll - Experimento 2.

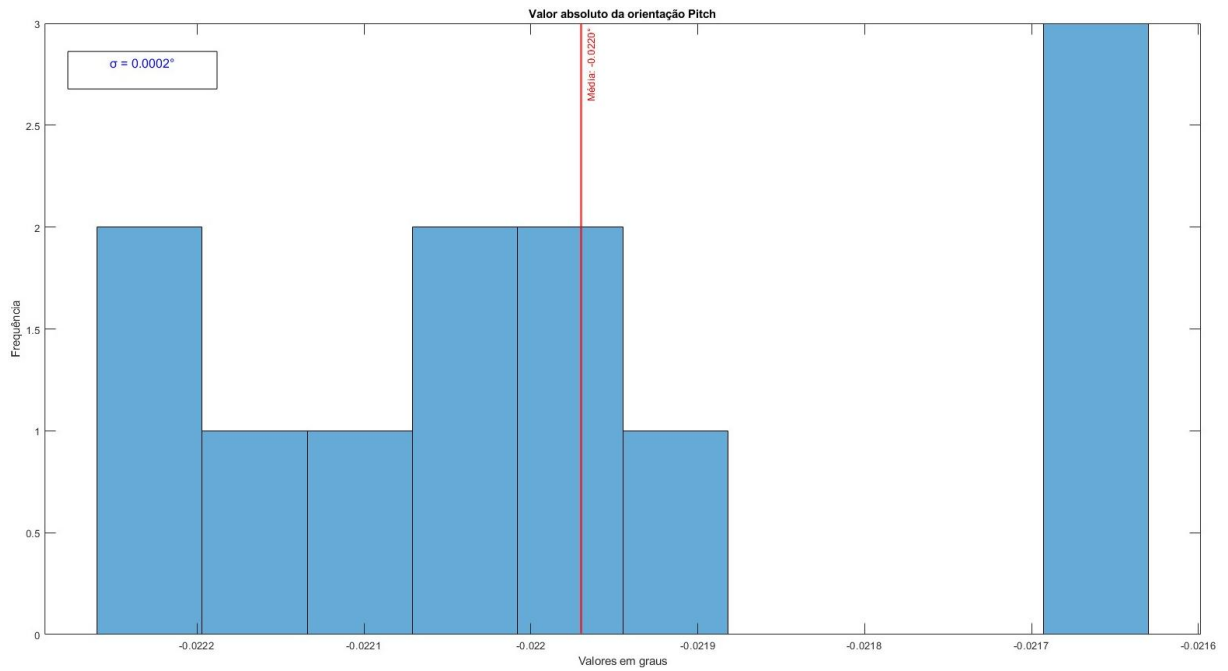


Figura D.8 – Valor absoluto da orientação Pitch - Experimento 2.

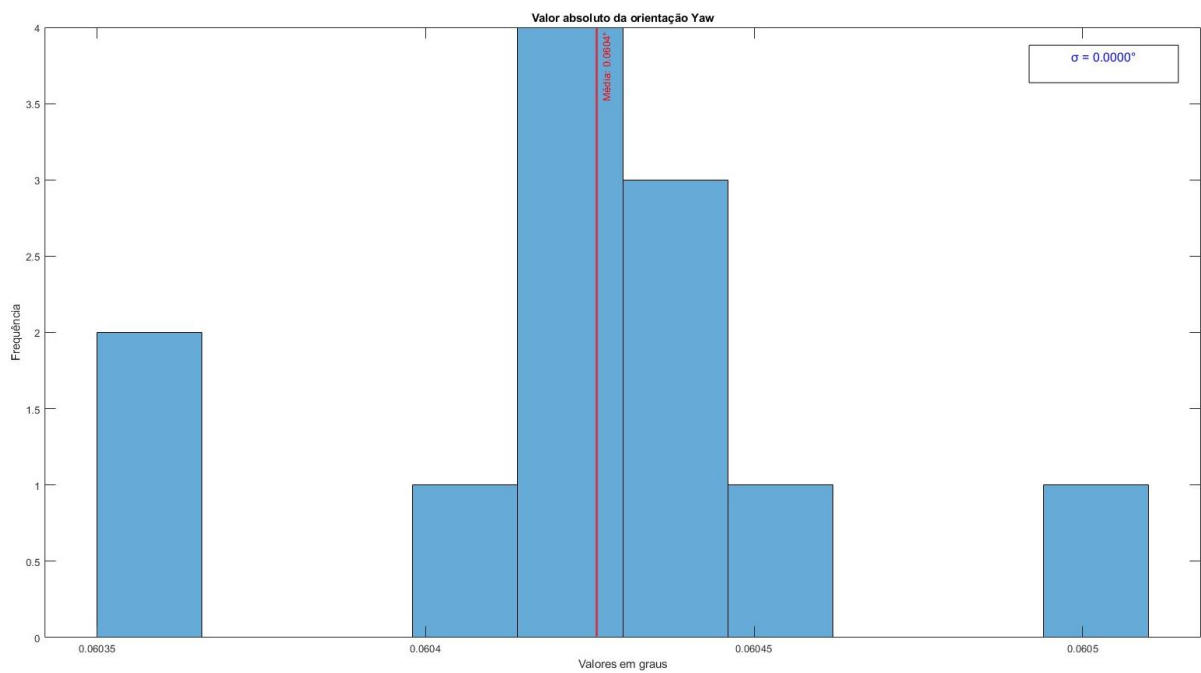


Figura D.9 – Valor absoluto da orientação Yaw - Experimento 2.

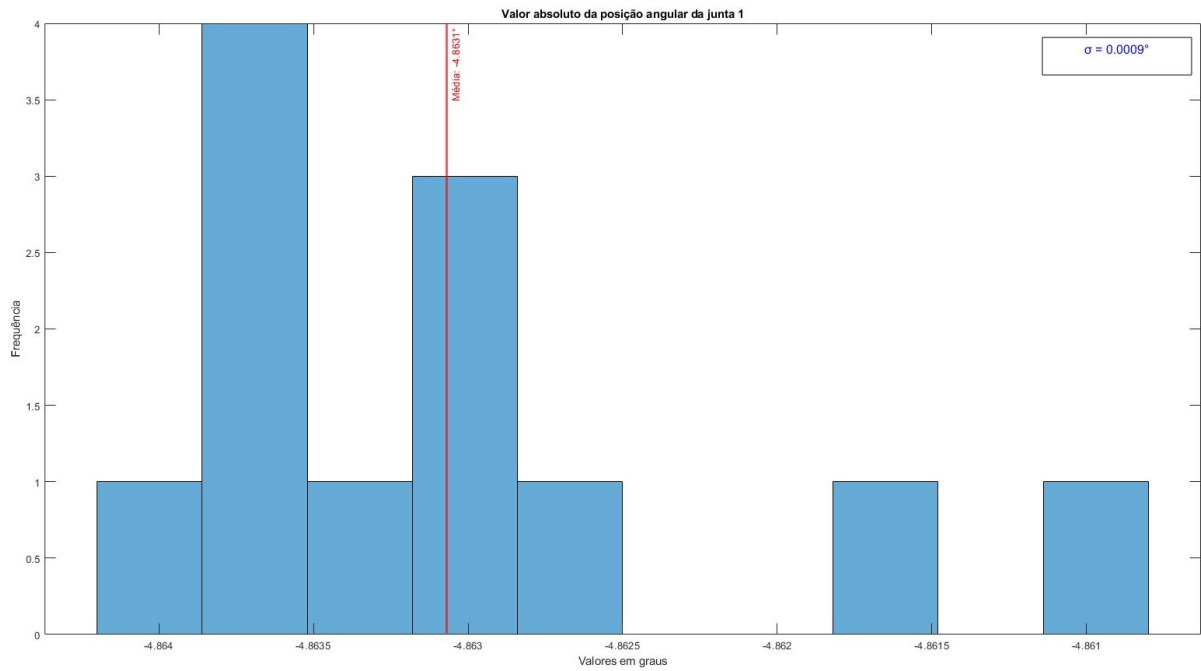


Figura D.10 – Valor absoluto da rotação da Junta 1 - Experimento 2.

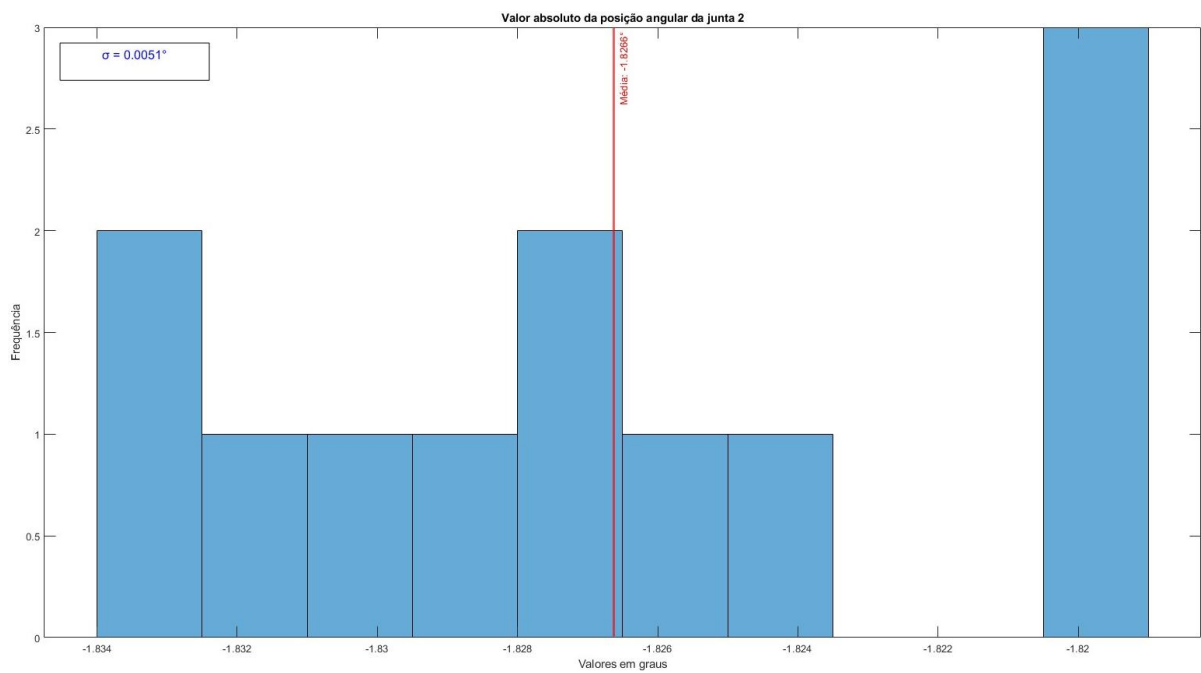


Figura D.11 – Valor absoluto da rotação da Junta 2 - Experimento 2.

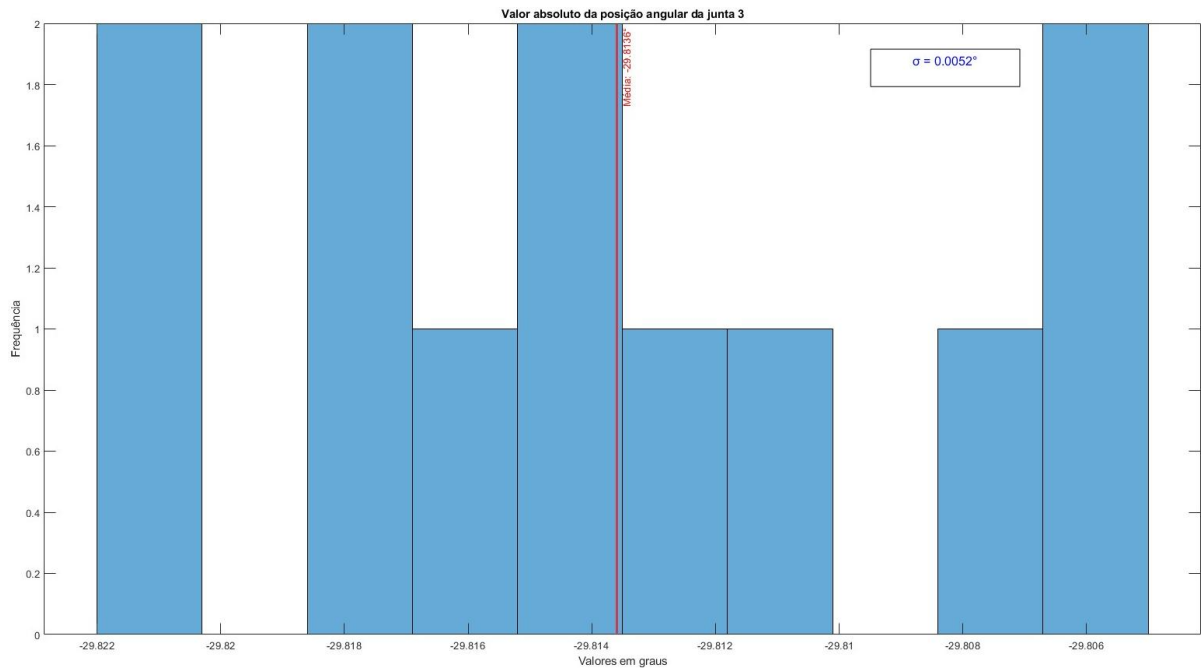


Figura D.12 – Valor absoluto da rotação da Junta 3 - Experimento 2.

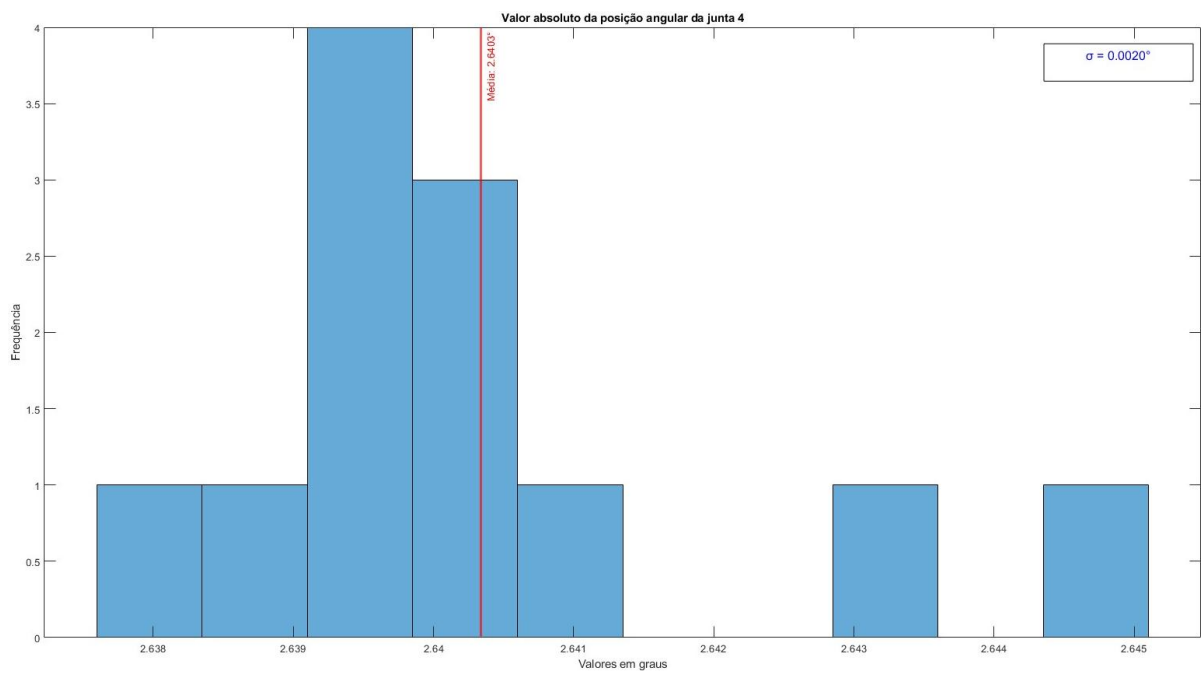


Figura D.13 – Valor absoluto da rotação da Junta 4 - Experimento 2.

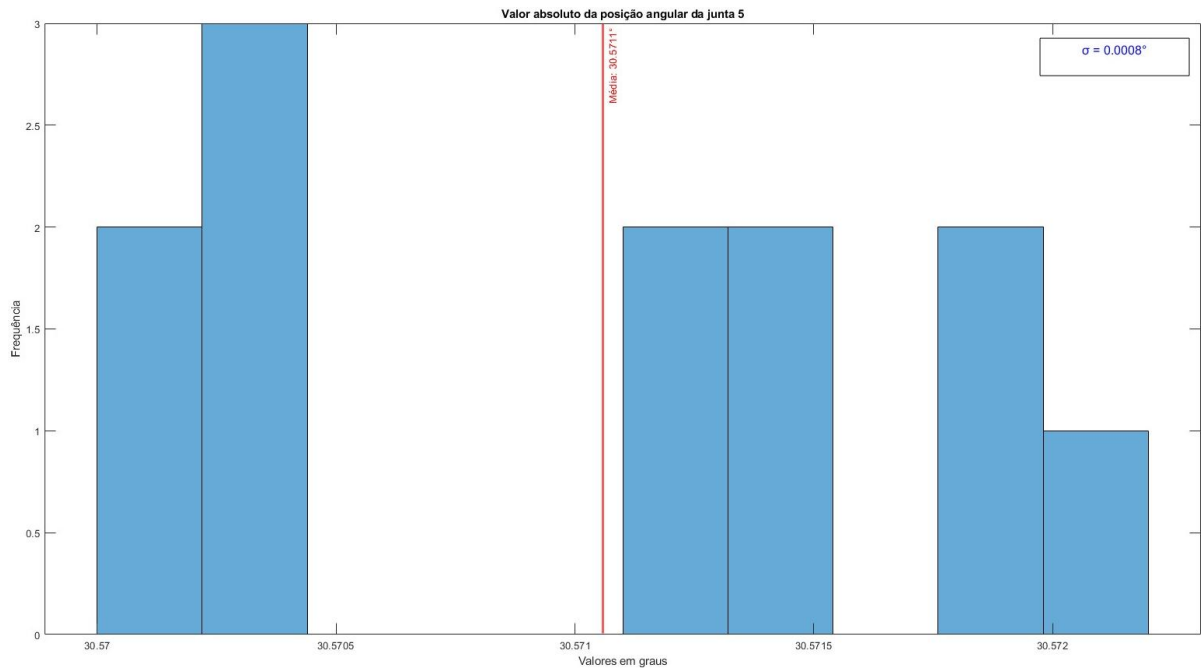


Figura D.14 – Valor absoluto da rotação da Junta 5 - Experimento 2.

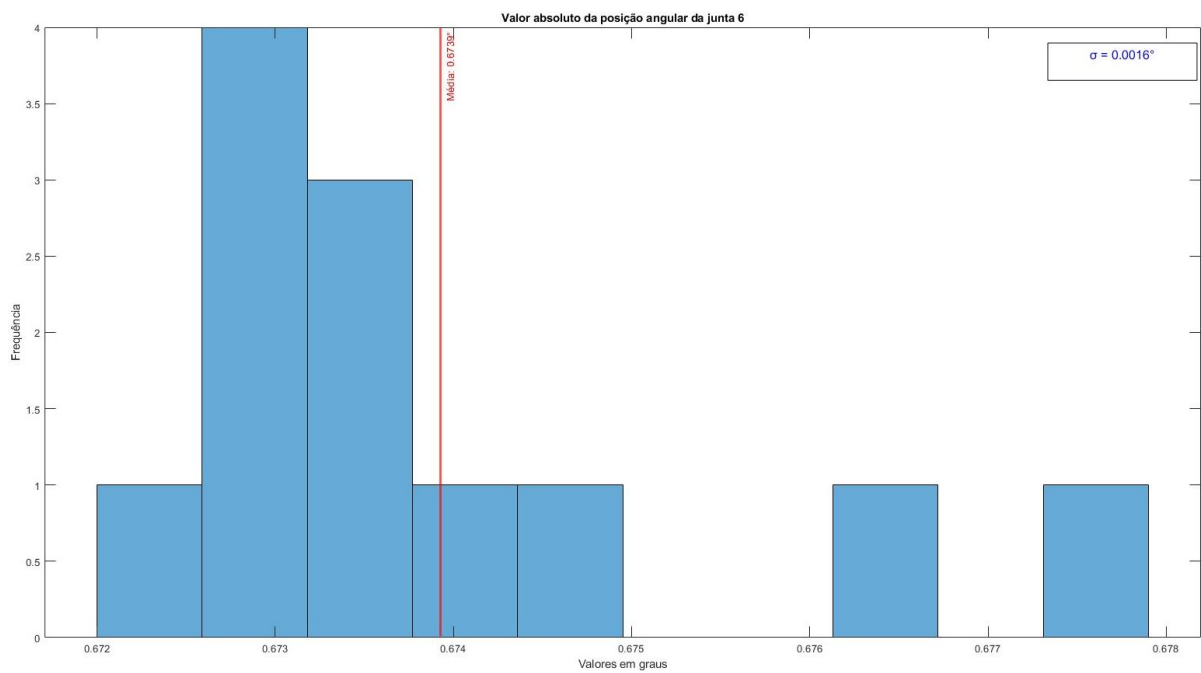


Figura D.15 – Valor absoluto da rotação da Junta 6 - Experimento 2.

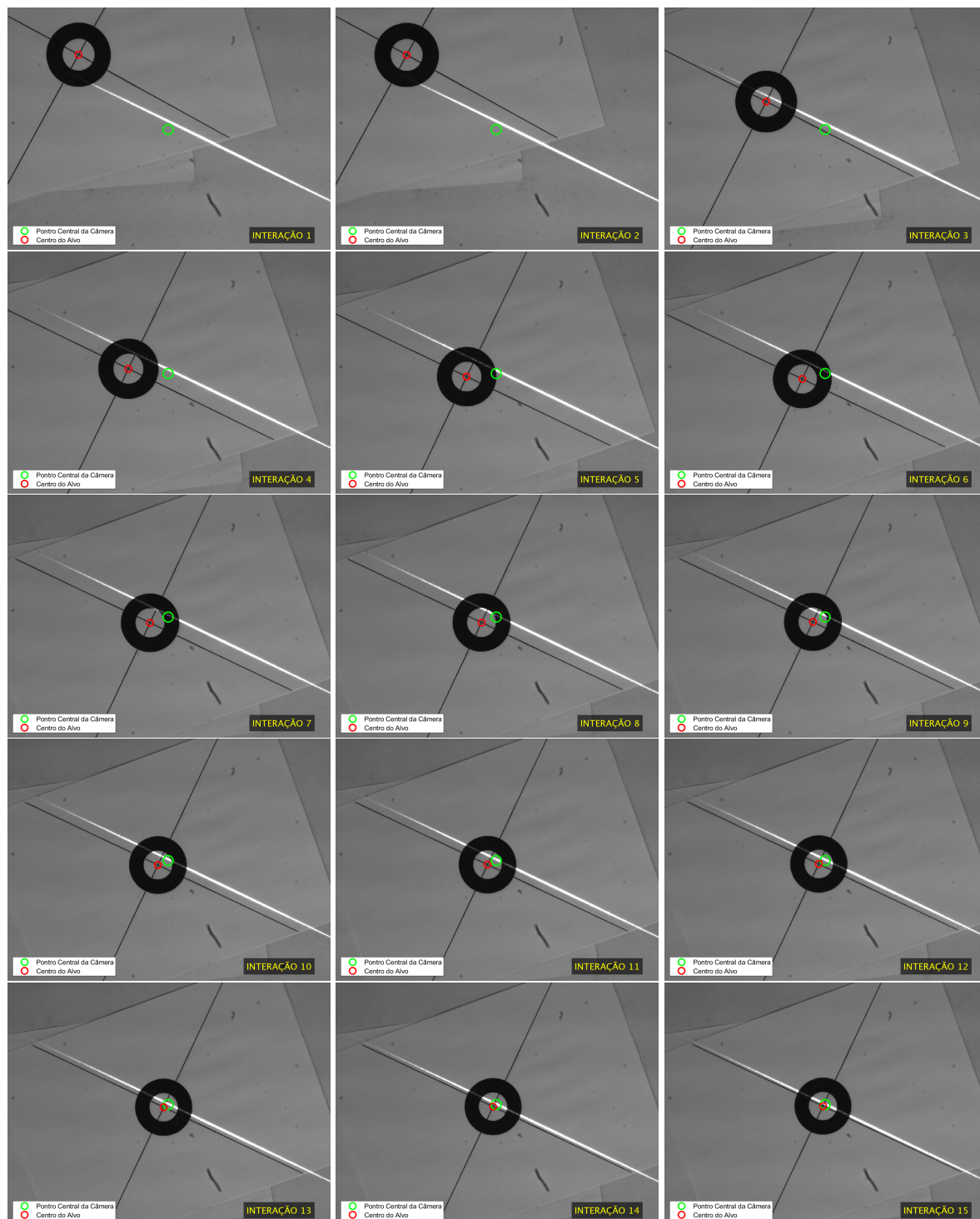


Figura D.16 – Imagens capturadas pela câmera no decorrer do movimento (1ª a 15ª interação) - Experimento 3.

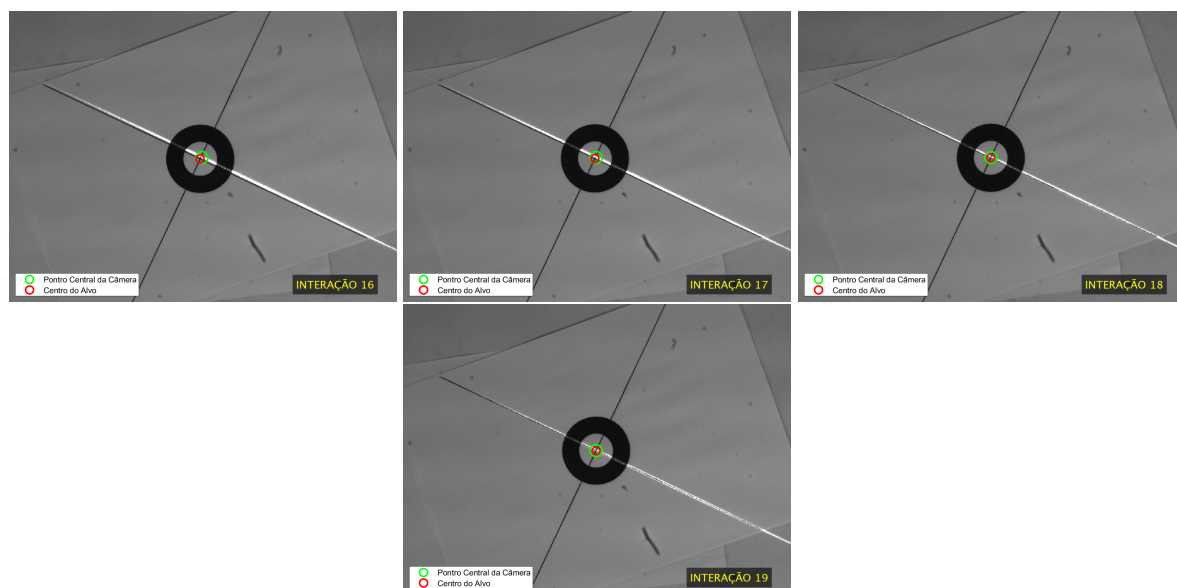


Figura D.17 – Imagens capturadas pela câmera no decorrer do movimento (16ª a 19ª interação) - Experimento 3.

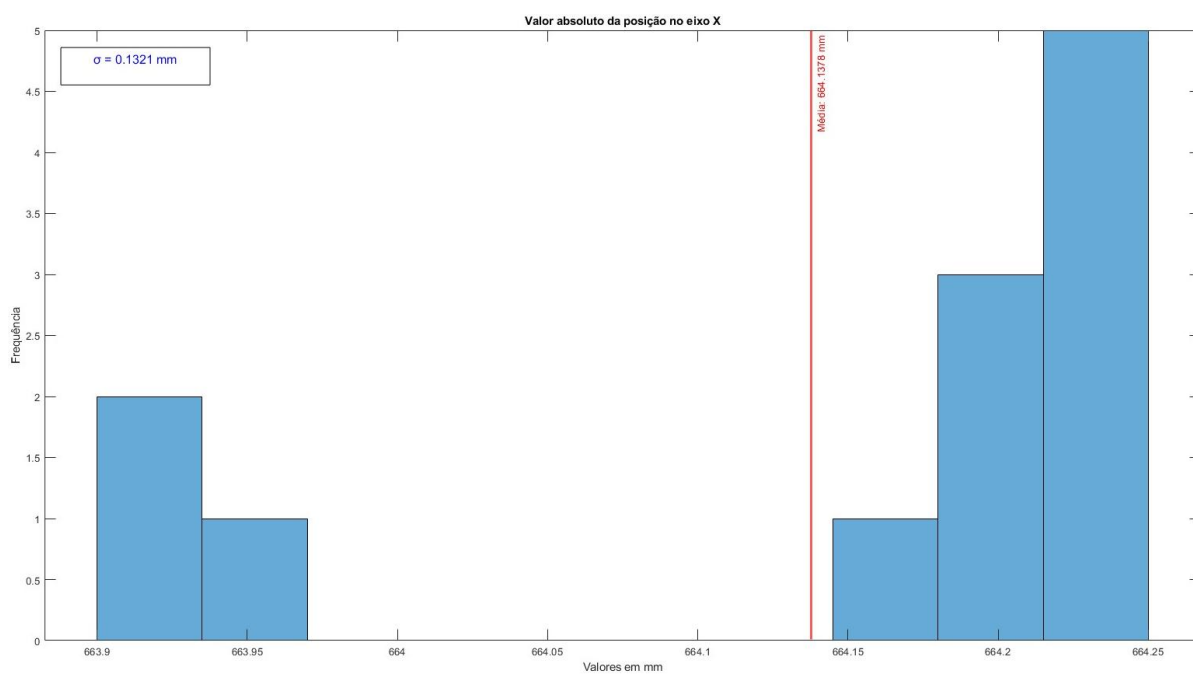


Figura D.18 – Valor absoluto da posição no eixo x - Experimento 3.

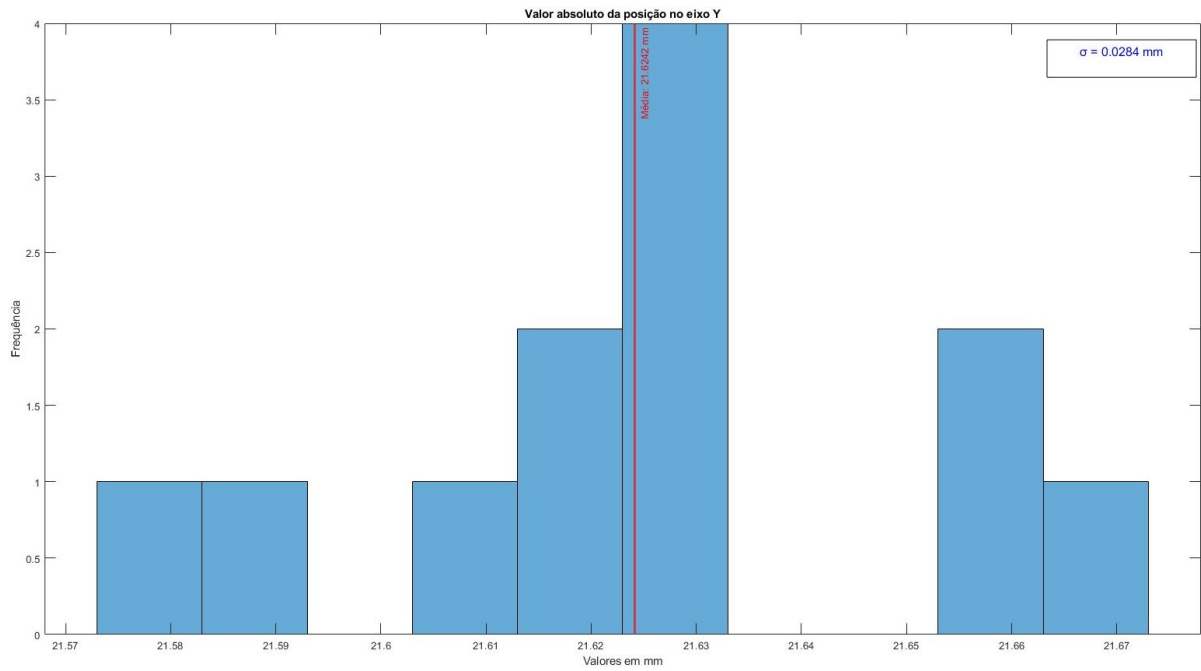


Figura D.19 – Valor absoluto da posição no eixo y - Experimento 3.

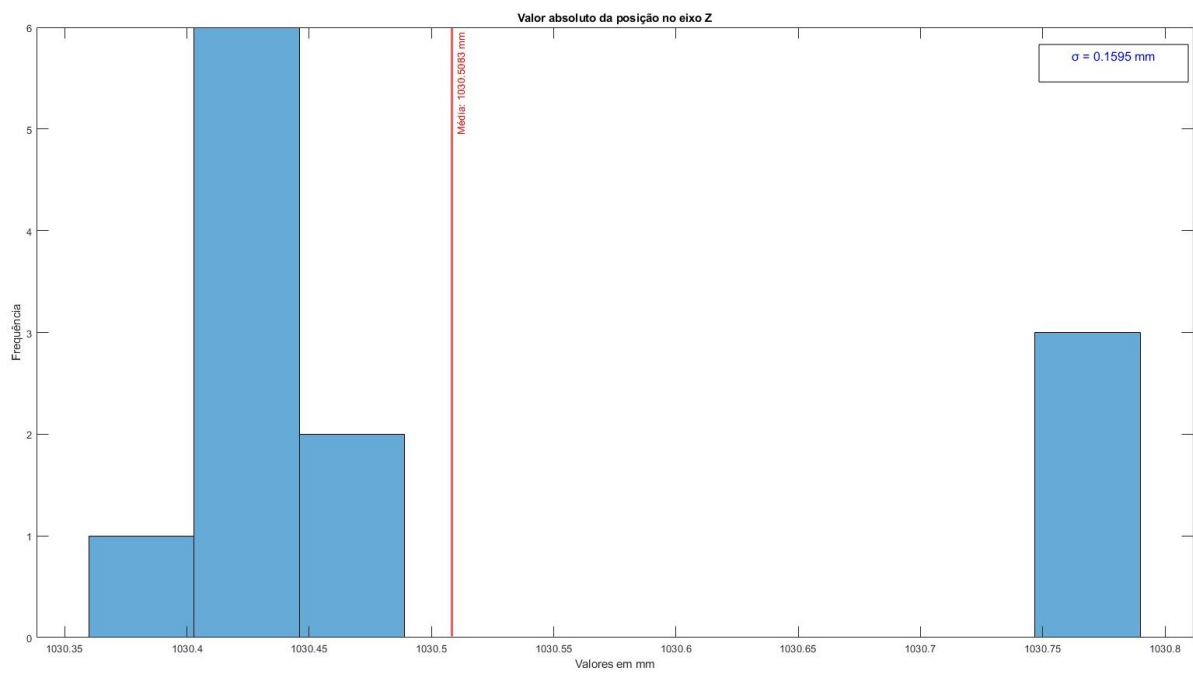


Figura D.20 – Valor absoluto da posição no eixo z - Experimento 3.

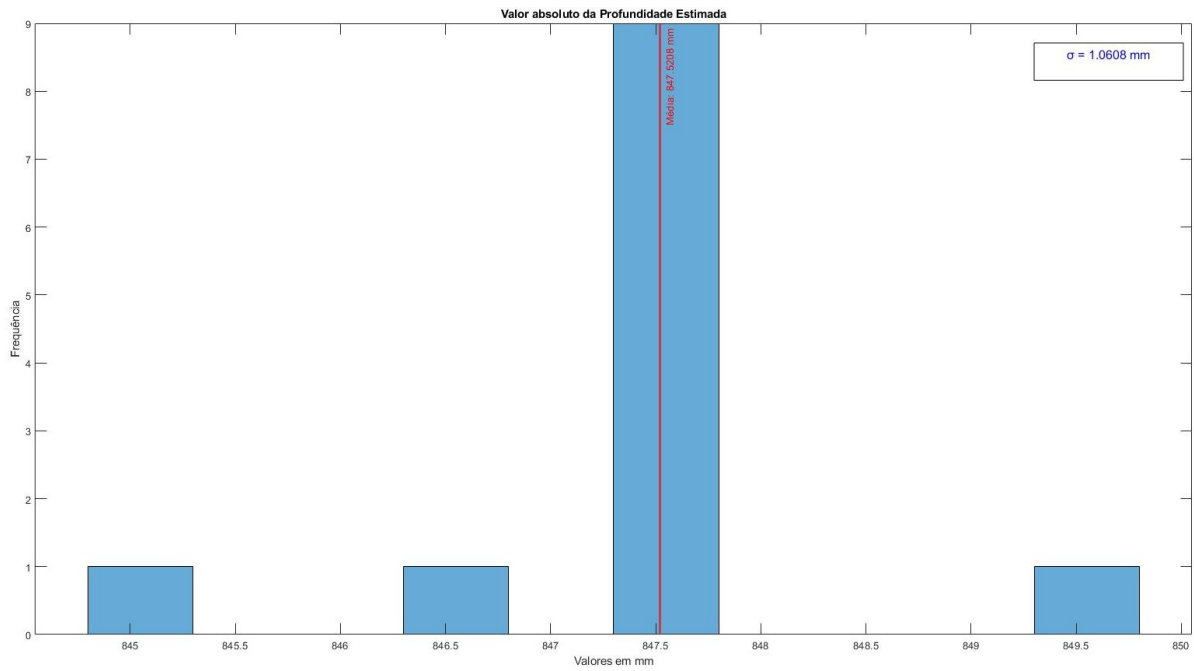


Figura D.21 – Valor absoluto da Profundidade Estimada - Experimento 3.

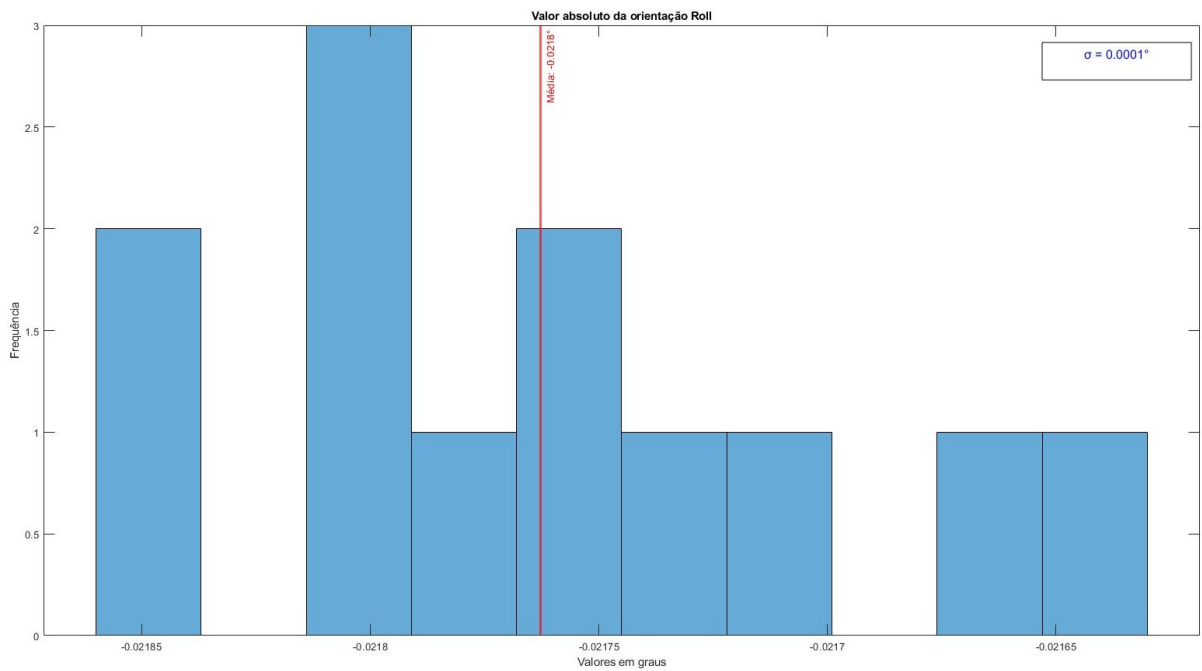


Figura D.22 – Valor absoluto da orientação Roll - Experimento 3.

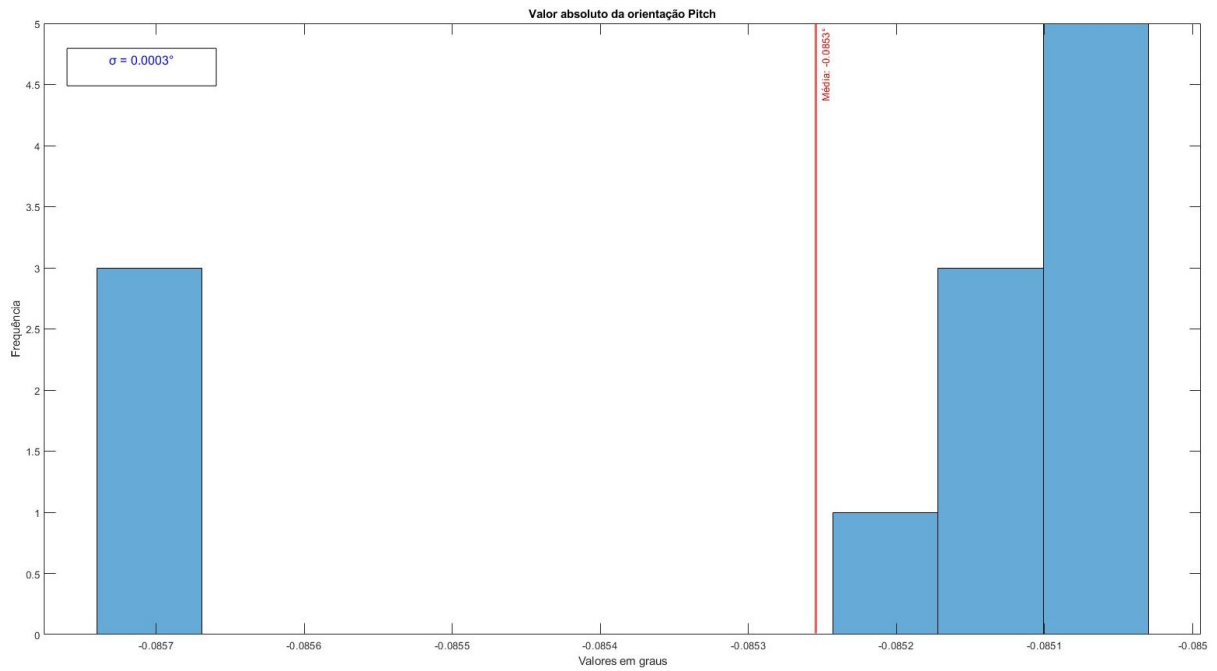


Figura D.23 – Valor absoluto da orientação Pitch - Experimento 3.

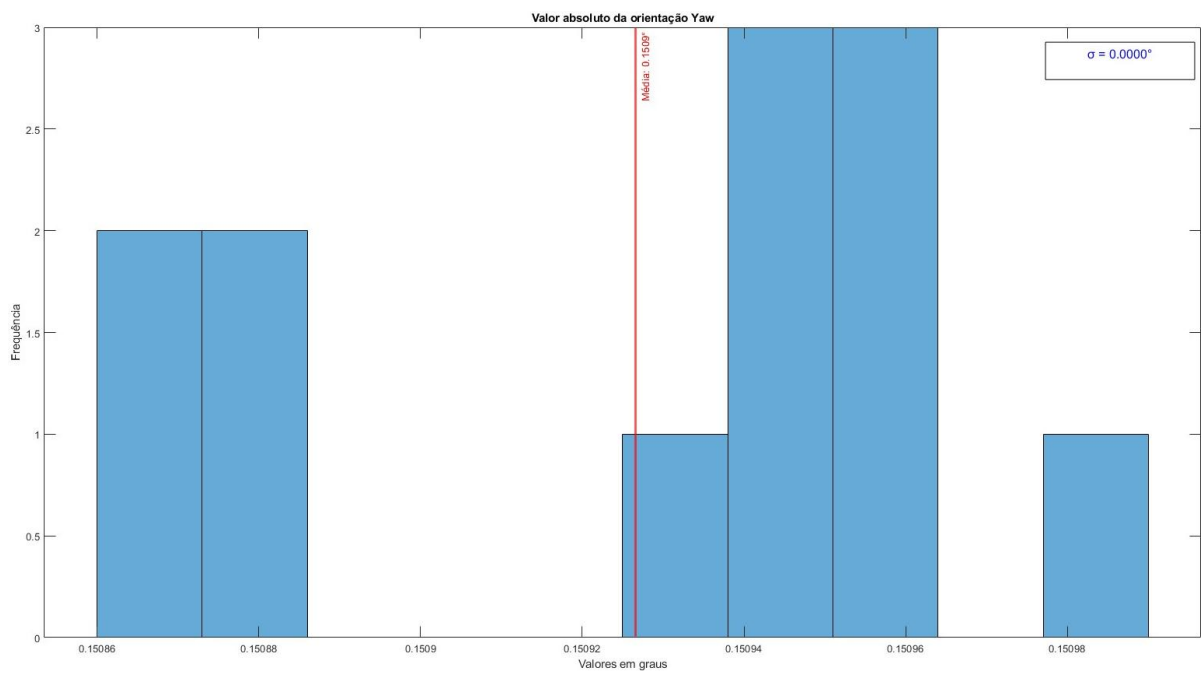


Figura D.24 – Valor absoluto da orientação Yaw - Experimento 3.

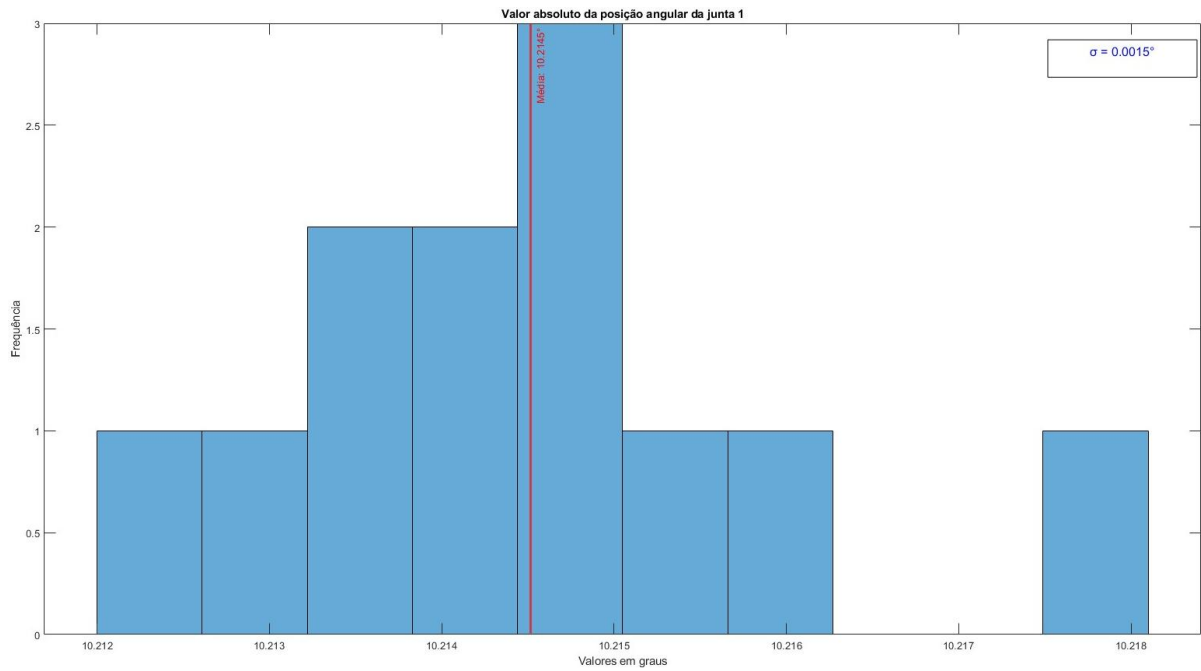


Figura D.25 – Valor absoluto da rotação da Junta 1 - Experimento 3.

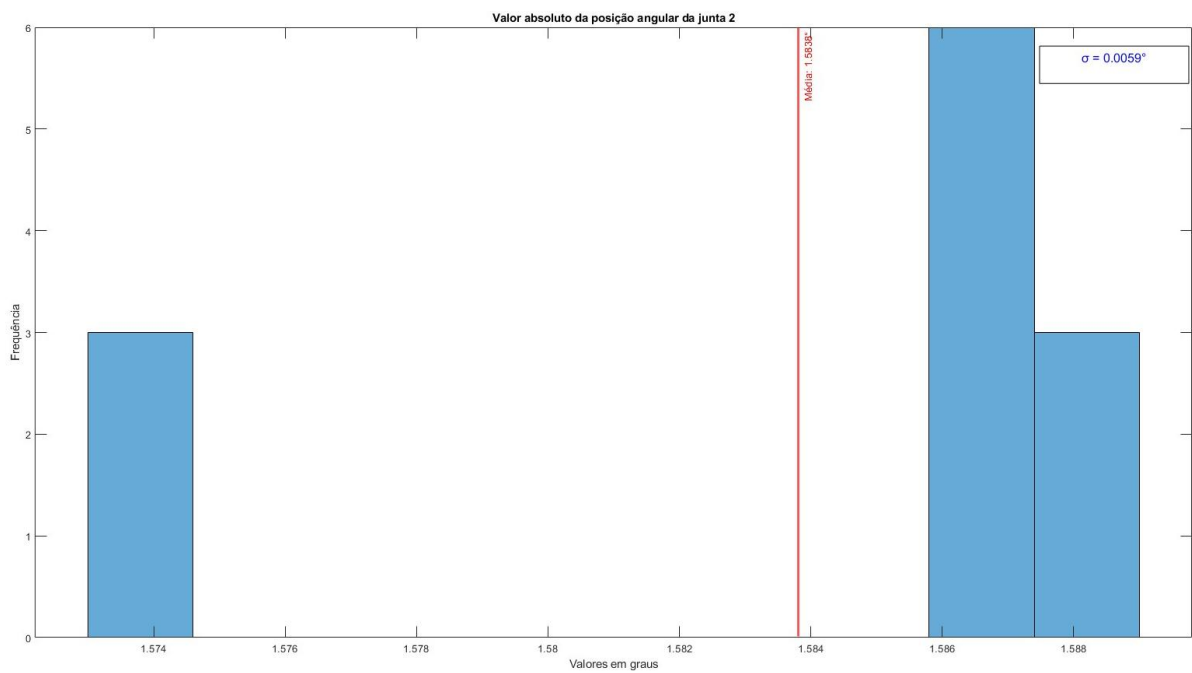


Figura D.26 – Valor absoluto da rotação da Junta 2 - Experimento 3.

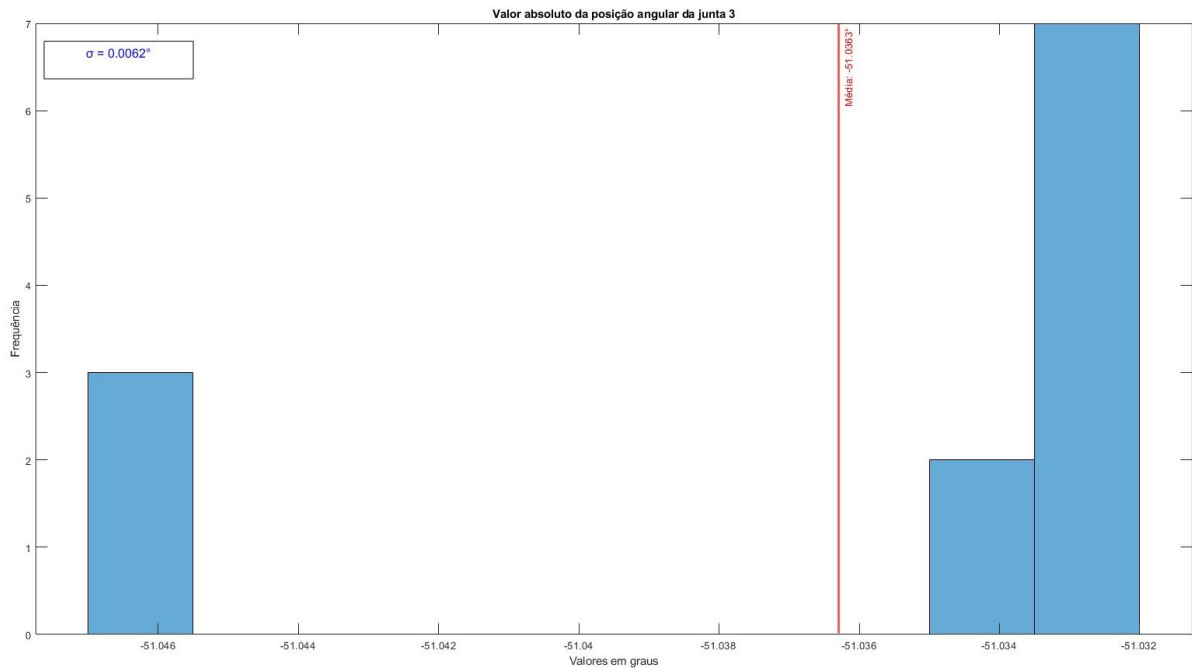


Figura D.27 – Valor absoluto da rotação da Junta 3 - Experimento 3.

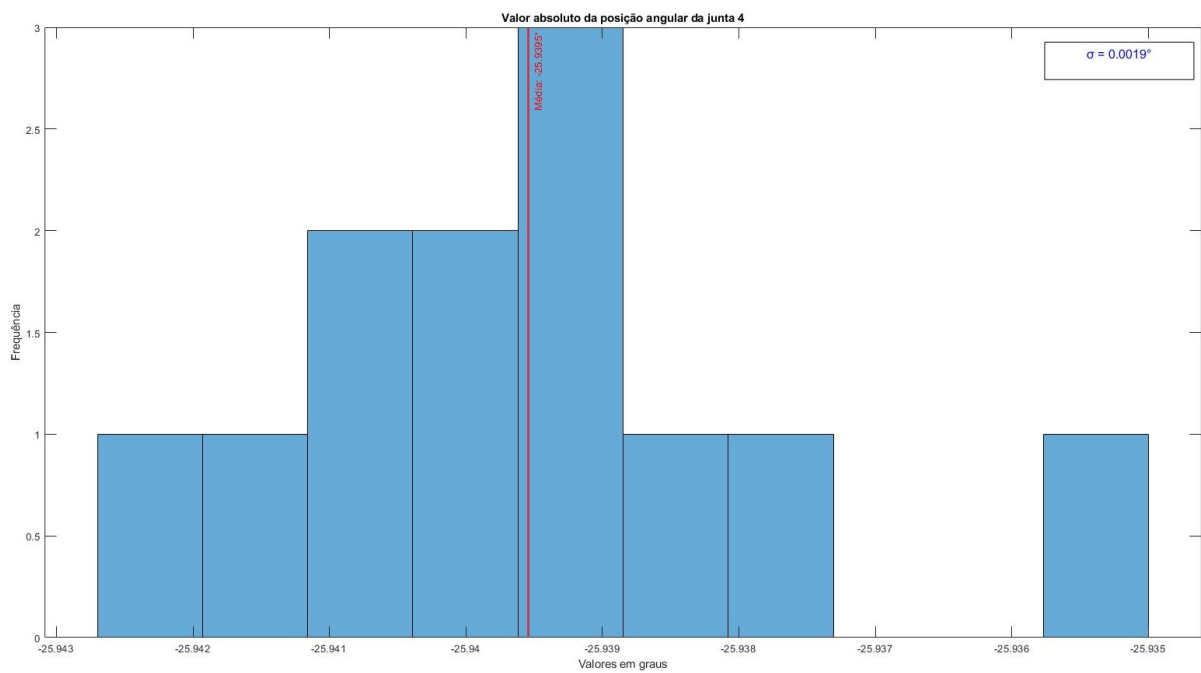


Figura D.28 – Valor absoluto da rotação da Junta 4 - Experimento 3.

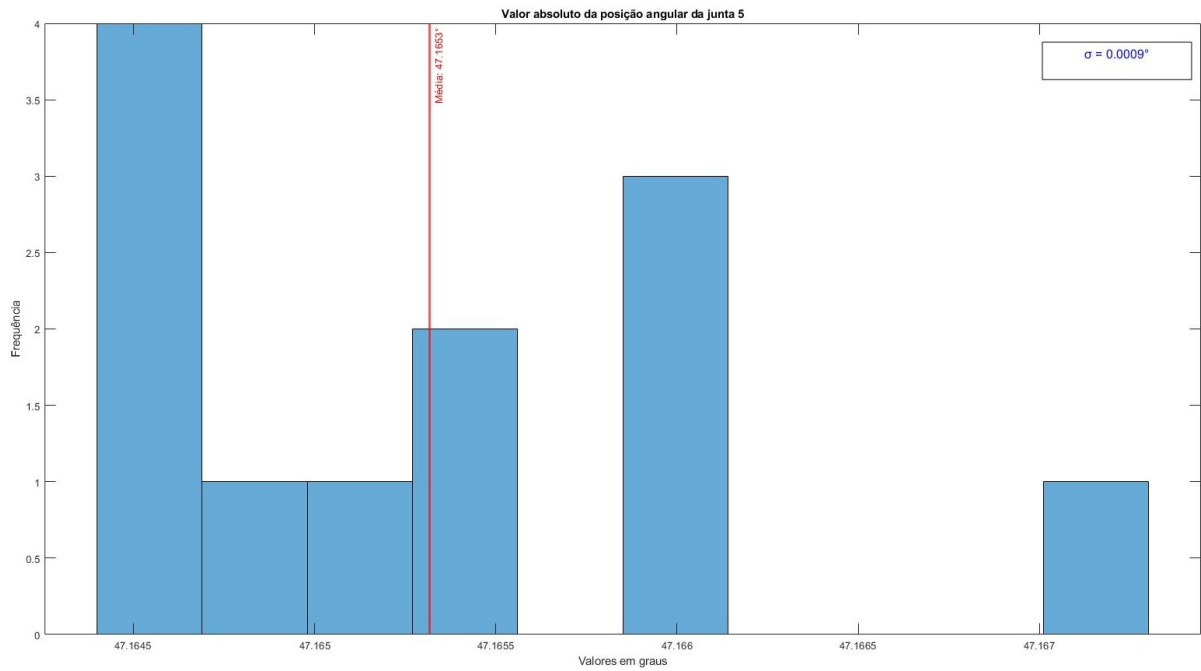


Figura D.29 – Valor absoluto da rotação da Junta 5 - Experimento 3.

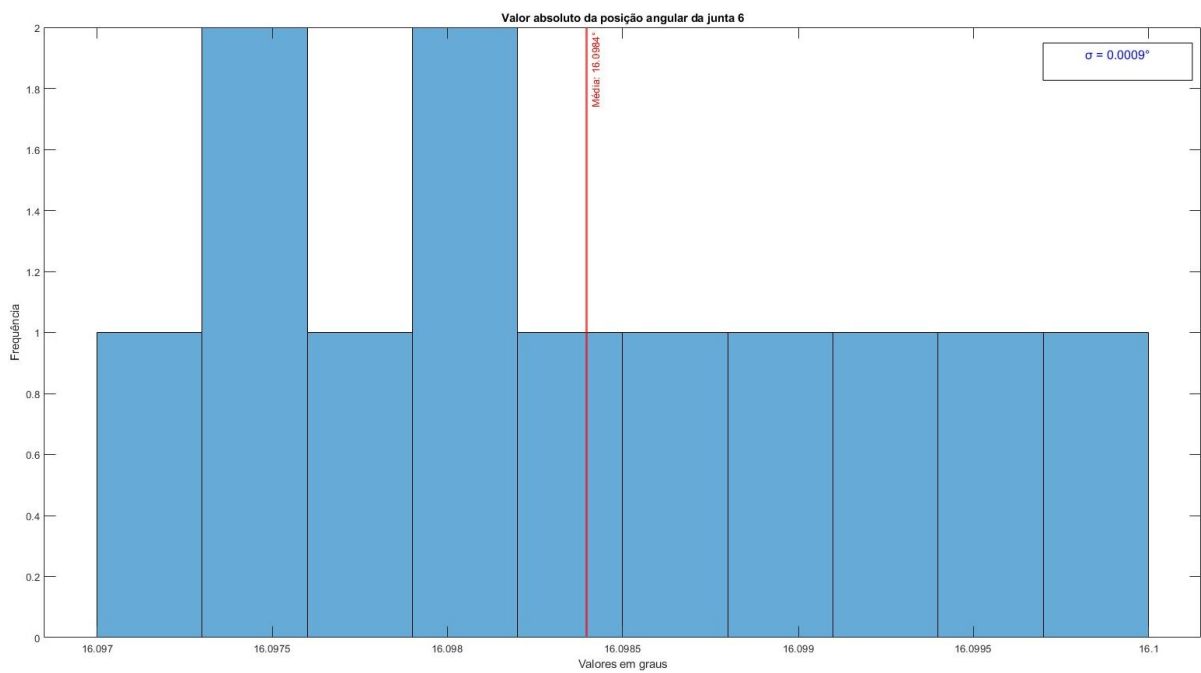


Figura D.30 – Valor absoluto da rotação da Junta 6 - Experimento 3.

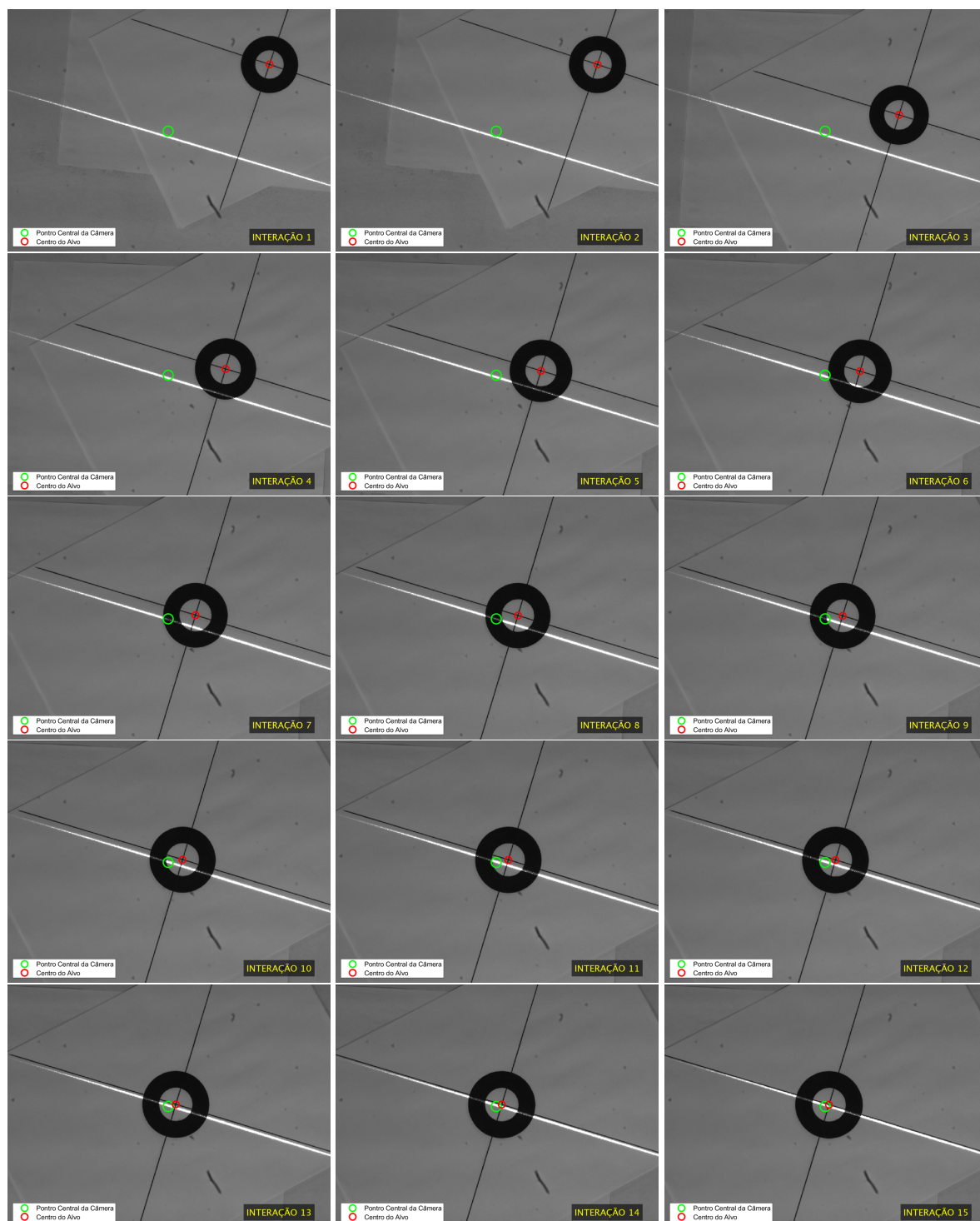


Figura D.31 – Imagens capturadas pela câmera no decorrer do movimento (1ª a 15ª interação) - Experimento 4.

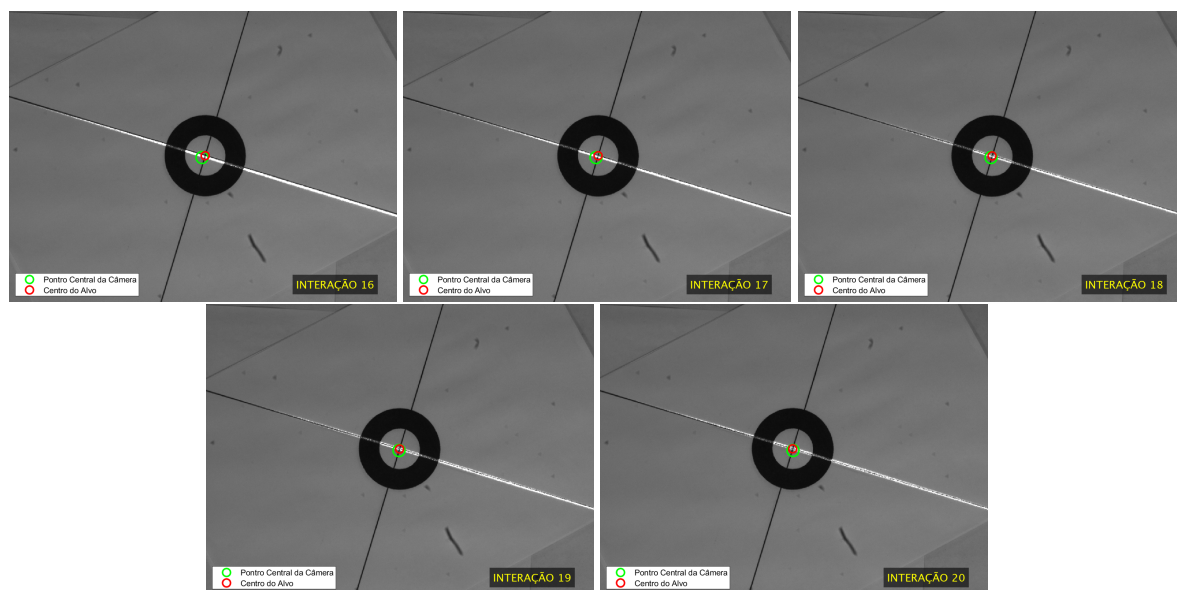


Figura D.32 – Imagens capturadas pela câmera no decorrer do movimento (16^a a 20^a interação) - Experimento 4.

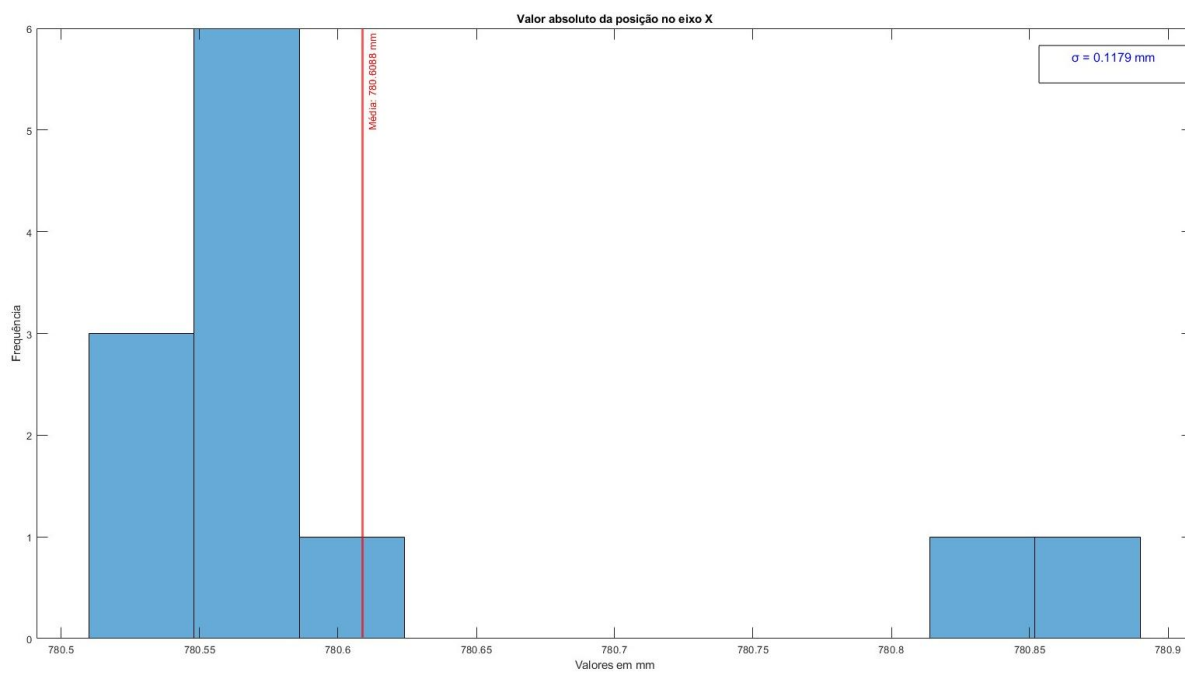


Figura D.33 – Valor absoluto da posição no eixo x - Experimento 4.

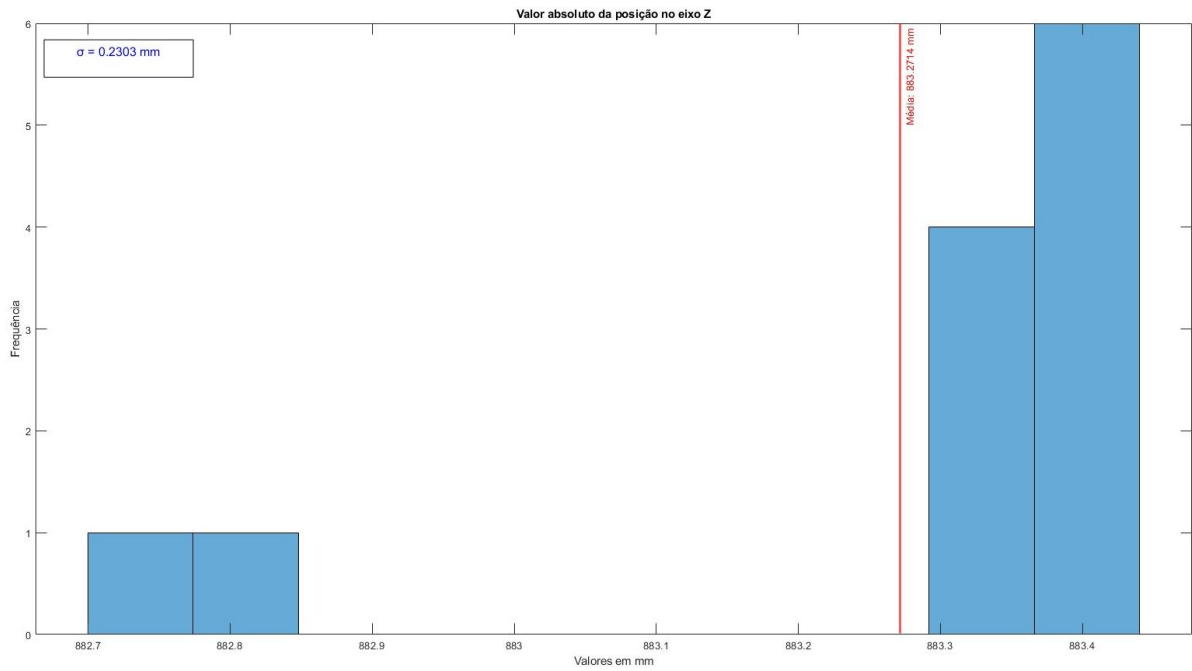


Figura D.34 – Valor absoluto da posição no eixo z - Experimento 4.

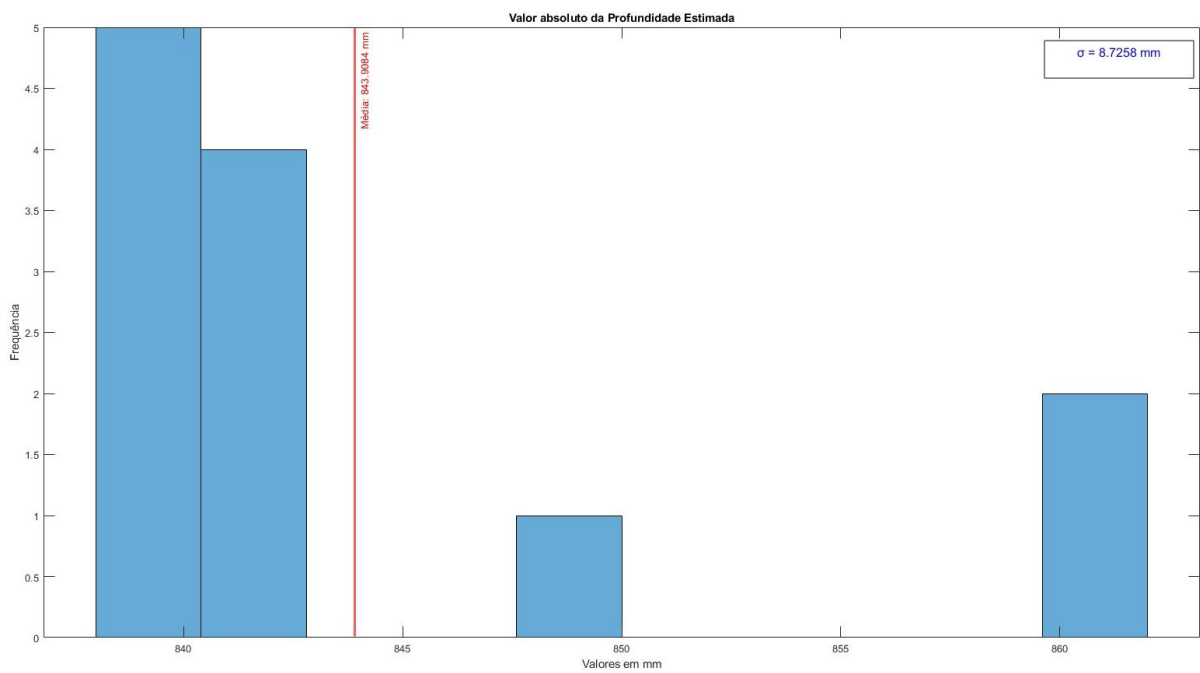


Figura D.35 – Valor absoluto da Profundidade Estimada - Experimento 4.

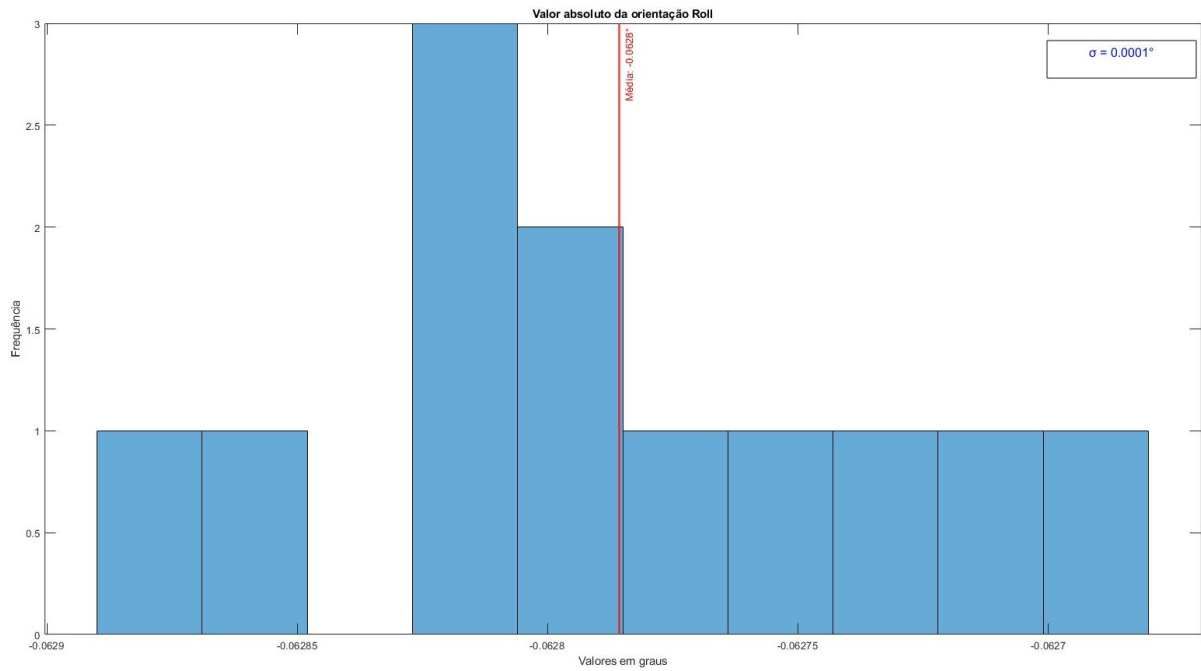


Figura D.36 – Valor absoluto da orientação Roll - Experimento 4.

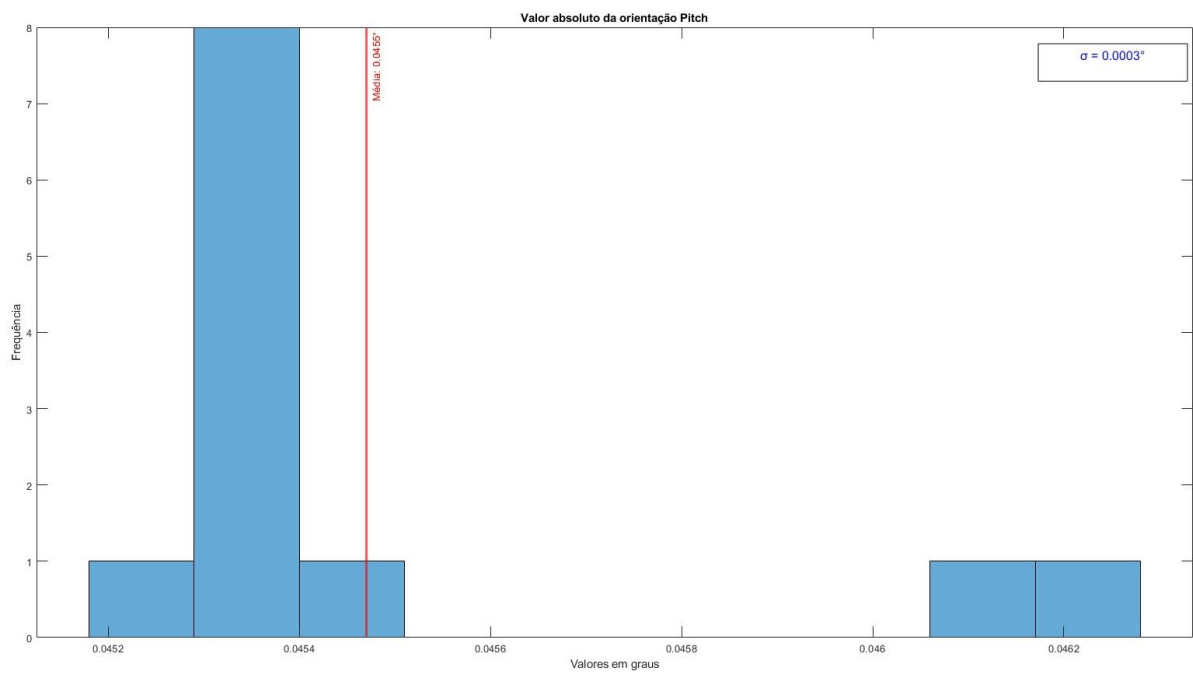


Figura D.37 – Valor absoluto da orientação Pitch - Experimento 4.

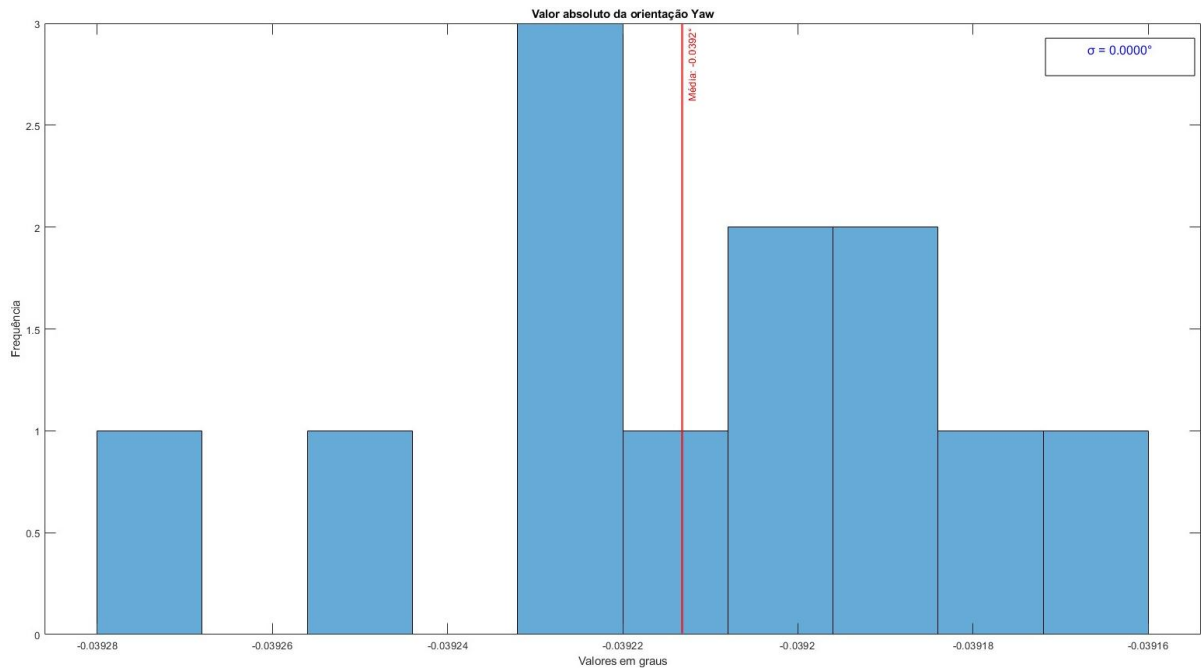


Figura D.38 – Valor absoluto da orientação Yaw - Experimento 4.

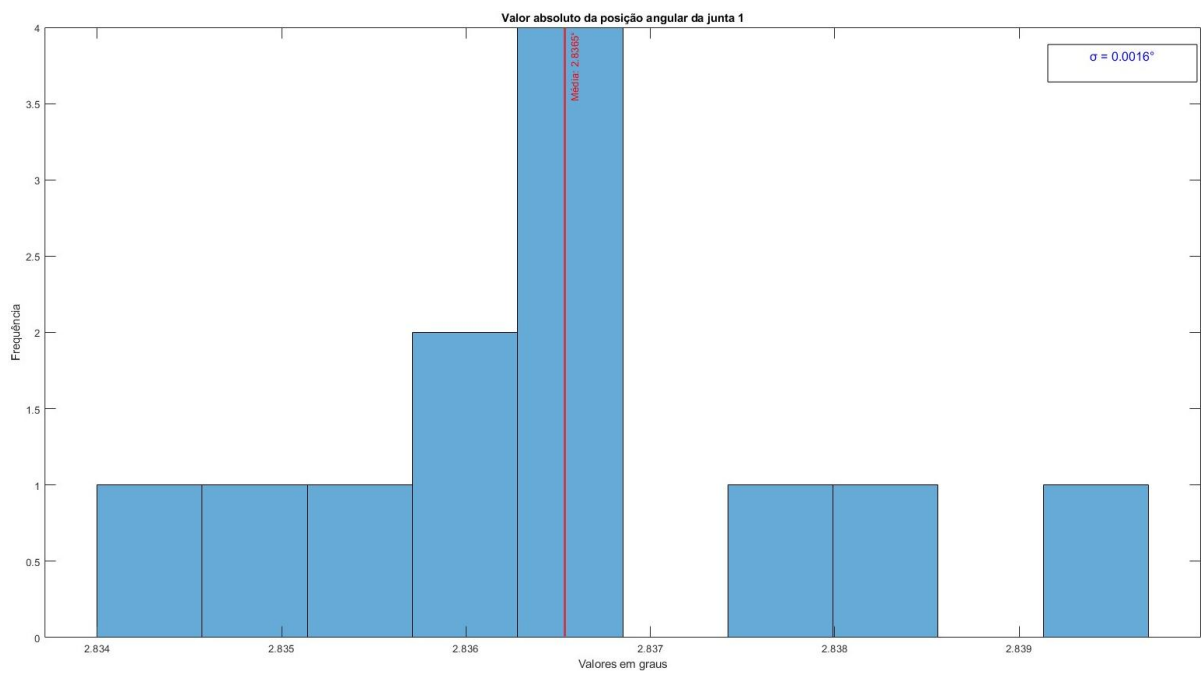


Figura D.39 – Valor absoluto da rotação da Junta 1 - Experimento 4.

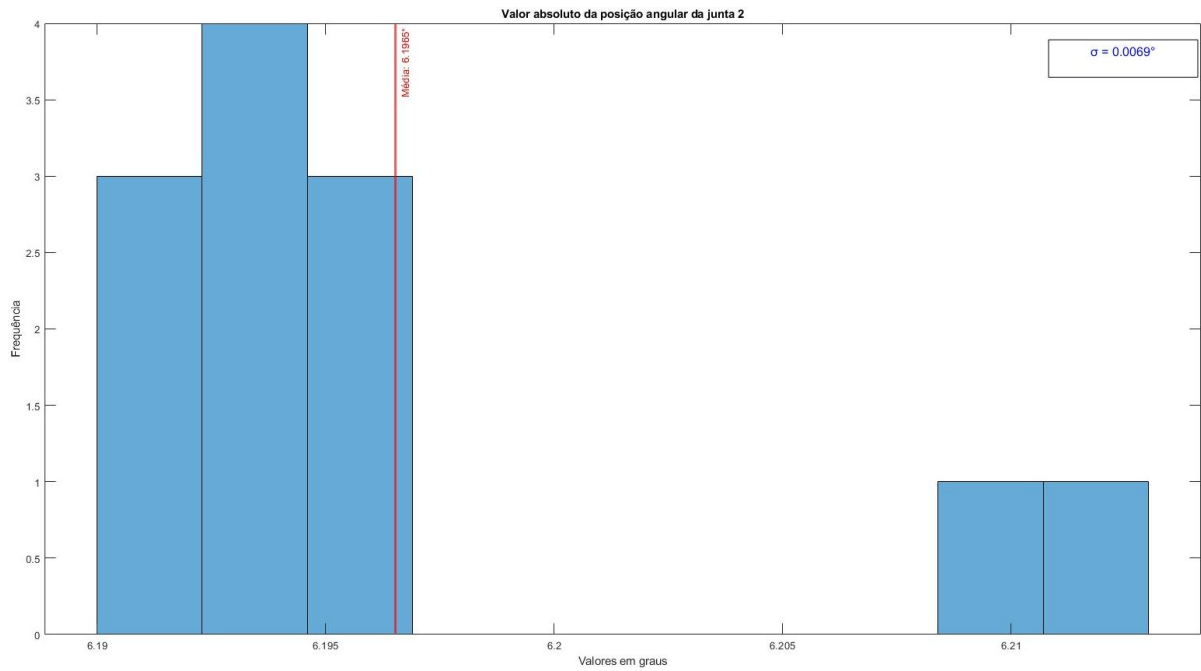


Figura D.40 – Valor absoluto da rotação da Junta 2 - Experimento 4.

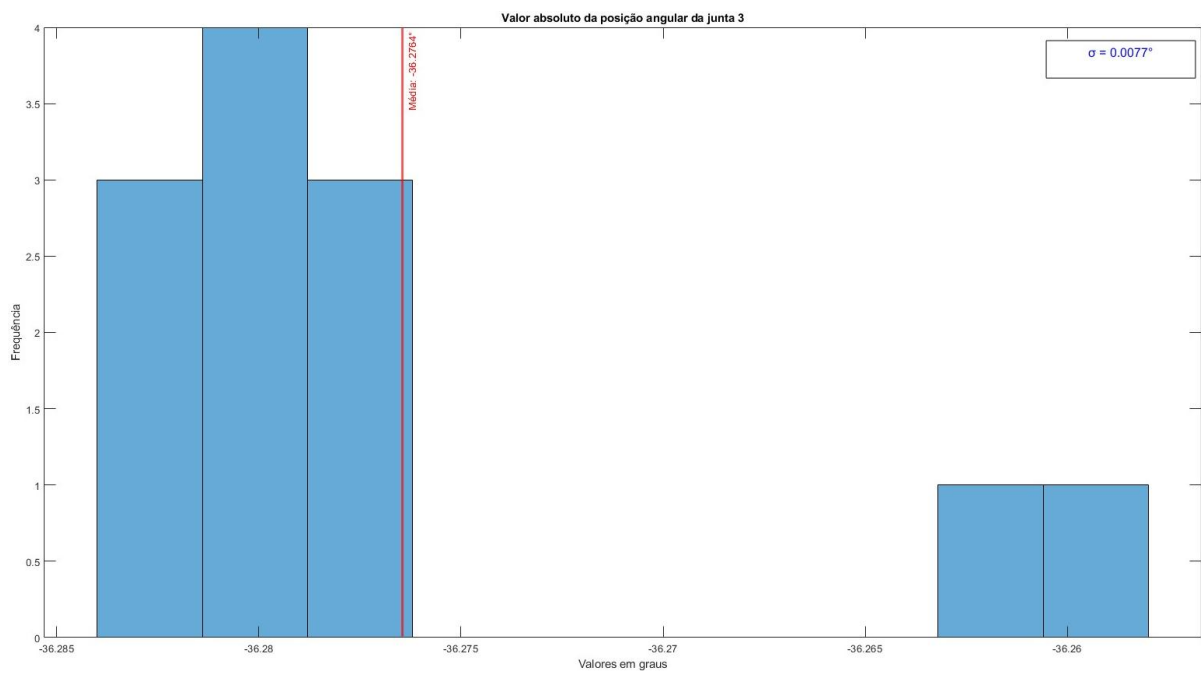


Figura D.41 – Valor absoluto da rotação da Junta 3 - Experimento 3.

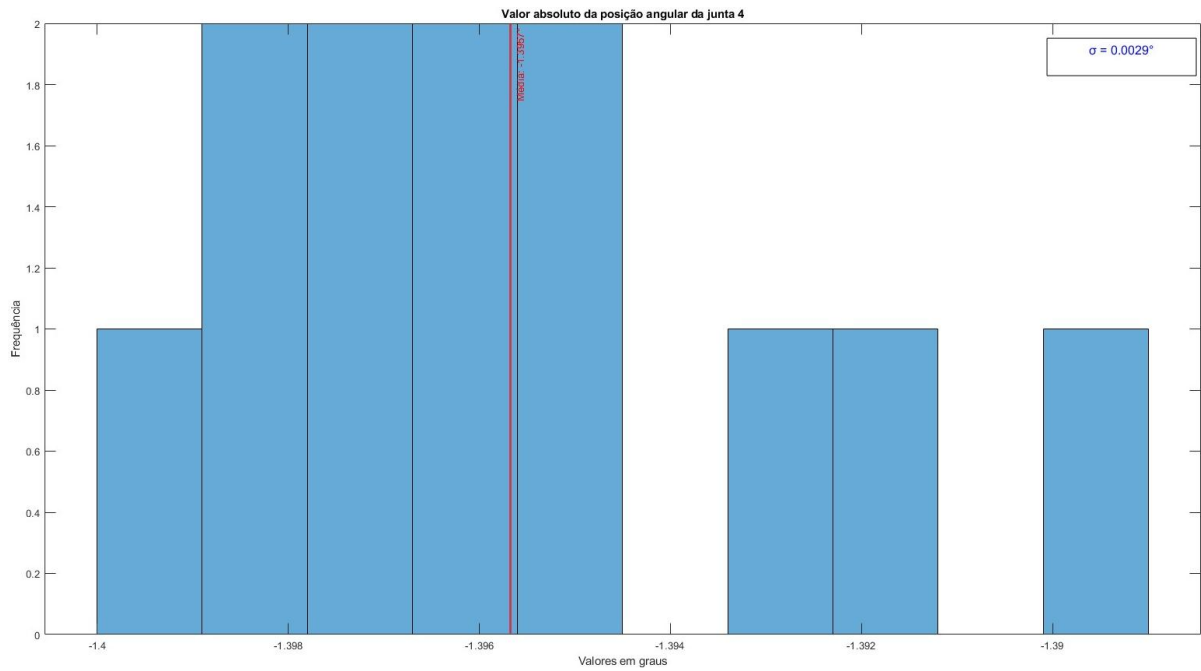


Figura D.42 – Valor absoluto da rotação da Junta 4 - Experimento 4

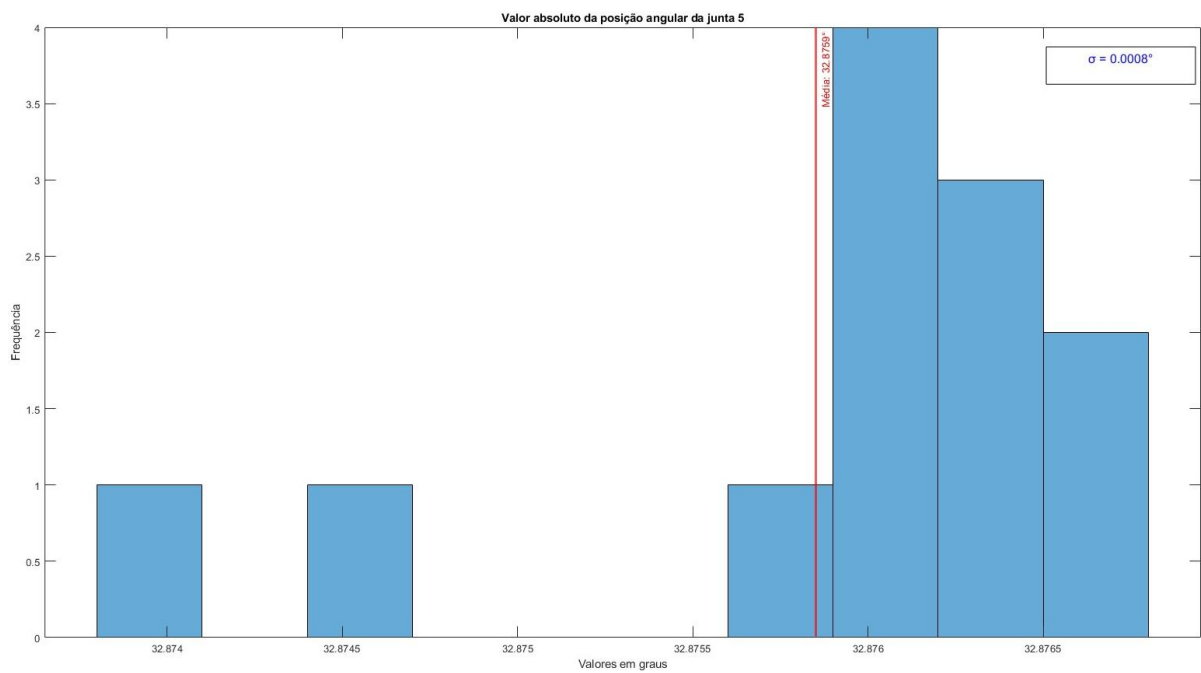


Figura D.43 – Valor absoluto da rotação da Junta 5 - Experimento 4.

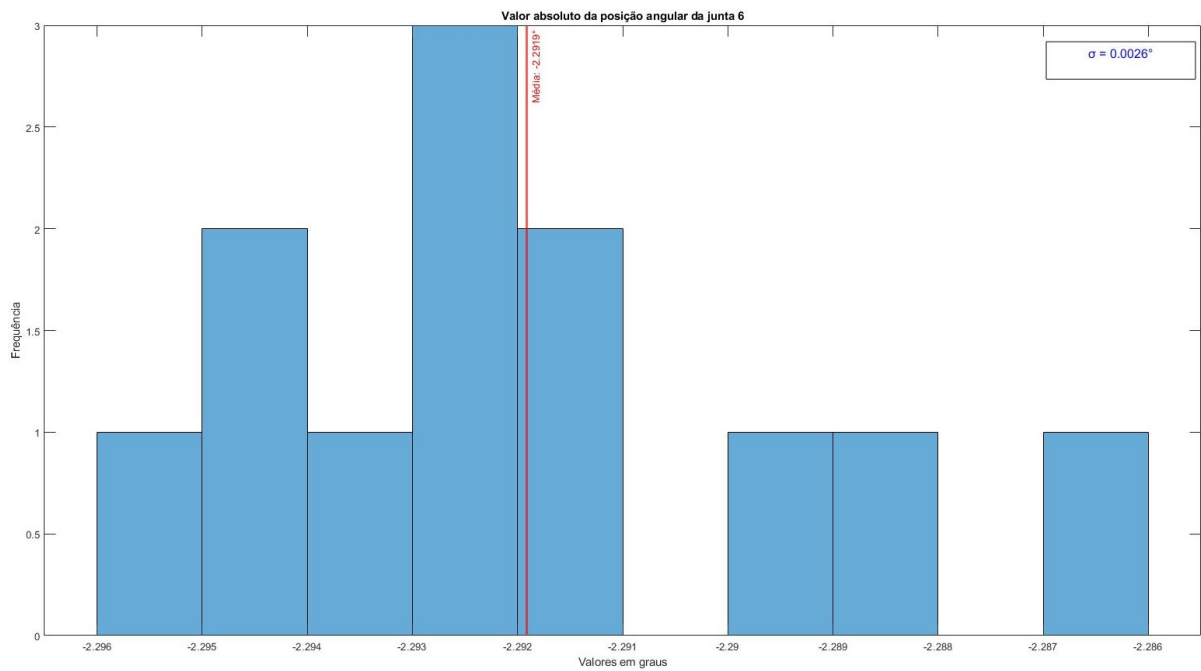


Figura D.44 – Valor absoluto da rotação da Junta 6 - Experimento 4.

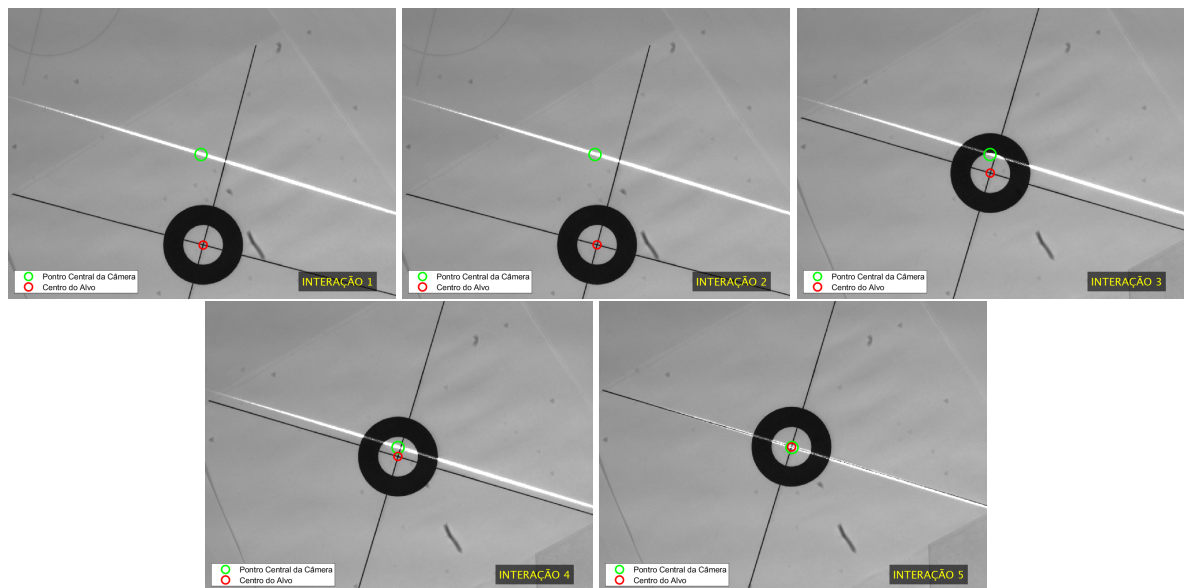


Figura D.45 – Imagens capturadas pela câmera no decorrer do movimento - Experimento 5

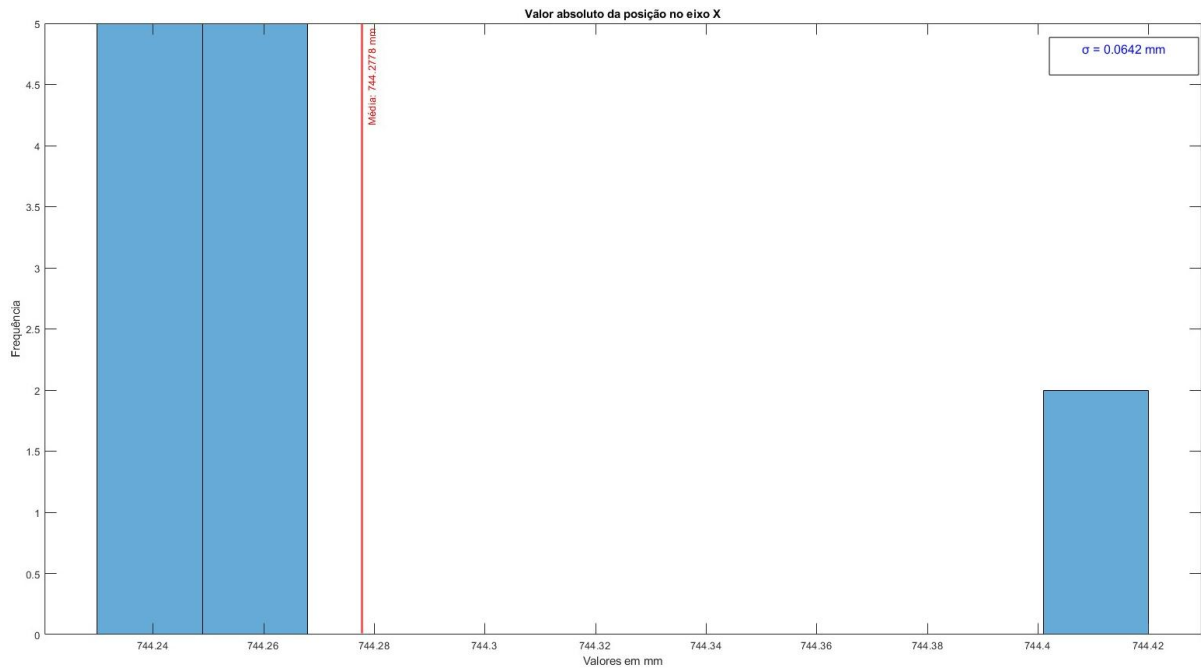


Figura D.46 – Valor absoluto da posição no eixo x - Experimento 5.

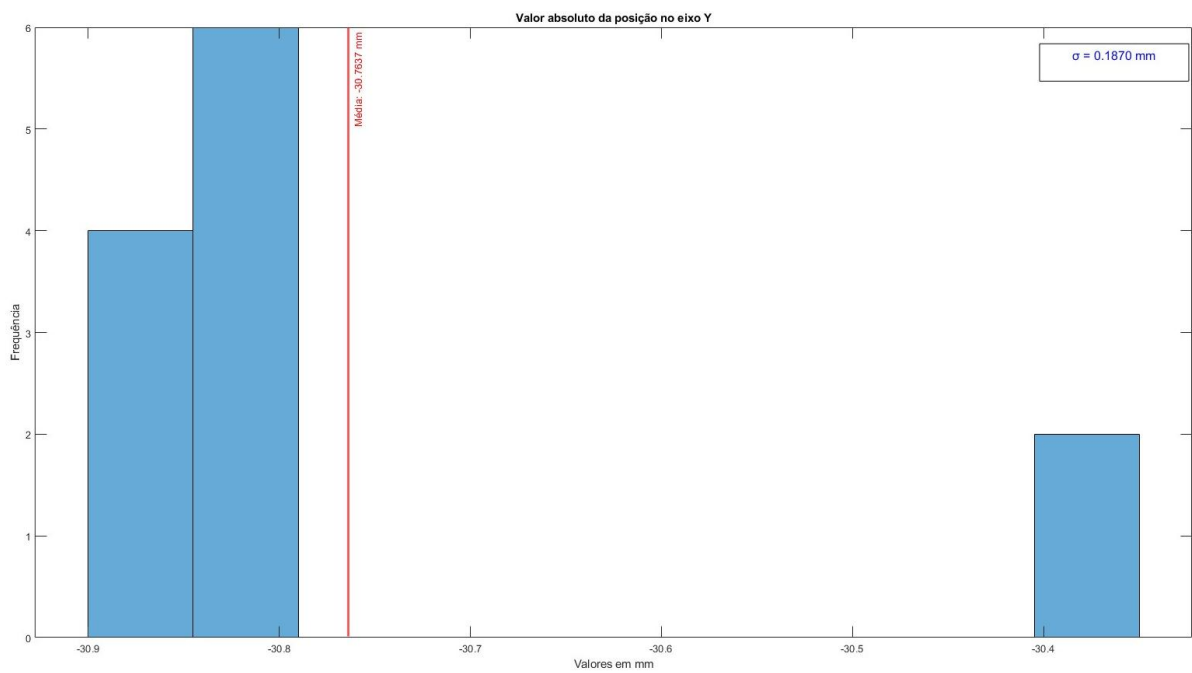


Figura D.47 – Valor absoluto da posição no eixo y - Experimento 5.

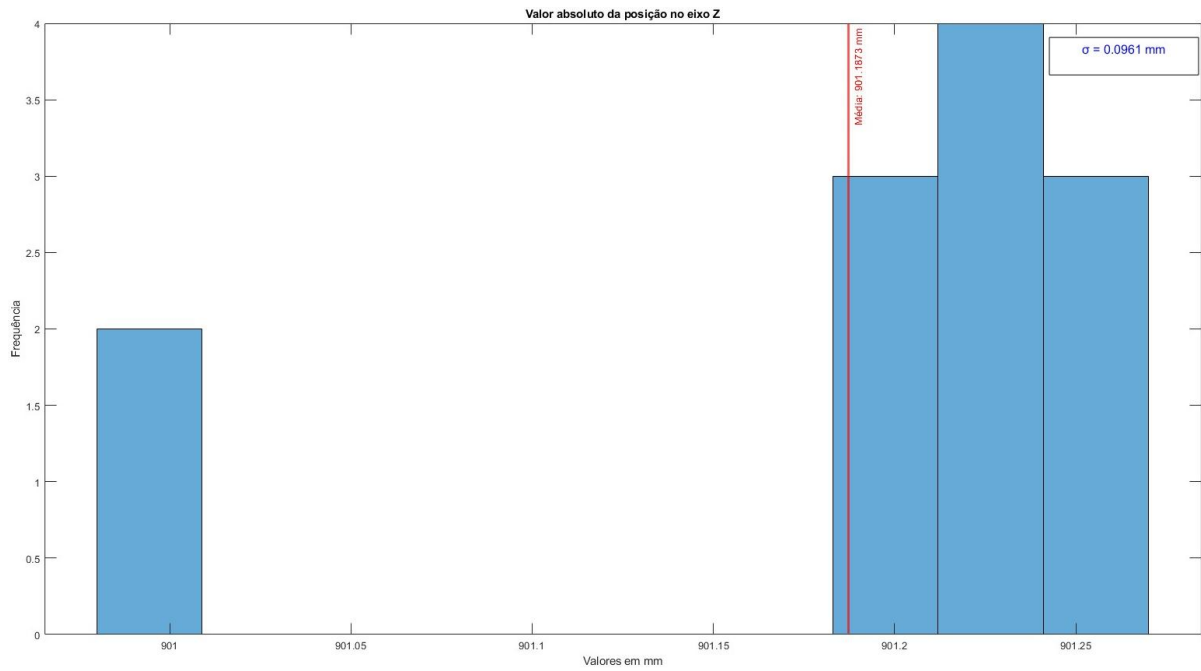


Figura D.48 – Valor absoluto da posição no eixo z - Experimento 5.

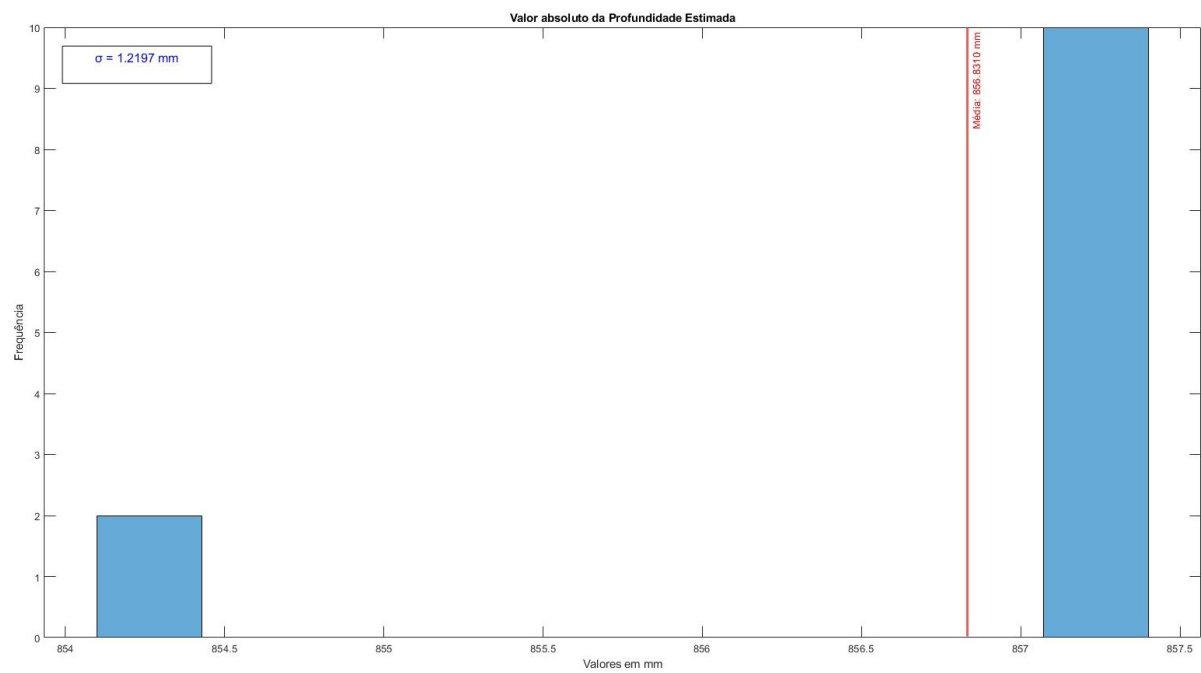


Figura D.49 – Valor absoluto da Profundidade Estimada - Experimento 5.

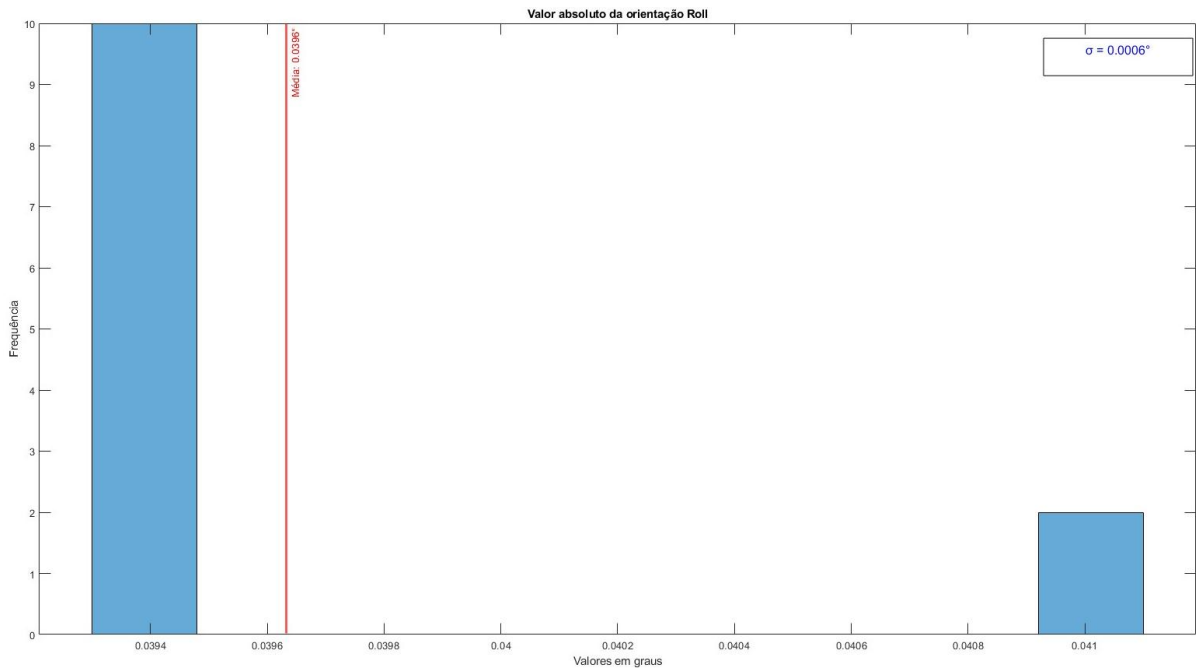


Figura D.50 – Valor absoluto da orientação Roll - Experimento 5.

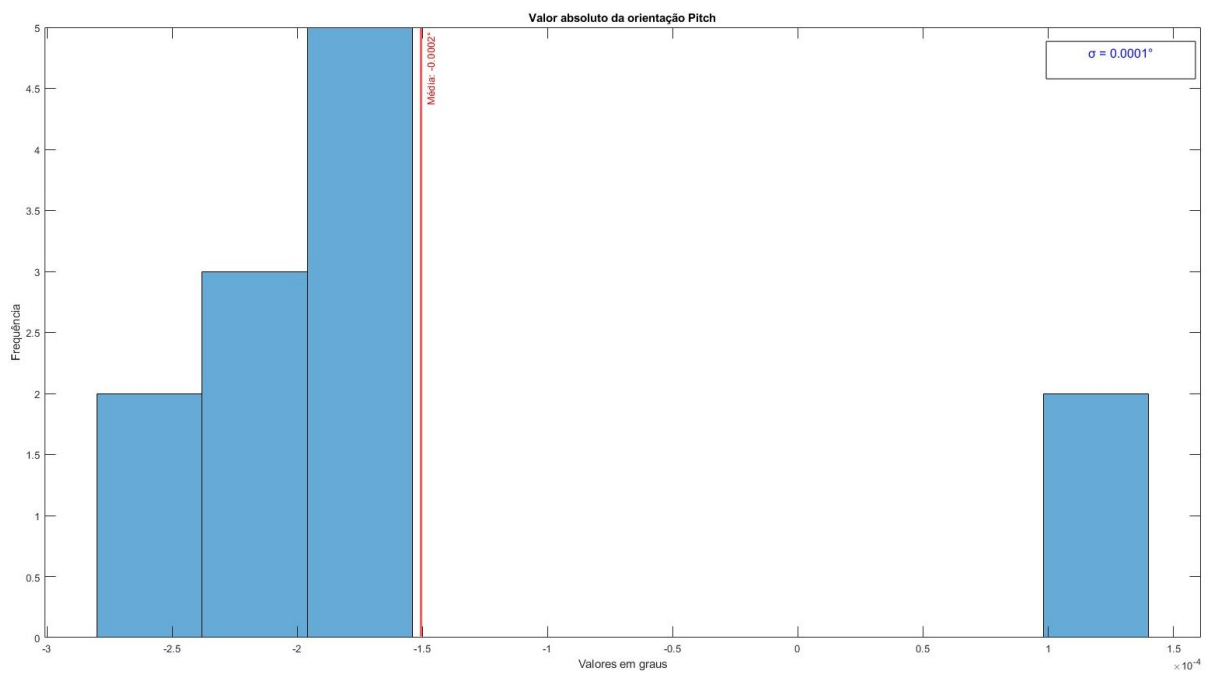


Figura D.51 – Valor absoluto da orientação Pitch - Experimento 5.

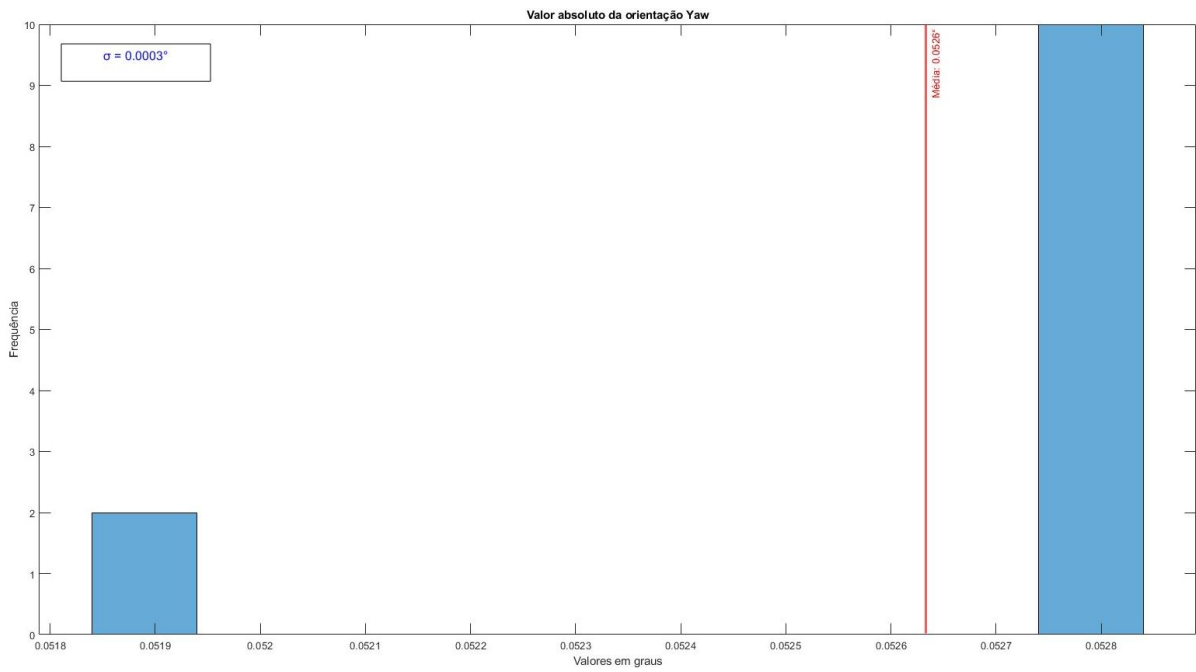


Figura D.52 – Valor absoluto da orientação Yaw - Experimento 5.

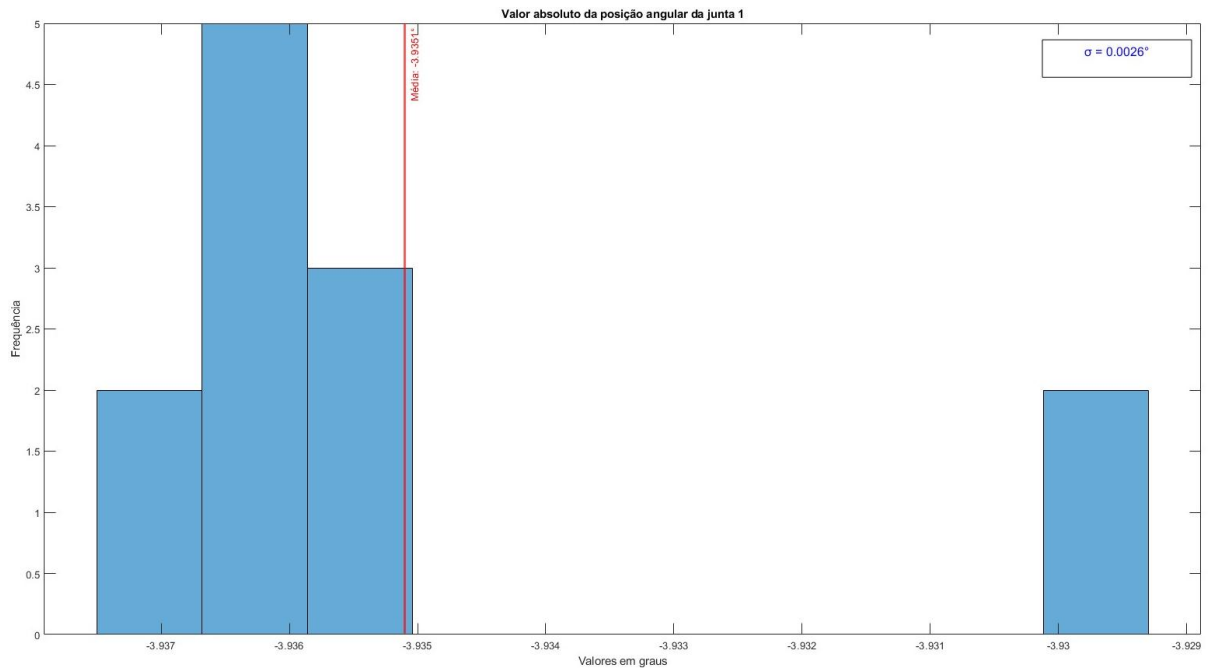


Figura D.53 – Valor absoluto da rotação da Junta 1 - Experimento 5.

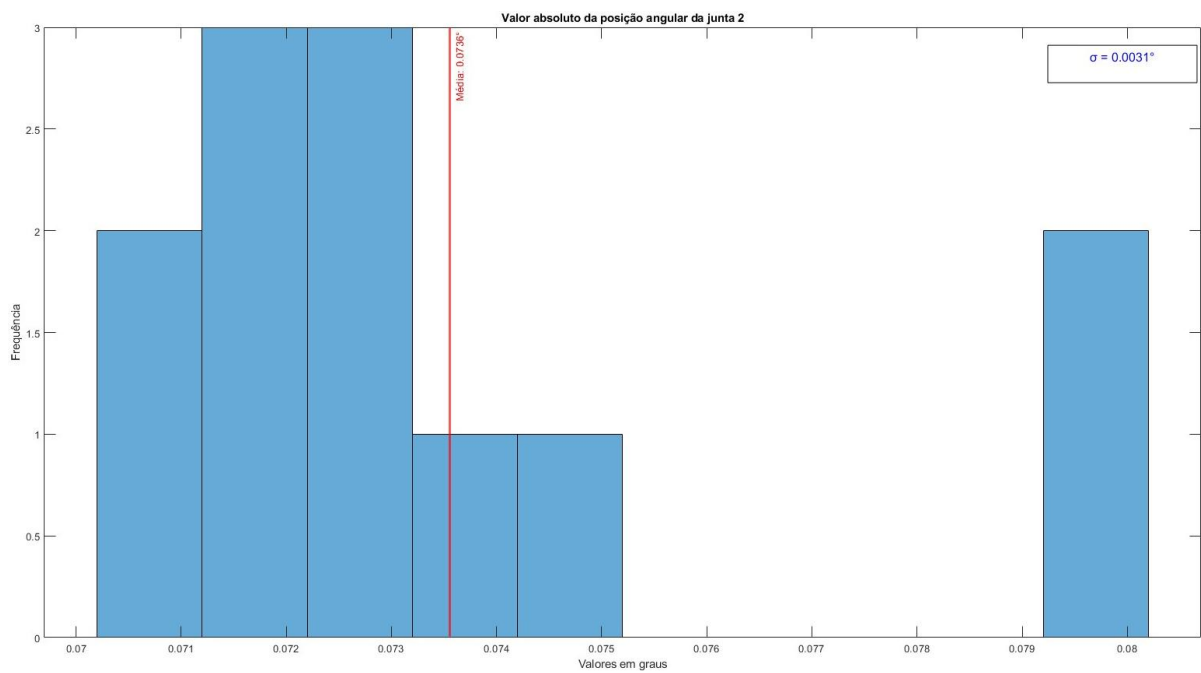


Figura D.54 – Valor absoluto da rotação da Junta 2 - Experimento 5.

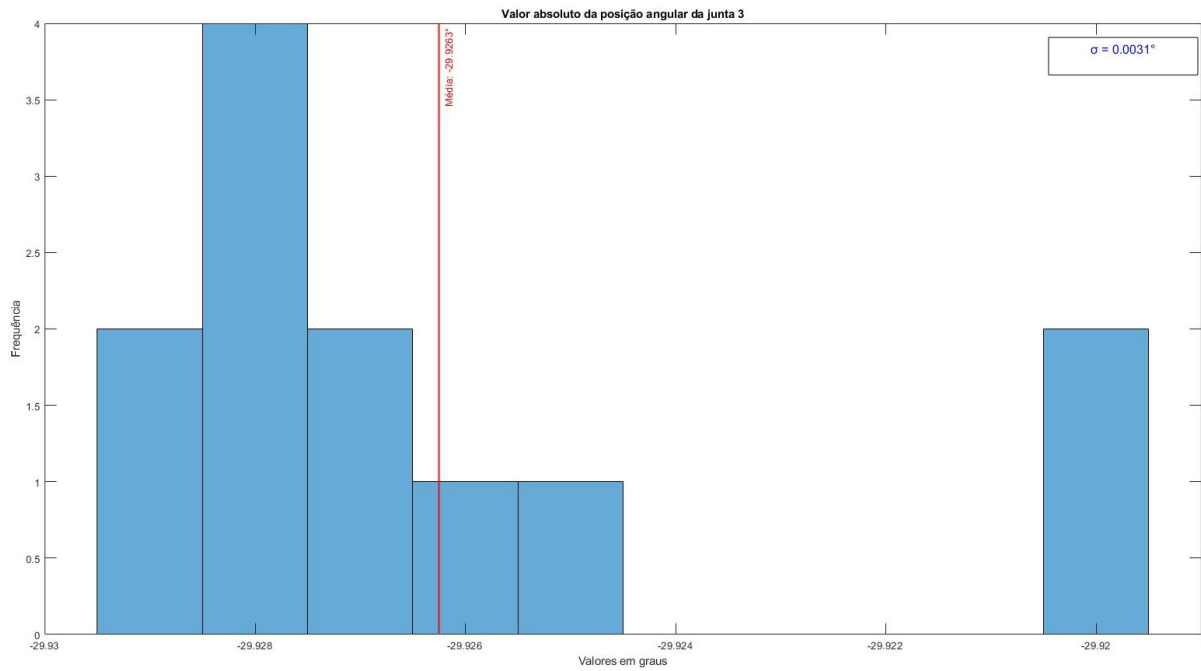


Figura D.55 – Valor absoluto da rotação da Junta 3 - Experimento 5.

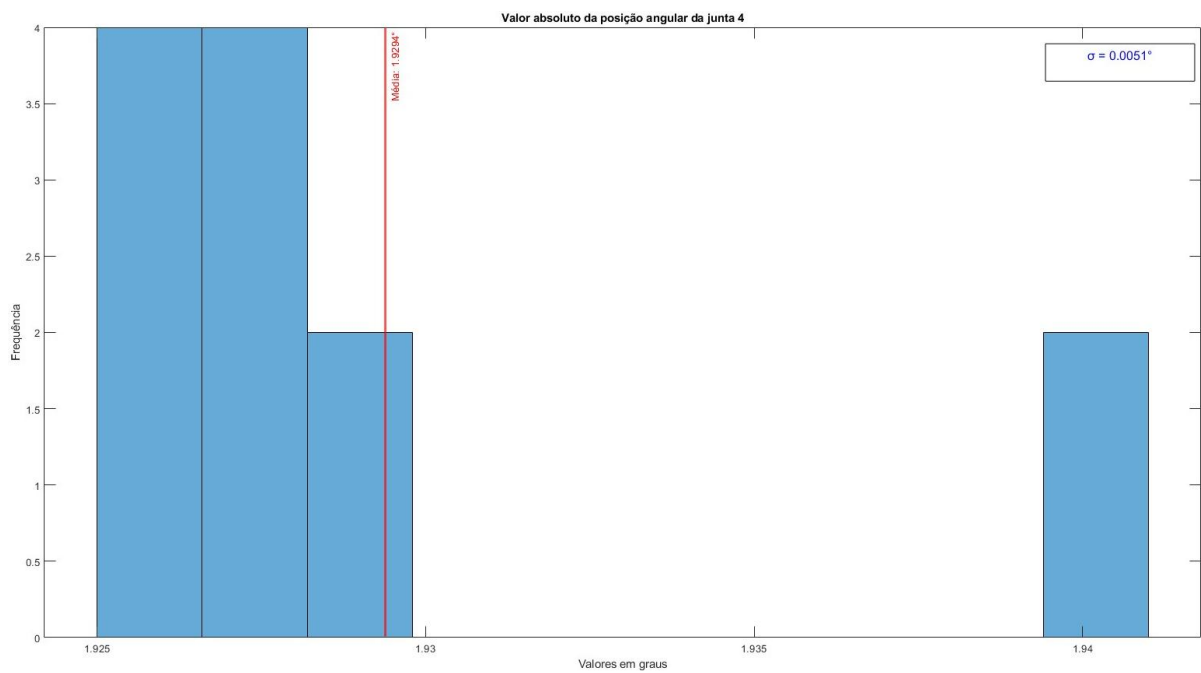


Figura D.56 – Valor absoluto da rotação da Junta 4 - Experimento 5.

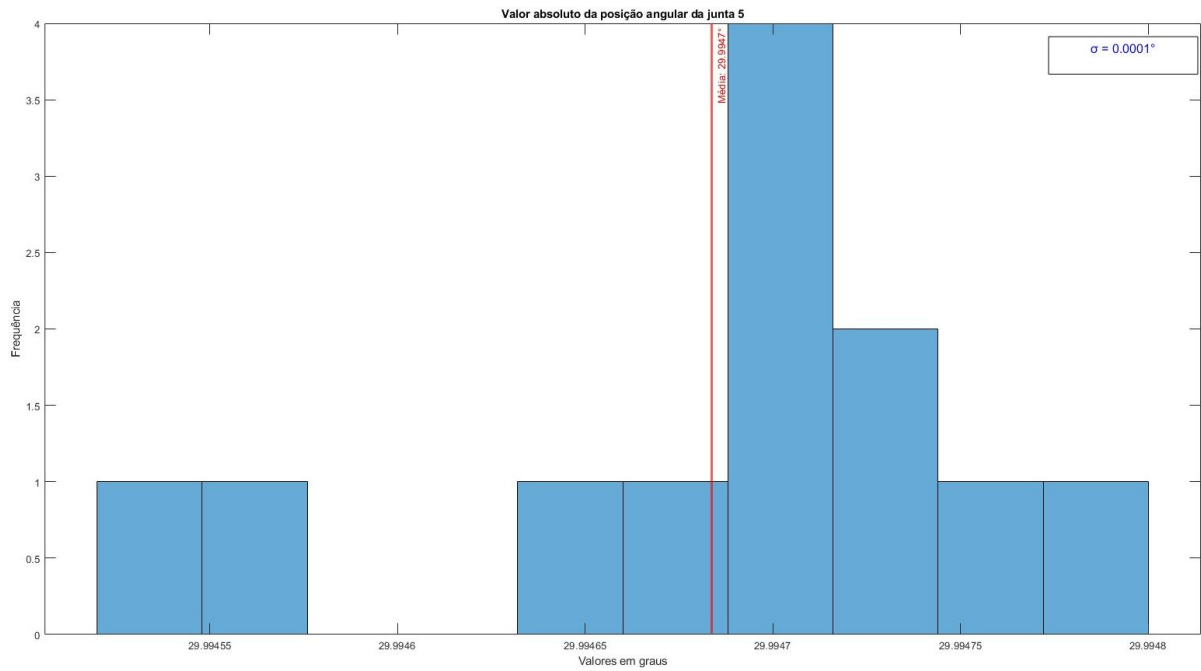


Figura D.57 – Valor absoluto da rotação da Junta 5 - Experimento 5.

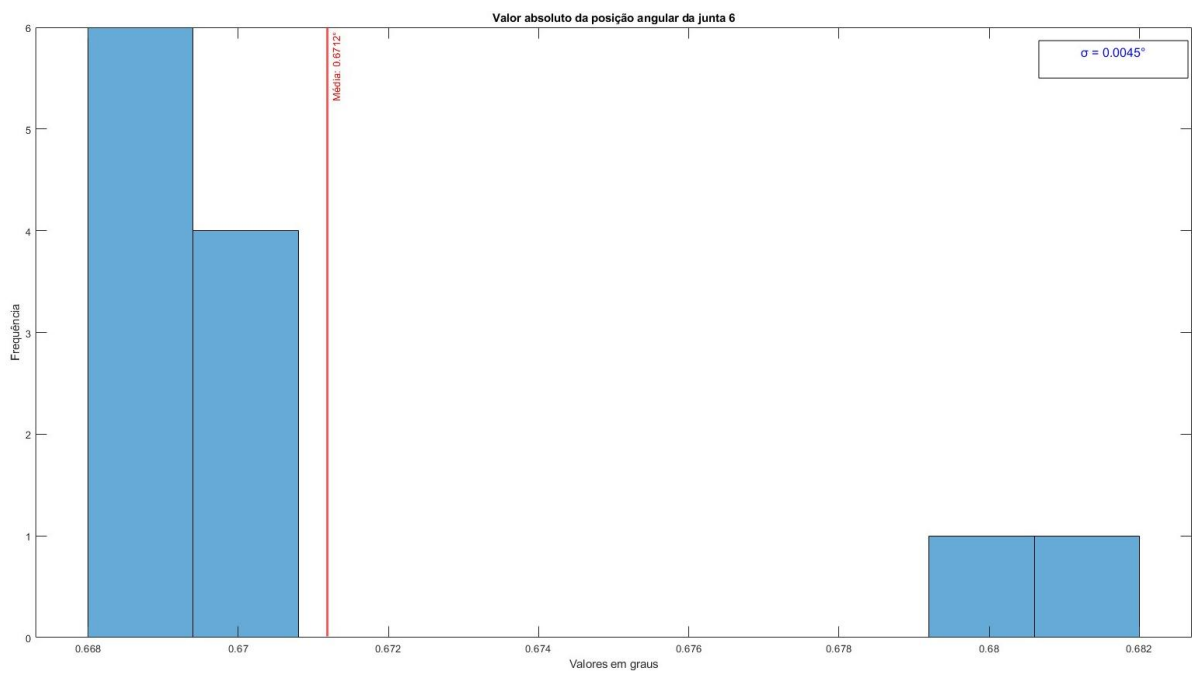


Figura D.58 – Valor absoluto da rotação da Junta 6 - Experimento 5.