



UNIVERSIDADE FEDERAL DA PARAIBA
PRO-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO

PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA
EM SISTEMAS DE COMPUTAÇÃO

ANA MARIA PINTO DA SILVA NASCIMENTO

RASTREAMENTO DE ALVOS MÓVEIS UTILIZANDO CONTROLE
GEOMÉTRICO BASEADO EM VISÃO EM VEÍCULOS AÉREOS NÃO
TRIPULADOS

JOÃO PESSOA - PB
2025

ANA MARIA PINTO DA SILVA NASCIMENTO

RASTREAMENTO DE ALVOS MÓVEIS UTILIZANDO
CONTROLE GEOMÉTRICO BASEADO EM VISÃO EM
VEÍCULOS AÉREOS NÃO TRIPULADOS

Dissertação apresentada ao Programa de Pós-Graduação em Informática da Universidade Federal da Paraíba, como parte das exigências para obtenção do título de Mestre em Informática em Sistemas de Computação.

Orientador: Prof. Dr. Alisson Vasconcelos de Brito

Coorientador: Dr^a. Sarah Pontes Madruga

JOÃO PESSOA - PB
2025

Catálogo na publicação
Seção de Catalogação e Classificação

N244r Nascimento, Ana Maria Pinto da Silva.

Rastreamento de alvos móveis utilizando controle geométrico baseado em visão em veículos aéreos não tripulados / Ana Maria Pinto da Silva Nascimento. - João Pessoa, 2025.

121 f. : il.

Orientação: Alisson Vasconcelos de Brito.

Coorientação: Sarah Pontes Madruga.

Dissertação (Mestrado) - UFPB/CI.

1. Informática. 2. Controle geométrico. 3. Veículos aéreos não tripulados (VANTs). 4. Rastreamento de alvos móveis. I. Brito, Alisson Vasconcelos de. II. Madruga, Sarah Pontes. III. Título.

UFPB/BC

CDU 004(043)



UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA



Ata da Sessão Pública de Defesa de Dissertação de Mestrado de **ANA MARIA PINTO DA SILVA NASCIMENTO**, candidata ao título de Mestre em Informática na área de Ciências da Computação, realizada em 27 de fevereiro de 2025.

1 Aos vinte e sete dias do mês de fevereiro do ano de dois mil e vinte e cinco, às 16h, no
2 Centro de Informática, reuniram-se os membros da Banca Examinadora constituída para
3 julgar o Trabalho Final da discente **ANA MARIA PINTO DA SILVA NASCIMENTO**, vinculada a
4 Universidade Federal da Paraíba sob a matrícula nº 20231004343, candidata ao grau de
5 Mestre em Informática, na área de “Ciências da Computação”, na linha de pesquisa
6 “Sistemas de Computação” do Programa de Pós-Graduação em Informática. A comissão
7 examinadora foi composta pelos professores Alisson Vasconcelos de Brito, orientador e
8 presidente da banca; Leonardo Vidal Batista, examinador interno ao programa; Sarah Pontes
9 Madruga, examinadora externa à instituição; e Saulo Oliveira Dornellas Luiz, examinador
10 externo à instituição. Dando início aos trabalhos, o Presidente da Banca cumprimentou os
11 presentes, comunicou a finalidade da reunião e passou a palavra à candidata para que
12 fizesse, oralmente, a exposição do trabalho de dissertação intitulado “**Rastreamento de**
13 **Alvos Móveis Utilizando Controle Geométrico Baseado em Visão em Veículos Aéreos não**
14 **Tripulados**”. Concluída a exposição, a candidata foi arguido pela Banca Examinadora, que
15 emitiu o seguinte parecer: “**aprovada**”. Do ocorrido, eu, Gean Paulo P. M. de Barros,
16 secretário, lavrei a presente ata que vai assinada por mim e pelos membros da Banca
17 Examinadora. João Pessoa, 27 de fevereiro de 2025.

Documento assinado digitalmente

gov.br GEAN PAULO PEREIRA MAURICIO DE BARROS
Data: 10/03/2025 21:12:38-0300
Verifique em <https://validar.iti.gov.br>

Gean Paulo P. M. de Barros

Secretário - SIAPE 2326476

Prof. Dr. Alisson Vasconcelos de Brito
Orientador (PPGI)

Prof. Dr. Leonardo Vidal Batista
Examinador interno ao Programa (PPGI)

Profª. Drª. Sarah Pontes Madruga
Examinadora externa à instituição (UFPA)

Prof. Dr. Saulo Oliveira Dornellas Luiz
Examinador externo à instituição (UFPE)

Documento assinado digitalmente

gov.br ALISSON VASCONCELOS DE BRITO
Data: 08/03/2025 15:41:42-0300
Verifique em <https://validar.iti.gov.br>

Documento assinado digitalmente

gov.br LEONARDO VIDAL BATISTA
Data: 08/03/2025 10:14:10-0300
Verifique em <https://validar.iti.gov.br>

Documento assinado digitalmente

gov.br SARAH PONTES MADRUGA
Data: 08/03/2025 10:50:07-0300
Verifique em <https://validar.iti.gov.br>

Documento assinado digitalmente

gov.br SAULO OLIVEIRA DORNELLAS LUIZ
Data: 08/03/2025 06:39:22-0300
Verifique em <https://validar.iti.gov.br>

Dedico este trabalho ao meu marido, Tiago Nascimento, aos meus pais e ao meu irmão, cujo apoio incondicional foi essencial para que eu seguisse em busca dos meus sonhos.

Agradecimentos

Agradeço a Deus pela força e coragem que me foram dadas ao longo desta jornada.

Ao meu marido, meu grande amor, Tiago Nascimento, que sempre me apoia na busca por seguir o que amo e que esteve incansavelmente ao meu lado, suportando meus surtos e desafios.

À minha família, especialmente à minha mãe, ao meu pai e ao meu irmão, pelo amor incondicional e apoio constante.

Ao professor Alisson Brito e à Sarah Madruga pela orientação, apoio e confiança.

Agradeço ao Wagner García, pela ajuda, companheirismo e paciência. Sei que, em muitos momentos, fiz você quase enlouquecer.

E a todos que, direta ou indiretamente, contribuíram para a realização deste trabalho.

Antes da morte, qualquer derrota é apenas psicológica.

Rastreamento de Alvos Móveis Utilizando Controle Geométrico Baseado em Visão em Veículos Aéreos não Tripulados

Resumo

Esta trabalho apresenta o desenvolvimento e implementação de um sistema de rastreamento de alvos móveis utilizando controle geométrico baseado em visão em veículos aéreos não tripulados (VANTs). O objetivo principal foi projetar um controlador robusto que permitisse aos VANTs detectar e rastrear alvos em tempo real e autônoma, garantindo precisão e estabilidade mesmo em cenários dinâmicos. Para isso, foi utilizado o controle geométrico no espaço $\mathbf{SE}(3)$, associadas a métodos de otimização como o Particle Swarm Optimization (PSO), visando aprimorar a resposta do sistema em comparação a controladores convencionais.

A pesquisa foi estruturada em quatro etapas principais. Primeiramente, foi realizada uma revisão da literatura sobre controle geométrico e visão computacional aplicados a UAVs, destacando desafios e lacunas existentes. Em seguida, foi desenvolvida uma modelagem matemática detalhada do quadrotor e do alvo, permitindo uma análise da dinâmica do sistema e fundamentando o projeto dos controladores. Posteriormente, foi implementado um sistema de rastreamento baseado em visão, que possibilitou a identificação e o acompanhamento de UAVs alvos, sendo validado por meio de experimentos simulados no ambiente GAZEBO e reais. Por fim, o controlador geométrico foi avaliado, demonstrando desempenho superior ao de abordagens tradicionais, com menor erro médio quadrático (RMSE) e maior estabilidade em manobras complexas no ambiente de simulação MATLAB.

Os resultados experimentais confirmam a eficácia da abordagem proposta, evidenciando sua aplicabilidade em cenários reais.

Palavras-chave: Controle geométrico, Veículos aéreos não tripulados (VANTs), Rastreamento de alvos móveis.

Tracking Moving Targets Using Vision-Based Geometric Control in Unmanned Aerial Vehicles

Abstract

This work portrays the development and implementation of a moving target tracking system using vision-based geometric control in unmanned aerial vehicles (UAVs). The main objective was to design a robust controller that would allow UAVs to detect and track targets in real time and autonomous, ensuring precision and stability even in dynamic scenarios. For this, geometric control was used in the $\mathbf{SE}(3)$ space, associated with optimization methods such as Particle Swarm Optimization (PSO), aiming to improve the system's response in comparison to conventional controllers.

The research was structured into four main stages. Firstly, a literature review was carried out on geometric control and computer vision applied to UAVs, highlighting existing challenges and gaps. Next, a detailed mathematical modeling of the quadcopter and target was developed, allowing an analysis of the system dynamics and providing a basis for the design of the controllers. Subsequently, a vision-based tracking system was implemented, which enabled the identification and tracking of target UAVs, being validated through simulated experiments in the GAZEBO environment and real ones. Finally, the geometric controller was evaluated, demonstrating superior performance compared to traditional approaches, with lower root mean square error (RMSE) and greater stability in complex maneuvers in the MATLAB simulation environment.

The experimental results confirm the effectiveness of the proposed approach, highlighting its applicability in real scenarios.

Keywords: Geometric control, Unmanned aerial vehicles (UAVs), Mobile target tracking.

Lista de figuras

Figura 1 – Imagem do drone x500	5
Figura 2 – Sistema de Navegação Proposto.	6
Figura 3 – Número de artigos por abordagem de controle desde 1999. . . .	12
Figura 4 – Convenção dos eixos para o modelo de quadricóptero.	30
Figura 5 – Retângulo delimitador visto em simulação e experimento real .	42
Figura 6 – Esquema da projeção de um VANT alvo e sua esfera delimitadora no plano de projeção da câmera $\mathcal{P}$. É mostrada uma visão em perspectiva da situação. O referencial do robô é definido pelo centro de massa do robô O_R e pelos vetores $\hat{b}_1$, $\hat{b}_2$ e $\hat{b}_3$. O referencial da câmera é definido pelos vetores $\hat{c}_1$ e $\hat{c}_2$. O referencial do mundo é definido pelos vetores $\hat{e}_1$, $\hat{e}_2$ e $\hat{e}_3$	46
Figura 7 – Visão geral das trajetórias do VANT1 e do VANT2 alvo, e a trajetória estimada do alvo (VANT2) durante as simulações. A linha verde representa a trajetória do VANT1 que está rastreando o alvo. A linha azul mostra o caminho estimado do VANT2 alvo, e a linha vermelha representa a trajetória real do VANT2 alvo.	50
Figura 8 – Erro quadrático médio da posição nas coordenadas X, Y e Z durante as simulações (linha preta). A linha vermelha representa o erro no eixo X, a linha verde representa o erro no eixo Y e a linha tracejada azul representa o erro no eixo Z.	51
Figura 9 – Trajetórias dos VANTs durante a simulação com dois drones. A linha verde representa a trajetória do VANT1 (drone detector), enquanto a linha vermelha representa a trajetória real do VANT2 (alvo). A linha azul indica a trajetória estimada pelo algoritmo de detecção para o VANT2. A linha preta mostra a trajetória do segundo alvo (VANT3).	52

Figura 10 – Erro quadrático médio da raiz da posição nas coordenadas X, Y e Z durante as simulações com dois alvos (linha preta). A linha vermelha representa o erro no eixo X, a linha verde representa o erro no eixo Y e a linha tracejada azul representa o erro no eixo Z.	53
Figura 11 – Ambas as situações mostram a eficácia do algoritmo na detecção do drone durante o experimento.	55
Figura 12 – Experimentos internos realizados utilizando dois VANTs.	56
Figura 13 – Imagem de vista superior das trajetórias do drone detector e do drone alvo. A linha vermelha sólida representa a trajetória real do alvo. As cruzes azuis mostram as posições estimadas do drone alvo, e a linha verde representa a posição fixa do drone detector.	57
Figura 14 – Erros de posição durante o experimento com robôs reais, sem orientação.	58
Figura 15 – Imagem de vista superior das trajetórias do drone detector e do drone alvo. A linha vermelha sólida representa a trajetória real do alvo. A linha azul mostra a trajetória estimada do drone alvo, e a linha verde representa a posição do drone detector enquanto altera sua orientação.	59
Figura 16 – Erros de posição durante o experimento com robôs reais, com mudança de orientação.	60
Figura 17 – Erros de orientação durante o experimento com robôs reais, com mudança de orientação.	61
Figura 18 – Sistema de Navegação Proposto.	64
Figura 19 – Convenção dos eixos para o modelo de quadricóptero: A imagem descreve o frame do mundo $\mathcal{W} = \{\hat{\mathbf{e}}_x, \hat{\mathbf{e}}_y, \hat{\mathbf{e}}_z\}$ no qual a posição 3D e a orientação do corpo VANT são expressas. O frame do robô $\mathcal{B} = \{\hat{\mathbf{b}}_x, \hat{\mathbf{b}}_y, \hat{\mathbf{b}}_z\}$ se relaciona com $\mathcal{W}$ pela translação $\mathbf{r} = [x, y, z]^\top$ e pela rotação $\mathbf{R}^\top$. O vetor de direção do VANT, $\mathbf{h}$, que é uma projeção de $\hat{\mathbf{b}}_x$ para o plano $span(\hat{\mathbf{e}}_x, \hat{\mathbf{e}}_y)$, forma o ângulo de direção $\eta = \text{atan2}(\hat{\mathbf{b}}_x^\top \hat{\mathbf{e}}_y, \hat{\mathbf{b}}_x^\top \hat{\mathbf{e}}_x) = \text{atan2}(\mathbf{h}_y, \mathbf{h}_x)$	66
Figura 20 – Diagrama de Controle	76

Figura 21 – Trajetória 3D e RMSE da posição em SE(3) para o movimento em quadrado.	78
Figura 22 – Trajetória 3D e RMSE da posição no PID para o movimento em quadrado.	79
Figura 23 – Trajetória 3D e RMSE da posição no SE3 para o movimento circular.	79
Figura 24 – Trajetória 3D e RMSE da posição no PID para o movimento circular.	79

Lista de tabelas

Tabela 1 – Abordagens de controle baseadas na visão e seus objetivos. . . .	23
Tabela 2 – O impacto da velocidade do rotor no movimento do quadricóptero.	30
Tabela 3 – Distribuição do número de imagens para o treinamento de IA .	43
Tabela 4 – Métricas alcançadas com os melhores valores.	43
Tabela 5 – Métricas para Simulações e Experimentos com Robôs Reais. . .	62
Tabela 6 – Parâmetros utilizados na sintonia dos ganhos	75
Tabela 7 – Valores dos ganhos obtidos pelo PSO	75
Tabela 8 – Parâmetros do VANT	76
Tabela 9 – Comparação RMSE	78

Lista de abreviaturas e siglas

UFPB	Universidade Federal da Paraíba
DI	Departamento de Informática
PPGI	Programa de Pós-Graduação em Informática
MAVs	Veículos Aéreos de Multirotores
VANTs	Veículos Aéreos Não Tripulados
CNNs	Redes Neurais Convolucionais
MRS	Multi-Robot System
PBVS	Position-Based Vision Servoing
IBVS	Image-Based Vision Servoing
M_{YOLOv8}	Modelo YOLOv8

Lista de símbolos

Γ	Matriz de acoplamento
λ	Comprimento de onda [m]
$\in$	Pertence
$\mathbf{r}$	Vetor de posição no referencial do mundo [m]
$\dot{\mathbf{r}}$	Vetor de velocidade no referencial do mundo [m/s]
$\ddot{\mathbf{r}}$	Vetor de aceleração no referencial do mundo [m/s^2]
φ	Vetor de orientação angular no referencial do mundo [rad]
$\dot{\varphi}$	Vetor de velocidade angular no referencial do mundo [rad/s]
ϕ	Ângulo de roll [rad]
θ	Ângulo de pitch [rad]
ψ	Ângulo de yaw [rad]
$\dot{\phi}$	Velocidade angular de roll [rad/s]
$\dot{\theta}$	Velocidade angular de pitch [rad/s]
$\dot{\psi}$	Velocidade angular de yaw [rad/s]
$\mathbf{v}$	Vetor de velocidade linear no referencial do VANT [m/s]
$\mathbf{w}$	Vetor de velocidade angular no referencial do VANT [rad/s]
δ	Velocidades angulares dos motores [rad/s]
T	Empuxo total no VANT, gerado pelos motores do drone [N]
$\mathbf{T}^v$	Vetor de empuxo nos eixos X, Y e Z [N]
τ	Vetor de torques em relação ao seu centro de massa [Nm]

τ_θ	Momento em torno do eixo X do corpo do VANT (roll) [Nm]
τ_ϕ	Momento em torno do eixo Y do corpo do VANT (pitch) [Nm]
τ_ψ	Momento em torno do eixo Z do corpo do VANT (yaw) [Nm]
R	Matriz de rotação
g	Aceleração gravitacional em $[m/s^2]$
m	Massa do veículo [kg]
F_t	Força de propulsão [N]
F_g	Força gravitacional [N]
F_{DA}	Força de arrasto [N]
M	Momento [Nm]
$\hat{\mathbf{e}}_x$	Vetor unitário no eixo x
$\hat{\mathbf{e}}_y$	Vetor unitário no eixo y
$\hat{\mathbf{e}}_z$	Vetor unitário no eixo z
J	Matriz de momentos de inércia $[\text{kg}\cdot\text{m}^2]$
C_d	Coeficiente de arrasto
ρ	Densidade do ar $[\text{kg}/\text{m}^3]$
A_i	Área do disco da hélice $[\text{m}^2]$
v_i	Velocidade da hélice $[\text{m}/\text{s}]$
M_u	Torque devido à velocidade angular do corpo do quadrotor [Nm]
M_t	Torque do Empuxo e Arrasto [Nm]
M_{gir}	Torque do giroscópio gerado pela rotação dos rotores [Nm]
w_x	Matriz assimétrica

x	Posição no eixo x [m]
y	Posição no eixo y [m]
z	Posição no eixo z [m]
$\dot{x}$	Velocidade no eixo x [m/s]
$\dot{y}$	Velocidade no eixo y [m/s]
$\dot{z}$	Velocidade no eixo z [m/s]
$\ddot{x}$	Aceleração no eixo x [m/s ²]
$\ddot{y}$	Aceleração no eixo y [m/s ²]
$\ddot{z}$	Aceleração no eixo z [m/s ²]
$SO(3)$	Grupo especial ortogonal, representa a rotação em 3D
$SE(3)$	Grupo especial Euclidiano, combina translação e rotação em 3D
$\boldsymbol{\omega}$	Vetor de velocidade angular [rad/s]
$\mathbf{h}$	Vetor de direção projetado no plano xy do quadrotor
η	Ângulo de direção do VANT, calculado a partir do vetor $\mathbf{h}$
$\mathbf{F}_c$	Força de controle no sistema [N]
$\mathbf{R}_d$	Matriz de rotação desejada
k_p, k_v, k_a	Ganhos do controlador Proporcional, Derivativo e Integral
$\mathbf{e}_p, \mathbf{e}_v, \mathbf{e}_a$	Erros de posição, velocidade e aceleração
$\boldsymbol{\Psi}(\mathbf{R}, \mathbf{R}_d)$	Função de erro de rotação entre $\mathbf{R}$ e $\mathbf{R}_d$
$\mathbf{e}_R$	Erro de rotação no espaço $SO(3)$
$\mathbf{e}_\Omega$	Erro na taxa de variação angular [rad/s]
$\boldsymbol{\tau}_d$	Torque desejado para o controlador [Nm]

$\mathbf{h}_d$	Vetor de direção desejado
γ_R	Parâmetro de ajuste do controle de rotação
$\mathbf{F}_d$	Força desejada para o controlador [N]
k_R, k_w, k_{wi}	Ganhos do controlador para a rotação e direção
$\mathbf{T}_d$	Empuxo desejado calculado pelo controlador [N]
$\mathbf{I}_{RGB}$	Imagem de entrada em formato RGB-D, com altura H , largura W e canais C
$\mathbf{I}_{resized}$	Imagem redimensionada para a resolução padrão de 1280×720
$\mathbf{O}$	Resultado da detecção, representando N caixas delimitadoras (bounding boxes)
$\mathbf{b}_i$	Caixa delimitadora representada por quatro componentes: coordenadas (u_i, v_i) e dimensões (w_i, h_i)
$\mathbf{R}_i$	Retângulo publicado com a posição e dimensões da caixa delimitadora no espaço da imagem
$\mathbf{R}_{ROI}$	Conjunto de retângulos representando todas as detecções realizadas
Z_i	Coordenada de profundidade no referencial da câmera, calculada a partir da imagem de profundidade
(X_i, Y_i)	Coordenadas do ponto no referencial da câmera, calculadas usando o modelo de pinhole
${}^{[c]}\hat{\boldsymbol{\xi}}_{[G]}$	Centróide do retângulo no referencial da câmera [m]
$\mathbf{T}_{cw}$	Matriz de transformação para converter coordenadas do referencial da câmera para o referencial do mundo
X_w, Y_w, Z_w	Coordenadas do ponto no referencial do mundo após a transformação [m]

$\mathbf{T}_{cr}$	Matriz de transformação para converter coordenadas do referencial da câmera para o referencial do robô
α, β	Ângulos de orientação entre o VANT e o drone alvo, relacionados aos ângulos de inclinação e guinada [rad]
η_d	Direção desejada calculada como o ângulo formado pelo vetor de direção do VANT [rad]
V	Velocidade do ponto estimado, usada para filtrar pontos discrepantes durante o rastreamento [m/s]
V_{max}	Limiar para filtrar pontos discrepantes com base na velocidade estimada

Sumário

1 – Introdução	1
1.1 Motivação	2
1.2 Objetivos	2
1.2.1 Objetivos Específicos	2
1.3 Contribuições Proposta	3
1.4 Materiais	5
1.4.1 VANT utilizado	5
1.5 Estrutura da Dissertação	7
2 – Trabalhos Relacionados	9
2.1 Controle de Movimento com Percepção Visual para VANT	9
2.2 Abordagens de Controle	11
2.2.1 Controle Geométrico	12
2.3 Detecção de Alvos utilizando Visão Computacional	18
2.4 Considerações Finais	21
3 – Modelagem Matemática	24
3.1 Servo Visual	24
3.1.1 Servovisual Baseado em Posição (Position-Based Visual Servoing)	25
3.1.2 Servovisual Baseado em Imagem (Image-Based Visual Servoing)	25
3.1.2.1 Movimento da Câmera e da Imagem	26
3.1.2.2 Controlando o Movimento	28
3.2 Modelagem do VANT	29
3.2.1 Cinemática	33
3.2.2 Dinâmica	34
3.2.3 Efeitos Aerodinâmicos	37
3.2.3.1 Arrasto Induzido	38
3.2.3.2 Arrasto Parasita	38

3.3	Conclusão	38
4	– Visão	40
4.1	Detecção de objetos voadores baseada em visão	40
4.1.0.1	Treinamento	42
4.1.1	Estimativa da Posição do Alvo	44
4.1.2	Navegação Guiada por Percepção	45
4.2	Resultados	49
4.2.1	Simulação	49
4.2.1.1	Simulação com um Único Alvo	49
4.2.1.2	Simulação com Multiplos Alvos	50
4.2.2	Experimentos Reais	54
4.2.2.1	Experimentos de robôs reais ao ar livre	55
4.2.2.2	Experimentos com robôs reais em ambiente in- terno sem rastreamento visual	56
4.2.2.3	Experimentos com robôs reais em ambiente in- terno com rastreamento visual	58
4.2.2.4	Discussão dos Resultados	61
4.3	Conclusões	62
5	– Controle de Postura para Vãos Ágeis	64
5.1	Introdução	64
5.2	Controlador Geométrico de Postura	65
5.2.1	Modelo Dinâmico do VANT	65
5.2.2	Projeto do Controlador	67
5.2.2.1	Cálculo de $\mathbf{R}_d$	70
5.3	Sintonia	71
5.3.1	PSO	73
5.4	Resultados	75
5.5	Considerações Finais	80
6	– Conclusão	81
6.0.1	Trabalhos Futuros	82

Referências	84
-----------------------	----

1 Introdução

O cenário da tecnologia aplicada aos Veículos Aéreos de Multirotores (MAVs) está em constante transformação, impulsionada pela evolução incessante de suas tecnologias. Esse avanço notável redefine suas aplicações e desencadeia um progresso significativo em seu desenvolvimento. No epicentro dessa revolução, destaca-se a exploração de estratégias avançadas, como o controle baseado em visão, que atua como um catalisador com o potencial de elevar a capacidade dos MAVs para operar de forma eficaz e autônoma em diversas situações ([LEE, 2016](#)).

O controle baseado em visão, fundamentado na análise de dados visuais capturados por câmeras embarcadas, emerge como uma abordagem crucial na tomada de decisões para o controle de robôs aéreos. Suas aplicações abrangem desde navegação até tarefas complexas como detecção e evasão de obstáculos, além do controle de formação, onde múltiplos MAVs operam em sincronia ([ABBASI et al., 2016](#)). O controle de formação baseado em visão, embora promissor em seu desenvolvimento, enfrenta desafios como a demanda por processamento de imagem em tempo real e a gestão da incerteza inerente aos dados visuais ([MáTHé; BUçONI, 2015](#)), ([TONG et al., 2023](#)).

A aplicação de técnicas de mapeamento 3D com fotogrametria representa um marco na representação tridimensional de objetos, contribuindo para a preservação cultural de construções históricas e o rastreamento preciso de alvos em ambientes complexos ([BALDIVIESO et al., 2022](#)). A utilização de Redes Neurais Convolucionais (CNNs) em imagens amplia ainda mais o escopo do controle baseado em visão, possibilitando o reconhecimento automatizado de alvos em ambientes dinâmicos, como o caso do reconhecimento de abelhas melíferas ([BARROS et al., 2021](#)).

Diante dessas considerações, a presente dissertação propõe-se a explorar o controle geométrico baseado em visão para o rastreamento de alvos móveis utilizando um VANT. Esta área de pesquisa, revelada como promissora, transcende as fronteiras disciplinares, encontrando aplicações desde a preservação cultural até a análise e correlação de dados na construção civil. A convergência de avanços em

visão computacional e técnicas de aprendizado de máquina promove o desenvolvimento de sistemas cada vez mais eficientes e autônomos para os VANTs, abrindo novas fronteiras de aplicação em cenários diversos.

1.1 Motivação

A motivação substancial deste trabalho decorre da crescente necessidade de aprimorar os sistemas de controle associados a Veículos Aéreos Não Tripulados (VANTs), conferindo-lhes maior eficiência e autonomia em uma diversidade de cenários operacionais. A integração de visão computacional e técnicas de controle surge como uma abordagem promissora para alcançar esse objetivo ambicioso.

O contexto do desenvolvimento de VANTs demanda a implementação de sistemas capazes de realizar o rastreamento e detecção de objetos em ambientes diversos. Essa capacidade torna-se fundamental para a identificação de alvos específicos em cenários tanto conhecidos como hostis, possibilitando, por exemplo, a detecção de invasores, como outros drones, em espaços aéreos compartilhados.

É relevante ressaltar que a detecção e rastreamento visual dos drones intrusos, figura como um desafio continuamente avaliado pela comunidade robótica. A necessidade de solucionar esse problema específico impulsiona a pesquisa neste domínio, buscando desenvolver abordagens inovadoras e eficazes para garantir a segurança e o desempenho desses sistemas em constante evolução. Este estudo se insere nesse contexto desafiador, contribuindo para a expansão do conhecimento e o avanço nas capacidades operacionais dos VANTs.

1.2 Objetivos

1.2.1 Objetivos Específicos

Os objetivos específicos desta dissertação foram a implementação e validação de um sistema de controle geométrico baseado em visão para o rastreamento de alvos móveis utilizando VANTs. Além disso, buscou-se identificar desafios e limitações inerentes ao problema e propor soluções eficazes. Os objetivos detalhados foram:

- implementar o algoritmo de visão baseado no YOLOv8 para a detecção e localização de alvos móveis durante voos autônomos do drone;
- validar o algoritmo de visão para detectar alvos móveis em ambiente simulado utilizando o GAZEBO, avaliando sua eficácia na detecção de alvos móveis;
- implementar o controle geométrico para o rastreamento de alvos móveis no espaço $SE(3)$;
- validar o desempenho do controle geométrico em ambiente simulado no MATLAB;
- realizar testes experimentais em um drone real para avaliar o desempenho do sistema de rastreamento de alvos móveis durante voos autônomos.

Esses objetivos específicos foram alcançados ao longo do desenvolvimento da pesquisa, garantindo a obtenção de resultados relevantes e contribuindo para o avanço do conhecimento na área de controle geométrico baseado em visão aplicado a VANTs.

1.3 Contribuições Proposta

As contribuições deste trabalho podem ser resumidas da seguinte forma:

- detector sem marcas para localizar UAVs dentro do campo de visão e acoplamos esse detector a uma estimativa de direção para melhorar o rastreamento do alvo;
- base de dados aberta com mais de quatro mil imagens usadas para treinar nosso algoritmo que está disponível publicamente para a comunidade; [adicionar o link aqui](#)
- lei de controle geométrico para rastreamento de atitudes;
- inclusão dos efeitos aerodinâmicos apresentada em ([MADRUGA et al., 2022](#)) na formulação da lei de controle.

Além das contribuições mencionadas anteriormente, este trabalho também resultou nas seguintes contribuições:

- **Artigo aceito** na revista *Drones*, intitulado *An Improved Spanning Tree-Based Algorithm for Coverage of Large Areas Using Multi-UAV Systems* (??).
- **Artigo aceito** no (*LARS*) *Latin American Robotics Symposium 2023, Brazilian Symposium on Robotics (SBR), and Workshop on Robotics in Education (WRE)*, intitulado *Vision-based K-Nearest Neighbor Approach for Multiple Search and Landing with Energy Constraints* ([SALES et al., 2023](#))
- **Submissão de um artigo** para o periódico *Robotics and Autonomous Systems*, intitulado *A Review on Vision-based Control for Multi-Rotor Aerial Vehicles*, em **18 de outubro de 2024**. O artigo encontra-se atualmente em processo de revisão.
- **Submissão de um artigo** para a revista *IEEE Robotics and Automation Practice*, intitulado *Practical Application of an Intruder UAV Detection and Tracking*, em **19 de Dezembro de 2024**. O artigo encontra-se atualmente em processo de revisão.
- **Submissão de um Artigo de Conferência** para *CROS 2025 (1st Conference on Robotics, organized by SBRobotica)*, intitulado *A Multi-Rotor UAV Modified Geometric Attitude Controller*, em **12 de janeiro de 2025**. O artigo encontra-se atualmente em processo de revisão.
- **Submissão de um Artigo de Conferência** para *CROS 2025 (1st Conference on Robotics, organized by SBRobotica)*, intitulado *Fast and Accurate Vision-based Flying Object Detection and Tracking for Pursuing Intruder UAVs*, em **12 de janeiro de 2025**. O artigo encontra-se atualmente em processo de revisão.

1.4 Materiais

Esta seção apresenta o VANT utilizado neste trabalho, bem como a arquitetura de navegação proposta de controle baseado em visão para rastreamento de alvos em VANTs.



Figura 1 – Imagem do drone x500

Fonte: Próprio Autor

1.4.1 VANT utilizado

A Figura 1 apresenta o drone fabricado e utilizado nesse trabalho ([Hert et al., 2022](#)). Foram utilizados dois drones desse tipo: um para ser utilizado como alvo, e um segundo que será o VANT que estima e rastreia a posição do alvo. Esse VANT possui os seguintes componentes descritos detalhadamente abaixo:

- **Pixhawk 4 + GPS M8M:** sistema embarcado para processar comandos de controle de voo e processar dados dos sensores como IMU, Barômetro e GNSS;
- **Garmin:** Sensor tipo lidar de único feixe para estimação de altura;
- **RPLIDAR - A3:** sensor tipo lidar rotativo para estimação de distância e detecção de obstáculos;
- **RealSense D435i e D455:** câmeras de profundidade utilizadas para percepção visual e mapeamento 3D;

- **Intel NUC 10 (NUC10i7FNHN, 64GB):** computador central utilizado para processamento de alto desempenho;
- **Placa de distribuição:** placa utilizada para distribuir energia para os diferentes componentes do sistema;
- **4 Motores T-Motors (mn3510 kv700):** motores de alta potência para propulsão do drone;
- **4 Hélices de fibra de carbono:** hélices utilizadas para gerar o impulso necessário para o voo;
- **Frame Holybro X500:** estrutura principal do drone que abriga todos os componentes;
- **Impressões 3D:** peças personalizadas impressas em 3D para fins específicos do projeto, como case de bateria e pernas.

Considerando o UAV descrito na seção 1.4.1, propomos neste trabalho a arquitetura de navegação proposta na figura abaixo:

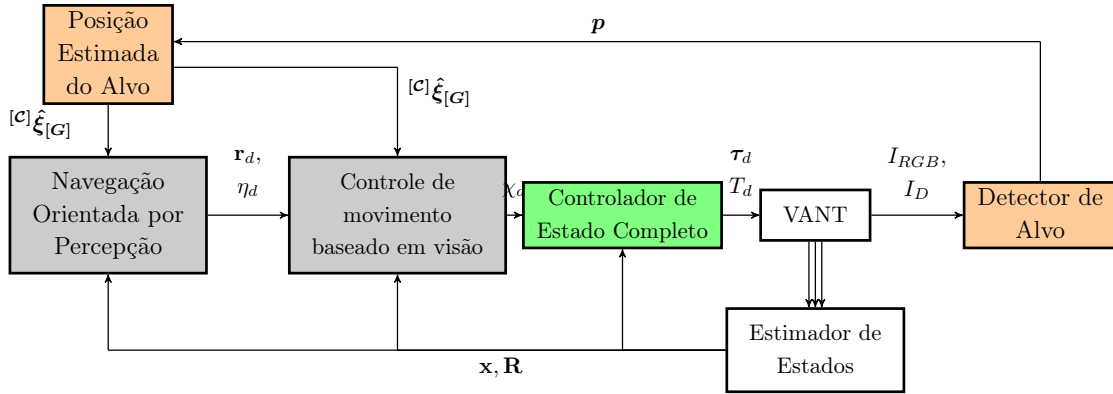


Figura 2 – Sistema de Navegação Proposto.

A arquitetura proposta nesse trabalho (Fig. 2) está inserida dentro do Framework MRS (BACA et al., 2021). Nesta mesma arquitetura, os blocos responsáveis pela detecção (*Detector de Alvo*) e estimação do alvo (*Posição Estimada do Alvo*) estão em laranja e serão vistos no capítulo 4. O bloco do controle geométrico proposto (*Controle do Estado Compelto*) será visto no capítulo 5. Os

blocos em cinza (*Navegação Orientada por Percepção e Controle de Movimento Baseado em Visão*) serão apresentados na sua versão simplificada, uma vez que estes são o foco de trabalhos futuros. O bloco de estimação de estados (*Estimador de Estados*) é parte do Framework MRS e o bloco que representa toda a eletromecânica do VANT é visto como **"VANT"**.

Dessa forma, na arquitetura da Fig. 2, o UAV captura as imagens em cores e profundidade (I_{RGB}, I_D) utilizando a câmera RGB-D e as envia para o bloco **"Target Detector"**. Então, este bloco as processa, criando uma imagem $\mathbf{p}$ com a região de interesse, onde $\mathbf{p} = [I_{RGB}, I_D, \mathbf{R}_{ROI}]^T$. O bloco **"Target Position Estimator"** recebe $\mathbf{p}$ e calcula a posição estimada do alvo ${}^{[W]}\hat{\xi}_{[G]}$ e a covariância ${}^{[W]}\hat{\Sigma}_{[G]}$ no frame do mundo. Então, o bloco **"Perception-Driven Navigation"** usa essas informações para calcular a posição desejada $\mathbf{r}_d$ e o rumo η_d para o UAV rastrear. Finalmente, o **"Vision-based Motion Controller"** recebe a trajetória contendo a posição desejada ($\mathbf{r}_d$) e o rumo (η_d), que por sua vez envia o estado desejado ($\chi_d = [x, \dot{x}, \ddot{x}, y, \dot{y}, \ddot{y}, z, \dot{z}, \ddot{z}, \eta, \dot{\eta}, \ddot{\eta}]^T$) para um controlador de postura chamado de **"Full-state Controller"**, o qual envia o Empuxo desejado (T_d) e a os Torques desejados (τ_d) para o UAV. Finalmente, o estado $\mathbf{x}$ e a matriz de rotação $\mathbf{R}$ são realimentados para o sistema de controle. Aqui, nossas contribuições estão dentro dos blocos laranja identificados como **"Target Detector"** e **"Target Position Estimator"**, dentro do bloco em verde identificado como **"Full-state Controller"**.

1.5 Estrutura da Dissertação

Neste capítulo, foram delineadas as motivações que impulsionaram esta pesquisa, os principais objetivos do trabalho e a metodologia adotada ao longo da dissertação. Agora, apresentamos a estrutura organizacional deste documento para facilitar a compreensão do leitor.

O Capítulo 2 abordará os trabalhos relacionados e a literatura recente, estabelecendo uma comparação entre nossas contribuições e o que já foi publicado e estudado nos últimos anos.

No Capítulo 3, serão introduzidos os conceitos fundamentais necessários

para compreender o modelo matemático do drone, destacando as contribuições deste trabalho. Este capítulo também discutirá a modelagem matemática do alvo, além de introduzir a técnica que iremos utilizar para futuramente modelarmos as leis de controle, representando uma das principais contribuições desta pesquisa.

O Capítulo 4 é dedicado à descrição da abordagem de detecção e rastreamento de objetos voadores baseada em visão, utilizando um UAV seguidor e um UAV alvo. Aqui, discutimos o uso de um sensor RGB-D e a implementação do algoritmo de detecção de objetos YOLOv8, validado em experimentos simulados e reais. A eficácia do sistema de rastreamento é demonstrada, mostrando sua aplicabilidade em operações de vigilância e monitoramento.

No Capítulo 5, apresentamos o desenvolvimento do controlador geométrico para o quadricóptero, utilizando o espaço $SE(3)$ para controle de postura. Discutimos a sintonia do controlador, utilizando a otimização por enxame de partículas (PSO) para ajustar os parâmetros de controle, e comparamos o desempenho do controlador proposto com o controlador PID, com base em simulações de voo ágil.

Finalmente, o Capítulo 6 traz as conclusões do trabalho e aponta as direções para trabalhos futuros, com ênfase na validação prática dos métodos propostos e na exploração de melhorias para o controlador geométrico.

2 Trabalhos Relacionados

Neste capítulo, realizamos uma análise dos trabalhos correlatos sobre controle de movimento e de formação baseado em visão para VANTs, identificando 220 artigos relevantes a partir de uma busca que resultou em 4335 artigos. A revisão abrangeu 16 metodologias distintas de controle e teve como objetivo fornecer uma visão abrangente do estado da arte nesse campo, destacando as técnicas que serão aplicadas em nossa pesquisa. A pesquisa foi fundamentada em artigos científicos revisados por pares, garantindo a confiabilidade e atualização das informações apresentadas

2.1 Controle de Movimento com Percepção Visual para VANT

O controle de movimento baseado em visão é uma abordagem inovadora na robótica moderna que tem demonstrado benefícios significativos em termos de interação eficaz e intuitiva dos robôs com o ambiente ao seu redor ([CHAUMETTE; HUTCHINSON, 2006](#)). Tradicionalmente, o movimento robótico em direção a objetos enfrenta desafios significativos, como a necessidade de determinar a pose precisa do objeto e garantir que o robô atinja essa pose. No entanto, a limitação dos robôs em "enxergar" suas ações desencadeou a proposta de um novo paradigma conhecido como "Controle de Movimento Baseado na Visão" ([CORKE, 2017](#)).

O controle baseado em visão surge como uma solução promissora, explorando a capacidade do robô de ver o objeto e usar informações visuais na malha de controle. Essa abordagem oferece vantagens significativas, como a flexibilidade no manuseio de peças, tolerância a variações de posição e capacidade de manusear objetos em movimento. Além disso, este sistema de controle baseado em visão envolve medição contínua do alvo e do robô, utilizando informações visuais para criar feedback em tempo real. Esta técnica difere substancialmente da abordagem convencional de simplesmente capturar uma imagem, determinar a posição do alvo e alcançá-lo. Medição e feedback contínuos oferecem robustez, mesmo diante de

interferências práticas ([HUTCHINSON et al., 1996](#)).

Existem duas abordagens principais na teoria do servocontrole visual: o Servo Visual Baseado em Posição (PBVS) e o Servo Visual Baseado em Imagem (IBVS). Ambas têm suas vantagens e desvantagens, e a escolha entre elas está ligada às especificidades da tarefa em questão. Além disso, o controle de movimento baseado na visão pode ser classificado como servo visual híbrido (HVS), que combina os pontos fortes do PBVS e do IBVS, oferecendo perspectivas promissoras para a robótica interativa ([CORKE, 2017](#)). No Capítulo ??, detalharemos a modelagem matemática dessas abordagens.

Um Veículo Aéreo Não Tripulado (VANT) é frequentemente considerado um robô com capacidade de movimento limitada devido à subatuação, o que significa que não é possível manipular todos os seis graus de liberdade no espaço de trabalho de forma independente. Embora possamos controlar a posição (X, Y, Z) e o ângulo de yaw (guinada), os ângulos de pitch (arfagem) e roll (rolagem) também precisam ser ajustados para realizar o deslocamento lateral no plano horizontal. É essencial que esses ângulos sejam mantidos próximos de zero quando o veículo está em estado de equilíbrio, como no voo pairado. Além disso, há uma interdependência entre os movimentos de rotação e arfagem do VANT.

No controle baseado em visão aplicado a VANT, o acoplamento entre os movimentos de roll e pitch e as coordenadas das características no plano da imagem pode dificultar o alcance da translação sem inclinação inicial. Isso pode resultar em aumento do erro das características da imagem, especialmente durante manobras agressivas. Além disso, os erros provenientes da percepção visual de cada robô são propagados por todos os robôs no grupo, especialmente em um contexto de controle de formação ([CORKE, 2017](#)).

Como mencionado anteriormente, o controle de movimento baseado em visão explora a capacidade do robô de perceber o alvo e utilizar informações visuais no loop de controle. Embora o controle baseado em visão seja amplamente classificado em controle de servo visual baseado em imagem (IBVS), controle de servo visual baseado em posição (PBVS) e controle de servo visual híbrido, essa classificação difere principalmente no tipo de referência de controle utilizado (seja características, pose ou ambos). Assim, pode-se perceber que poderíamos usar qualquer abordagem de controle para lidar com a convergência da pose do robô com base

na pose/características do objeto rastreado. De fato, a literatura está repleta de abordagens desse tipo (THUILOT et al., 2002; MEHTA et al., 2019; QIU et al., 2019; LIN et al., 2023).

Um dos primeiros trabalhos sobre controle baseado em visão remonta a 1987 (WEISS et al., 1987). Após 12 anos, um dos primeiros trabalhos sobre controle baseado em visão para helicópteros autônomos foi publicado por Shakernia et al. (SHAKERNIA et al., 1999). No entanto, o controle de movimento baseado em visão foi aplicado apenas em VANT de multirrotores 20 anos depois, com o trabalho de Hugo et al. (HUGO et al., 2007). Isso demonstra como essa área de pesquisa é relativamente recente, com menos de 20 anos.

2.2 Abordagens de Controle

Ao longo da última década de investigações sobre controle baseado em visão para veículos aéreos multirrotores, surgiram diversas metodologias de controle. Algumas foram introduzidas e posteriormente abandonadas, enquanto outras passaram por análises extensas. No entanto, como ocorre no campo da robótica aérea, a pesquisa em controle baseado em visão busca continuamente aprimorar aspectos como tratamento de ruído, resposta a perturbações, manobras agressivas e controle de formação.

A partir desta investigação, foi compilada a Tabela 1, que oferece uma visão geral das metodologias propostas ao longo dos anos para loops de controle baseados em visão e controle de formação. Destacam-se, em particular, as publicações mais recentes sobre o controle de veículos aéreos multirrotores. Embora esta dissertação aborde o controle baseado em visão, o foco principal está no desenvolvimento e análise de técnicas de controle adaptativo aplicadas a esses veículos.

Além disso, a Figura 3 apresenta a distribuição do número de pesquisas por abordagem de controle. Observa-se que a pesquisa em abordagens não lineares ainda é limitada, sugerindo a necessidade de mais investigações nesse campo para explorar seu potencial.

Número de Artigos por Abordagens de Controle

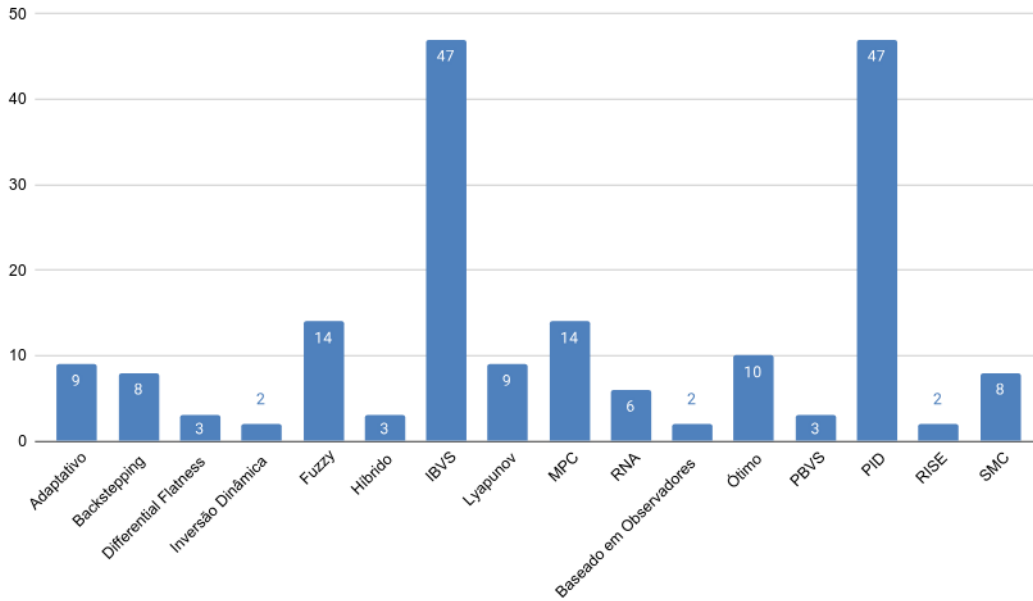


Figura 3 – Número de artigos por abordagem de controle desde 1999.

2.2.1 Controle Geométrico

O controle geométrico tem se consolidado como uma abordagem essencial para o controle de veículos aéreos não tripulados (VANTs), eliminando as limitações associadas às representações locais, como ângulos de Euler e quaternions. Esta abordagem se destaca pela capacidade de operar diretamente em variedades não lineares, como o grupo $SE(3)$ ou $SO(3)$, proporcionando maior robustez e precisão em manobras agressivas.

O trabalho pioneiro de Lee et al. (2010) (LEE et al., 2010) introduziu um controlador geométrico baseado em $SE(3)$, permitindo a estabilização e rastreamento de postura do quadricóptero. A formulação opera diretamente em $SO(3)$, evitando singularidades e garantindo estabilidade assintótica global mesmo em condições dinâmicas adversas. Este trabalho estabeleceu as bases para o controle geométrico aplicado a UAVs.

Como extensão, Lee et al. (2011) (LEE et al., 2011) desenvolveram um controle geométrico voltado para manobras complexas, lidando com sistemas su-

batuados, em que quatro entradas controlam seis graus de liberdade. Apesar de sua eficácia, a estabilidade global não é garantida devido a restrições topológicas do $SO(3)$. Simulações numéricas validaram a abordagem, mas limitações em regiões específicas do espaço de estado foram identificadas.

Goodarzi et al. (2015) (GOODARZI et al., 2015) propuseram um controlador adaptativo não linear baseado em $SE(3)$ para rastreamento de trajetória de quadrotor UAV, com capacidade de lidar com incertezas na dinâmica translacional e rotacional. A validação foi realizada por meio de simulações numéricas e testes reais utilizando um sistema de captura de movimento (Vicon), comprovando a eficácia do controlador em manobras agressivas.

Jean et al. (2018) (PEREIRA et al., 2021) introduziram um controle preditivo não linear (NMPC) formulado em $SE(3)$ para rastreamento de trajetória em ambientes obstruídos. O trabalho destacou a execução de manobras agressivas com rejeição de distúrbios, mas as validações foram restritas a simulações numéricas, sem testes experimentais.

Simha et al. (2016) (SIMHA et al., 2016) desenvolveram um controle de atitude adaptativo para um drone esférico, formulado diretamente em $SO(3)$. A proposta eliminou singularidades associadas a parametrizações Euclidianas, garantindo agilidade e precisão mesmo em cenários com incertezas significativas.

Em 2022, Zhao et al. (DUONG; ATANASOV, 2022) apresentaram um controlador adaptativo geométrico com aprendizado de distúrbios via redes neurais ODE Hamiltonianas. A proposta foi validada por simulações no motor de física PyBullet, demonstrando eficácia na compensação de distúrbios em tempo real.

Lee et al. (2013) (FERNANDO et al., 2011) desenvolveram um controlador adaptativo robusto para rastreamento de atitude em $SO(3)$, sem necessidade de conhecimento prévio da matriz de inércia do quadrotor. Testes experimentais validaram a eficácia do método, mesmo sob distúrbios não estruturados.

Kotaru et al. (2020) (KOTARU et al., 2020) propuseram um controlador adaptativo L1 geométrico para atitude do quadricóptero em $SO(3)$, combinando MRAC e filtros passa-baixa para lidar com incertezas e distúrbios externos. A abordagem foi validada em simulações e testes práticos com um quadrotor Autel X-star.

Outro avanço significativo foi realizado por Zhao et al. (2020) (BISHEBAN;

LEE, 2021), que integraram redes neurais a um controlador geométrico em SE(3) para compensar distúrbios aerodinâmicos. A proposta foi validada por meio de simulações e testes de voo real em ambientes com rajadas de vento.

Lee et al. (2015) (LEE, 2015) desenvolveram um controle geométrico adaptativo para transporte aéreo de cargas rígidas com múltiplos quadricópteros. As simulações numéricas demonstraram a eficácia do sistema, mas não foram realizados testes experimentais.

Prasanth et al. (2020) (KOTARU et al., 2020) introduziram um controlador geométrico L1 robusto, validado por simulações e experimentos, consolidando a eficácia da abordagem L1 para UAVs.

Cai et al. (2024) (CAI; XIAN, 2024) propuseram um controle hierárquico robusto para transporte aéreo com múltiplos UAVs e carga suspensa. A abordagem foi validada em simulações, mas sem testes práticos.

O controle adaptativo surge como uma abordagem dinâmica no projeto de sistemas de controle, impulsionado pela busca por desempenho otimizado em condições variáveis. Um marco histórico nesse desenvolvimento foi o controlador "autoadaptável" Minneapolis Honeywell MH-96, utilizado no programa NASA X-15, que realizou quase 200 voos bem-sucedidos entre 1959 e 1968 (DYDEK et al., 2010). Embora bem-sucedido em muitos aspectos, o MH-96 enfrentou críticas devido à falta de uma análise analítica de sua estabilidade, destacando a necessidade de fundamentação teórica sólida (JENKINS, 2000).

Desde então, uma variedade de métodos tem sido desenvolvida para lidar com sistemas dinâmicos, tanto lineares quanto não lineares, enfrentando incertezas paramétricas e dinâmicas (WISE et al., ; KARASON; ANNASWAMY, 1994; IOANNOU; TSAKALIS, 1986b; IOANNOU; TSAKALIS, 1986a). A aplicação dessas técnicas se estendeu ao controle de aeronaves, com resultados promissores em simulações e testes de voo (TAO; KOKOTOVIC, 1995; ANDERSON et al., 2009; SHIN et al., 2008). Exemplos notáveis incluem o uso de algoritmos sequenciais de mínimos quadrados em aeronaves como o F-16 e a implementação de redes neurais adaptativas em diversos tipos de aeronaves (KIM; CALISE, 1997; CALISE; RYSDYK, 1998; PACHTER et al., 1995). Mesmo em aplicações específicas, como munição guiada, como o Laser JDAM, o controle adaptativo mostrou-se eficaz, permitindo adaptação sem a necessidade de redesenhos extensivos (GROVES et

al.,). Contudo, desafios persistentes, como a coordenação entre múltiplos veículos, ainda são objeto de pesquisa ativa (GIBSON et al., 2009).

Além dos avanços na teoria dos sistemas adaptativos estáveis, diversas modificações foram propostas na abordagem padrão do MRAC, visando melhorar o desempenho e a robustez. Entre elas, destacam-se o MRAC Combinado (CM-RAC), que combina controle adaptativo direto e indireto, proporcionando transientes mais suaves (SLOTINE; LI, 1989; NADA, a; DUARTE; NARENDRA, 1989; LAVRETSKY, 2009). Outra modificação relevante é o uso de ganhos adaptativos variantes no tempo, como a lei de esquecimento de ganho limitado (BGF), que permite uma convergência mais rápida e suave dos parâmetros adaptativos (SLOTINE; LI, 1991). Embora não especificamente projetadas para sistemas com atraso de tempo, essas modificações podem contribuir para uma margem de atraso melhorada, reduzindo o conteúdo de alta frequência dos sinais no sistema. Além disso, a abordagem Adaptive Posicast Controller (APC) surge como uma extensão adaptativa do Smith Predictor, direcionada especialmente para o controle de sistemas com atraso de tempo conhecido, proporcionando uma maneira eficaz de cancelar o efeito do atraso no sistema (YILDIZ et al., 2010). Essas modificações representam ferramentas valiosas no projeto de controladores para sistemas com atraso de tempo, oferecendo soluções promissoras para uma variedade de problemas de controle (YILDIZ et al., 2010).

O desenvolvimento dos quadrotores remonta ao século XX, com recentes avanços significativos que os transformaram em uma plataforma de pesquisa popular devido à sua simplicidade de design e baixo custo. No entanto, o controle preciso dessas aeronaves é desafiador devido a fontes variadas de incerteza no sistema, exigindo abordagens adaptativas e robustas (MCKERROW, 2004; HOFFMANN et al., 2007; BOUABDALLAH et al., 2004; ANDERSON, 1981; LEISHMAN, 2006). Apesar dos esforços, ainda há muito a ser explorado, especialmente no que diz respeito à adaptação dinâmica às incertezas em ambientes imprevisíveis (MICHAEL et al., 2009)(BACHRACH et al., 2009). O controle cooperativo entre múltiplos veículos também é uma área em crescimento, com potencial para resolver muitos dos desafios enfrentados pelos VANT (CASTANON; WOHLLETZ, 2009; OGREN et al., 2004; SEPULCHRE et al., 2008).

A transição para o controle baseado em visão (CBV) oferece novas pers-

pectivas, especialmente em ambientes dinâmicos e imprevisíveis. O CBV utiliza informações visuais para controlar o movimento de um robô, e o controle adaptativo desempenha um papel fundamental nesse contexto, permitindo a adaptação automática às mudanças no ambiente ou no sistema (LIN et al., 2021).

A seguir, serão apresentados alguns trabalhos recentes que exploram a aplicação do controle adaptativo em sistemas baseados em visão. Esses exemplos ilustram a relevância e o potencial dessas abordagens na robótica contemporânea, conforme destacado na tabela 1.

Xie et al. propuseram um controlador IBVS adaptativo para VANT que utiliza momentos de imagem como features visuais e ajusta seus ganhos online para lidar com incertezas no sistema. Sua pesquisa introduz uma nova abordagem IBVS adaptativa, validada por testes de voo real (XIE et al., 2016).

No trabalho de Liu et. al. foi proposto um sistema de rastreamento visual de objetos e controle de servo para quadrotor em nanoescala usando uma câmera monocular. O sistema combina um rastreador de recursos visuais com um controlador não linear para rastrear e seguir um objeto alvo em tempo real. O rastreador usa um filtro de correlação para rastrear o objeto alvo e em seu controlador usa um controle PID para ajustar a posição e a velocidade do quadrotor. A pesquisa contribui demonstrando um sistema de rastreamento e controle baseado em visão para VANT em nanoescala que pode rastrear e seguir um objeto alvo em tempo real (LIU et al., 2021).

Em Croon et. al. apresentam um método para melhorar o controle baseado em fluxo óptico para robôs voadores. O método utiliza aprendizado de máquina para extrair informações visuais do ambiente que auxiliam na estimativa de pose e velocidade. O resultado é um controle mais preciso e robusto em comparação com métodos tradicionais de fluxo óptico. O trabalho contribui para o estado da arte em controle visual ao demonstrar a viabilidade de integrar aprendizado de máquina para melhorar o desempenho do controle (CROON et al., 2021).

Kendou et. al. desenvolve um piloto automático adaptativo baseado em visão para mini VANT usando apenas uma câmera voltada para baixo. Ele combina um odômetro visual adaptativo (VO) com um controlador não linear para navegação e controle em ambientes sem GPS. O VO aproveita o fluxo óptico e os dados inerciais para estimar a posição e a velocidade 3D, permitindo tarefas como

pairar, rastreamento de trajetória e rastreamento de alvo. Esta pesquisa contribui ao demonstrar um piloto automático estável baseado em visão, realizando manobras complexas em ambientes desconhecidos por meio de estimativa e controle adaptativos ([KENDOUL et al., 2009](#)).

No trabalho intitulado "GPS-denied Indoor and Outdoor Monocular Vision Aided Navigation and Control of Unmanned Aircraft" os autores apresentam um Sistema de Navegação Inercial Auxiliados por Visão, em inglês Vision aided Inertial Navigation Systems (V-INS), monocular para controle em malha fechada em tempo real de um VANT instável em ambientes sem GPS. Ele combina recursos visuais e medições inerciais para uma navegação robusta e testou a abordagem para voos autônomos de longo prazo, tanto internos quanto externos. Esta pesquisa avança o que há de mais moderno ao demonstrar V-INS confiáveis, adequados para navegação e controle de VANT negados por GPS no mundo real ([CHOWDHARY et al., 2013](#)).

Em Arif et. al., os autores propõem uma lei de Controle Adaptativo de Tempo Finito (AFTC) para servocontrole visual de um quadrotor pousando em uma plataforma móvel. AFTC combina resposta transitória rápida e alta precisão do controle de tempo finito com aproximação adaptativa para lidar com incertezas. O controlador proposto atinge convergência em tempo finito e supera os métodos existentes em simulações. Esta pesquisa representa a primeira tentativa de usar controle de rastreamento em tempo finito para pouso autônomo de quadrotores em plataformas móveis ([ARIF et al., 2021](#)).

Por fim, Bhargavapuri apresenta um sistema autônomo de rastreamento e pouso baseado em visão para um rotorcraft totalmente acionado. O sistema combina um odômetro visual com um controlador não linear para rastrear um alvo em movimento e pousar de forma autônoma. O odômetro visual usa fluxo óptico para estimar a posição e a velocidade do helicóptero, enquanto o controlador usa uma abordagem de recuo para obter rastreamento e pouso robustos. Esta pesquisa contribui ao demonstrar um sistema baseado em visão que pode rastrear e pousar um helicóptero autônomo em uma variedade de ambientes ([BHARGAVAPURI et al., 2019](#)).

2.3 Detecção de Alvos utilizando Visão Computacional

Atualmente, os UAVs (Veículos Aéreos Não Tripulados) são amplamente estudados na comunidade científica de robótica devido às suas inúmeras aplicações. Eles são empregados em setores como construção para inspeção e monitoramento de segurança (HAM et al., 2016), segurança pública (CALDERÓN et al., 2020), missões de busca e salvamento (KARACA et al., 2018) e inspeção de infraestrutura (SMRCKA et al., 2021). Dadas essas diversas aplicações, é essencial desenvolver métodos robustos e eficientes para UAVs, especialmente para detecção e rastreamento de objetos. Detectar objetos em ambientes naturais apresenta desafios consideráveis devido à grande variação entre objetos do mesmo tipo e mudanças na aparência, iluminação e perspectiva, afetando drasticamente o desempenho dos detectores de objetos (SAQIB et al., 2017a).

Um dos principais desafios nesta área de pesquisa é a detecção autônoma de um UAV intruso dentro de perímetros designados (Stasinchuk et al., 2021). Em locais com requisitos restritos de controle aéreo, como prisões e aeroportos, detectar veículos aéreos não tripulados é crucial para garantir a segurança dos indivíduos e do meio ambiente. A detecção e o rastreamento precisos de UAVs intrusos são essenciais para a segurança e eficiência das operações em ambientes dinâmicos. Uma vez que um UAV intruso é detectado, é imperativo identificar sua posição exata e iniciar o rastreamento contínuo para manter o alvo sob vigilância.

Para resolver esse problema, propomos uma nova abordagem baseada em visão para detectar objetos voadores usando dados visuais obtidos de uma câmera em um UAV autônomo. Usamos um sensor RGB-D para capturar imagens, que são processadas por um algoritmo *You only look once* (YOLO) (REDMON; FARHADI, 2018) para detectar um UAV intruso. Esses dados são então combinados com informações do UAV rastreador por meio de uma Nuvem de Pontos, permitindo a estimativa precisa da posição 3D e da direção do movimento do UAV intruso. Além disso, detalhamos as técnicas de visão empregadas neste trabalho para a detecção e rastreamento de UAV intruso dentro de um perímetro específico. Também fornecemos uma descrição completa do UAV usado, incluindo suas capacidades e configurações.

Nos últimos anos, técnicas de visão computacional têm mostrado grande

potencial na criação de soluções mais eficientes e precisas para detecção de objetos usando drones. Com sistemas avançados de planejamento de missão a bordo, o rastreamento autônomo e a interceptação de alvos são possíveis. Por exemplo, em um artigo de 2015, Rozantsev et al. ([ROZANTSEV et al., 2015](#)) apresentaram uma abordagem que combina estabilização de movimento baseada em regressão com cubos de imagem espaço-temporal para classificação de descritores 3D. Essa técnica permite detecção eficaz mesmo em fundos complexos. A metodologia foi validada em dois conjuntos de dados desafiadores, demonstrando superioridade sobre técnicas de última geração existentes, destacando-se na prevenção de colisões e melhorando algoritmos de rastreamento guiados por visão.

Em 2016, Li et al. ([LI et al., 2016](#)) apresentaram uma abordagem para detecção e rastreamento de Veículos Aéreos Não Tripulados (VANTs) intrusos usando uma única câmera montada em um VANT. A metodologia combina técnicas de subtração de fundo e fluxo óptico para isolar e identificar alvos em movimento, classificando-os com base na comparação entre seus padrões de movimento e movimentos locais. A detecção é refinada usando um filtro de Kalman, que melhora a consistência temporal e reduz detecções e alarmes falsos. A eficácia do método foi demonstrada por meio de experimentos reais e validada com conjuntos de dados de vídeo capturados por UAVs, mostrando que o algoritmo é eficaz na detecção e rastreamento de pequenos UAVs. No mesmo ano, Sapkota Et. Al. ([SAPKOTA et al., 2016](#)), propuseram uma estrutura de vários estágios para a detecção online de pequenos UAVs e a estimativa de suas posições e velocidades relativas a partir de uma única câmera em movimento. Usando uma abordagem baseada em AdaBoost para detecção rápida de objetos e um algoritmo de rastreamento visual online, a metodologia combina estabilização de movimento e fusão de sensores para atingir precisão em tempo real. A abordagem foi validada em ambientes internos onde sua eficácia foi comprovada.

Além disso, Schumann et al. ([SCHUMANN et al., 2017](#)) propuseram uma estrutura de detecção de UAV baseada em imagens de vídeo, usando subtração de fundo mediano para câmeras estáticas e um método de proposta de objeto baseado em aprendizado profundo para câmeras em movimento. A classificação das regiões detectadas é realizada por uma rede neural convolucional (CNN) treinada com um conjunto de dados de imagens de drones e pássaros. A abordagem foi validada em

seqüências de vídeo, demonstrando que o sistema é capaz de distinguir entre drones e pássaros. Além disso, em 2017, Saqib et al. ([SAQIB et al., 2017b](#)) avaliaram técnicas de detecção de objetos baseadas em aprendizado profundo para detecção de drones usando diferentes arquiteturas de rede neural convolucional, como ZF e VGG16. Devido à escassez de dados de treinamento, eles empregaram aprendizado de transferência com modelos ImageNet pré-treinados. Os experimentos mostraram que o VGG16 com Faster R-CNN teve melhor desempenho no conjunto de dados de treinamento. Os autores destacaram que anotar pássaros como uma classe separada pode reduzir falsos positivos, melhorando a diferenciação entre drones e pássaros no desafio Drone-vs-Bird.

Recentemente, Liu et al. ([LIU et al., 2020](#)) propuseram um método para detectar pequenos objetos da perspectiva do UAV com base no YOLOv3 aumentando a operação de convolução nas camadas iniciais para enriquecer as informações espaciais. Eles coletaram um conjunto de dados específico do UAV e propuseram um método de treinamento otimizado. Os resultados experimentais mostraram melhorias significativas na detecção de pequenos objetos, mantendo o desempenho em condições normais. Este trabalho aborda os desafios da detecção de objetos em UAVs, como baixa qualidade de imagem e fundos complexos, e propõe soluções para melhorar a precisão e a robustez da detecção. Da mesma forma, Mishra et al. ([MISHRA et al., 2021](#)) apresentaram um sistema multirrotor com um mecanismo de rede para a coleta autônoma de objetos de superfícies de água usando apenas sensores de bordo. Eles desenvolveram um algoritmo de detecção de objetos de duas fases, que inclui remoção de reflexão e detecção de contorno, bem como um controle de modo deslizante para lidar com incertezas de modelagem. O sistema foi validado em testes ao ar livre, demonstrando uma taxa de sucesso de 91,6%. As contribuições incluem um algoritmo eficiente para detecção de objetos em diferentes condições de iluminação e controle robusto para coleta de objetos sob diversas condições climáticas.

Com relação à estabilidade de detecção, Miranda et al. ([MIRANDA-MOYA et al., 2022](#)) propuseram um controle visual adaptativo de modo deslizante para quadricópteros com o objetivo de rastrear alvos estáticos e dinâmicos, mesmo na presença de perturbações. Usando uma abordagem de câmera virtual e técnicas de servovisão baseadas em imagem, eles melhoraram a robustez e a adaptabilidade do

sistema. A análise de estabilidade foi garantida pela teoria de Lyapunov. Comparações com métodos recentes mostraram a superioridade do sistema em simulações e experimentos reais, incluindo rastreamento de alvos com um Parrot AR. Drone 2.0.

Finalmente, Vrba e Saska ([VRBA; SASKA, 2020](#)) apresentaram um sistema de localização relativa sem marcadores para UAVs, usando uma rede neural convolucional (CNN) para detectar UAVs em imagens de uma câmera de bordo. Este sistema elimina a necessidade de marcadores especializados e hardware extra, tornando-o ideal para UAVs com capacidade de carga útil limitada. A abordagem permite a detecção e localização em tempo real usando um modelo de controle preditivo para estabilizar e coordenar UAVs em formações ou enxames. A metodologia foi validada em experimentos do mundo real, mostrando robustez e eficácia, e serviu como base para a solução proposta neste artigo.

2.4 Considerações Finais

Neste capítulo, realizamos uma análise abrangente dos trabalhos relacionados ao controle adaptativo e à visão computacional para VANTs. A partir de uma revisão detalhada da literatura, identificamos 220 artigos relevantes, o que nos permitiu mapear o estado atual da arte e as abordagens mais promissoras nesse campo. A análise dos estudos existentes demonstrou que, embora avanços significativos tenham sido alcançados, ainda há desafios importantes a serem enfrentados, especialmente em relação à implementação prática das técnicas discutidas.

Dentre as abordagens investigadas, a escolha do controle geométrico baseado em $SE(3)$ para o rastreamento de alvos móveis se mostra uma abordagem estratégica. O $SE(3)$ permite que o controle seja implementado diretamente em espaços de estado não lineares, como o espaço tridimensional com orientações, o que é crucial para a estabilidade e precisão nas manobras de um VANT, especialmente em cenários com movimento agressivo e obstáculos imprevisíveis. A robustez do $SE(3)$ ao lidar com essas variáveis é um dos principais fatores que motivaram a sua escolha para este trabalho.

Essa abordagem, ao integrar a percepção visual com o controle geométrico, oferece uma solução para o rastreamento de alvos em tempo real. A capacidade

de adaptar o controle conforme a movimentação do alvo, aliada à robustez do SE(3), promete ser fundamental para garantir a eficácia do sistema em diferentes condições operacionais.

Em resumo, este capítulo forneceu uma base sólida para a pesquisa que segue, destacando os avanços mais recentes nas áreas de controle geométrico e visão computacional, e apresentando uma direção clara para a aplicação do controle SE(3) no rastreamento de alvos móveis.

Tabela 1 – Abordagens de controle baseadas na visão e seus objetivos.

Objetivo/ Controle	Voo Ágil	Pouso	Rastreamento de Alvo	Abordagens de Controle	Capítulo 2. Trabalhos Relacionados
Controle Adaptativo		(ARIF et al., 2021) (LIU et al., 2021) (BHARGAVAPURI et al., 2019) (CROON et al., 2021) (LIN et al., 2022)	(HE; YUAN, 0)	(KENDOUL et al., 2009) (CHOWDHARY et al., 2013) (XIE et al., 2016) (METNI et al., 2004) (ZHENG et al., 2017) (ALSHAHIR et al., 2023) (LI et al., 2020)	
Controle Backstepping					
Planicidade Diferencial Inversão Dinâmica	(TANG et al., 2018)	(SHAKERNIA et al., 1999) (NABAVI et al., 2022)			
Controle Fuzzy		(BOUAISS et al., 2022) (GHASEMI et al., 2022) (DELBENE et al., 2022)	(OLIVARES-MENDEZ et al., 2015) (ALGABRI et al., 2017)	(MEJIAS et al., 2010) (JURADO et al., 2014) (ARAAR; AOUF, 2015) (NGUYEN et al., 2016)	
Controle Híbrido				(GUENARD et al., 2008a) (BRAS et al., 2008b) (BRAS et al., 2011) (GOMEZ-BALDERAS et al., 2012) (CEREN; ALTUG, 2012)	
IBVS		(SERRA et al., 2016) (TRUONG et al., 2016) (RAKOTOMAMONJY; TRUONG, 2017) (CHEN et al., 2021a) (KEIPOUR et al., 2022)	(ASL; BOLANDI, 2014) (GHOMMAM et al., 2016) (ZARUDZKI et al., 2017) (ZHANG; JI, 2022) (XIE et al., 2022)	(GUENARD et al., 2008b) (BRAS et al., 2008b) (BRAS et al., 2011) (GOMEZ-BALDERAS et al., 2012) (CEREN; ALTUG, 2012)	
Baseado em Lyapunov		(JIANG et al., 2019) (NIU et al., 2022)	(METNI; HAMEL, 2007) (LI et al., 2021) (WANG et al., 2023)	(GUENARD et al., 2008b) (MEBARKI et al., 2015) (LIU et al., 2018)	
MPC	(QIN et al., 2023)	(MARANTOS et al., 2018) (MOHAMMADI et al., 2020) (GUPTA et al., 2023) (ZHU et al., 2023)	(CHEN et al., 2021b)	(KANELLAKIS et al., 2020) (ZHANG et al., 2021) (RECALDE et al., 2022) (LYU et al., 2023)	
Redes Neurais				(WANG et al., 2015) (LOPEZ-FRANCO et al., 2017) (PARK; OH, 2020) (WEI et al., 2022)	
Controle baseado em Observação				(BRAS et al., 2008a) (HASHIM et al., 2023)	
Controle Ótimo		(MARTINEZ et al., 2013) (YU et al., 2007)	(ZHENG et al., 2018) (LIU et al., 2023)	(ZHENG et al., 2019) (ZHANG et al., 2020)	
PBVS			(SHAO et al., 2020)	(CHUANG et al., 2019)	
Controle PID		(CESETTI et al., 2010) (YANG et al., 2013) (ARAAR et al., 2017) (LIN et al., 2017)	(LIN et al., 2009) (GOMEZ-BALDERAS et al., 2013) (KIM et al., 2013) (ROBERTS et al., 2017)	(MURATET et al., 2005) (MEJIAS et al., 2006a) (MEJIAS et al., 2006b) (HUGO et al., 2007)	
RISE				(BIN et al., 2016) (ASL; YOON, 2016)	
Controle Modo Deslizante			(GONZALEZ-GARCIA et al., 2021) (MIRANDA-MOYA et al., 2022)	(ZHAO et al., 2018) (ZHAO et al., 2020)	23
Controle Geométrico				(SANTANA et al., 2016; FRANCHI et al., 2012)	

3 Modelagem Matemática

Neste capítulo, apresentamos a modelagem matemática do sistema, fornecendo uma base teórica para compreender o movimento do quadricóptero, a modelagem do alvo. A formulação matemática apresentada aqui é essencial para a construção dos modelos de controle e para a implementação eficiente das estratégias de rastreamento.

Na Seção 3.1, abordamos os fundamentos da técnica de controle Servo Visual, destacando as abordagens baseadas em posição (PBVS) e em imagem (IBVS), essenciais para a estabilização e rastreamento visual do alvo. Em seguida, na Seção 3.2, discutimos a modelagem do quadricóptero, detalhando sua cinemática e dinâmica, incluindo a influência de forças externas e efeitos aerodinâmicos.

Dessa forma, este capítulo estabelece os fundamentos matemáticos essenciais, fornecendo a base teórica necessária para o desenvolvimento e a implementação de estratégias de controle eficazes no contexto desta pesquisa.

3.1 Servo Visual

Um servocontrole visual (ou visual servoing, em inglês) é uma técnica de controle robótico que utiliza informações visuais (geralmente obtidas por câmeras ou sensores ópticos) para guiar o movimento de um robô ou sistema automatizado. O objetivo é ajustar a posição, orientação ou movimento do robô com base no feedback visual em tempo real, permitindo que ele interaja de forma precisa com o ambiente (CHAUMETTE; HUTCHINSON, 2006)..

Existe duas abordagens principais na teoria do servocontrole visual: o Servo Visual Baseado em Posição (PBVS) e o Servo Visual Baseado em Imagem (IBVS). Essas abordagens são fundamentais para o controle preciso de sistemas robóticos baseados em visão. A seguir, detalharemos a modelagem matemática de cada abordagem, fornecendo uma base sólida para a compreensão de seu funcionamento e implementação em sistemas robóticos.

3.1.1 Servovisual Baseado em Posição (Position-Based Visual Servoing)

Um **Sistema Servovisual Baseado em Posição** (*Position-Based Visual Servoing* - PBVS) é uma técnica de controle que utiliza informações visuais para guiar o movimento de um robô em relação a um alvo. O PBVS opera estimando a **pose** do alvo (posição e orientação) e calculando o movimento necessário para que a câmera do robô alcance a configuração desejada.

Para que o PBVS funcione corretamente, é essencial estimar a pose do alvo de forma precisa para determinar o movimento apropriado da câmera para que ela alcance a pose desejada. Esse processo depende de fatores como a geometria do objeto alvo, os parâmetros intrínsecos da câmera (foco, resolução, etc.) e as características visuais extraídas da imagem capturada, que são fundamentais para a estimativa precisa da pose.

Assim, para determinar o movimento necessário $\xi\Delta$ para que a câmera alcance a pose relativa desejada da câmera em relação ao alvo ${}^{\mathbb{C}^*}\xi_G$ e obter a pose estimada do alvo em relação à câmera ${}^{\mathbb{C}}\hat{\xi}_G$, temos a equação 1.

$$\xi\Delta = {}^{\mathbb{C}}\hat{\xi}_G \ominus {}^{\mathbb{C}^*}\xi_G, \quad (1)$$

A mudança na pose pode ser significativa, portanto, não tentamos fazer esse movimento de uma vez, mas optamos por nos mover gradualmente para um ponto mais próximo de $\{\mathbf{C}^*\}$, da seguinte forma:

$$\xi C\langle k+1 \rangle = \xi C\langle k \rangle \oplus \lambda \xi_{\Delta}\langle k \rangle, \quad (2)$$

que é a representação de uma fração λ , que está no intervalo $(0, 1)$, da tradução e rotação necessárias (CORKE, 2017; HUTCHINSON et al., 1996). Além disso, o referencial $\{\mathbf{C}\}$ é a pose atual da câmera, e o referencial $\{\mathbf{C}^*\}$ é a pose desejada da câmera.

3.1.2 Servovisual Baseado em Imagem (Image-Based Visual Servoing)

O Servovisual Baseado em Imagem, também conhecido como Image-Based Visual Servoing (IBVS), tem se destacado como uma abordagem eficaz para per-

mitir que robôs interajam de forma mais eficiente e intuitiva com o ambiente ao seu redor (CHAUMETTE; HUTCHINSON, 2006). Ao contrário do PBVS, o IBVS dispensa a estimativa da pose relativa do alvo, inferindo-a diretamente dos valores das características visuais presentes na imagem.

3.1.2.1 Movimento da Câmera e da Imagem

Para compreender o funcionamento do IBVS, é fundamental analisar o movimento da câmera e da imagem. Considere uma câmera se movendo com uma velocidade do corpo $\nu = (v, \omega)$ no referencial do mundo, onde $\mathbf{v} = [\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z]^T$ representa a velocidade linear nos eixos $\mathbf{x}, \mathbf{y}, \mathbf{z}$, e $\mathbf{w} = [\mathbf{w}_x, \mathbf{w}_y, \mathbf{w}_z]^T$ representa a velocidade angular nos eixos $\mathbf{x}, \mathbf{y}, \mathbf{z}$, e observando um ponto P no mundo com coordenadas relativas à câmera $\mathbf{P} = (\mathbf{X}, \mathbf{Y}, \mathbf{Z})$. A velocidade do ponto relativa ao referencial da câmera é dada por:

$$\dot{P} = -\omega \times P - v. \quad (3)$$

Expresso em forma escalar, é:

$$\begin{aligned} \dot{X} &= Y\omega_z Z - \omega_y - v_x, \\ \dot{Y} &= Z\omega_x - X\omega_z - v_y, \\ \dot{Z} &= X\omega_y - Y\omega_x - v_z, \end{aligned} \quad (4)$$

Ao analisarmos como a posição do ponto $\mathbf{P}$ se projeta na imagem obtida pela câmera, normalizamos o plano da imagem. Aplicando a regra do quociente e a derivada temporal, obtemos a seguinte forma matricial:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} -\frac{1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & -\frac{1}{Z} & \frac{y}{Z} & 1+y^2 & -xy & -x \end{bmatrix} \cdot \begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}. \quad (5)$$

A Equação 5 representa a relação da velocidade espacial da câmera com a velocidade da característica em coordenadas normalizadas. Reorganizando, temos:

$$x = \frac{\rho_u}{f} \bar{u}, y = \frac{\rho_v}{f} \bar{v}, \quad (6)$$

onde $\bar{u} = u - u^0$ e $\bar{v} = v - v^0$, $(\mathbf{u}, \mathbf{v})$ são as coordenadas do pixel da imagem, $(\mathbf{u}^0, \mathbf{v}^0)$ são as coordenadas do centro da imagem, f é o comprimento focal da câmera em unidades métricas e por fim ρ_u e ρ_v representam o tamanho do pixel em unidades métricas na direção horizontal e vertical, respectivamente. Assim para a derivada temporal, obtemos a seguinte equação abaixo:

$$\dot{x} = \frac{\rho_u}{f} \dot{\bar{u}}, \dot{y} = \frac{\rho_v}{f} \dot{\bar{v}} \quad (7)$$

Substituindo as equações (6) e (7) na equação ((5)), obtemos a equação 8:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} -\frac{f}{\rho_u Z} & 0 & \frac{\bar{u}}{Z} & \frac{\rho_v \bar{u} \bar{v}}{f} & -\frac{f^2 + (\rho_u^2 \bar{u}^2)}{\rho_u f} & \frac{\rho_v \bar{v}}{\rho_u} \\ 0 & -\frac{f}{\rho_v Z} & \frac{\bar{v}}{Z} & \frac{f^2 + (\rho_v^2 \bar{v}^2)}{\rho_v f} & -\frac{\rho_v \bar{u} \bar{v}}{f} & -\frac{\rho_v \bar{v}}{\rho_u} \end{bmatrix} \cdot \begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}. \quad (8)$$

Para simplificar a notação, definimos o comprimento focal expresso em pixels como:

$$f' = \frac{f}{\rho} \quad (9)$$

e reescrevemos a equação 8 como:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \underbrace{\begin{bmatrix} -\frac{f'}{Z} & 0 & \frac{\bar{u}}{Z} & \frac{\bar{u} \bar{v}}{f'} & -\frac{f'^2 + \bar{u}^2}{f'} & \bar{v} \\ 0 & -\frac{f'}{Z} & \frac{\bar{v}}{Z} & \frac{f'^2 + \bar{v}^2}{f'} & -\frac{\bar{u} \bar{v}}{f'} & -\bar{u} \end{bmatrix}}_{J_p(p, Z)} \cdot \begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}. \quad (10)$$

Com isso, temos a relação entre as coordenadas do pixel $p = (u, v)$ na imagem, e a velocidade espacial da câmera ν , da seguinte forma:

$$\dot{p} = J_p(p, Z)\nu, \quad (11)$$

onde J_p é a matriz jacobiana da imagem 2×6 , que relaciona a velocidade de um ponto característico na imagem às velocidades linear e angular da câmera. A matriz depende das coordenadas $p = (u, v)$ que são as coordenadas do pixel na imagem, em relação ao ponto principal (centro óptico da imagem), e Z é a distância do ponto P até a câmera. Essa relação nos permite capturar como o movimento da câmera no espaço 3D afeta o movimento do pixel na imagem.

3.1.2.2 Controlando o Movimento

Para controlar o movimento da câmera de forma eficiente, uma abordagem amplamente utilizada é analisar o deslocamento das características da imagem. Características da imagem são elementos visuais distintivos, como pontos de interesse, bordas, contornos ou formas específicas, que podem ser detectados e rastreados ao longo do tempo. Neste trabalho, a característica de interesse é a forma do drone alvo, que será utilizada como referência para o controle da câmera.

Ao monitorar o deslocamento dessa característica no plano da imagem, é possível determinar como ajustar a posição e a orientação da câmera para alinhá-la com a localização desejada. Essa estratégia permite estabelecer uma relação direta entre a velocidade espacial da câmera e a variação das coordenadas da característica na imagem, conforme descrito pela seguinte equação:

$$\nu = \begin{bmatrix} J_p(p_1, Z_1) \\ J_p(p_2, Z_2) \\ J_p(p_3, Z_3) \end{bmatrix}^{-1} \cdot \begin{bmatrix} \dot{u}_1 \\ \dot{v}_1 \\ \dot{u}_2 \\ \dot{v}_2 \\ \dot{u}_3 \\ \dot{v}_3 \end{bmatrix}, \quad (12)$$

onde $\mathbf{v}$ é a velocidade da câmera, $\mathbf{J}_p(\mathbf{p}_i, \mathbf{Z}_i)$ é a matriz Jacobiana associada a cada ponto característico $\mathbf{p}_i$ considerando a sua profundidade $\mathbf{Z}_i$, e $\dot{\mathbf{u}}_i$ e $\dot{\mathbf{v}}_i$ representam as variações das coordenadas dos pontos característicos na imagem.

Com isso, podemos calcular o movimento necessário da câmera para alinhar as características da imagem às suas posições desejadas. Para isso, é utilizado um controlador linear simples, descrito por:

$$\dot{p}^* = \lambda(p^* - p), \quad (13)$$

onde $\mathbf{p}^*$ é a posição desejada da características da imagem, $\mathbf{p}$ é a posição atual, e λ é um ganho de controle positivo.

Ao expressarmos essa equação em termos da velocidade da câmera iremos obter a seguinte equação:

$$v = \lambda \begin{bmatrix} J_p(p_1, Z_1) \\ J_p(p_2, Z_2) \\ J_p(p_3, Z_3) \end{bmatrix}^{-1} \cdot [p^* - p] \quad (14)$$

Esta lei de controle guiará a câmera para que as características pontuais se movam em direção à posição desejada na imagem. O controlador linear proposto é simples e eficiente para muitas aplicações, mas pode ser facilmente substituído por outras abordagens de controle mais avançadas, dependendo das necessidades específicas do sistema.

3.2 Modelagem do VANT

Nesta seção, exploramos a modelagem do VANT, essencial para compreender seu comportamento dinâmico. O modelo quadricóptero clássico assume uma estrutura rígida em forma de "X" com quatro braços (veja a Figura 4), estabelecendo a convenção de eixos utilizada nesta dissertação. Conforme a Figura 4 ilustra, consideraremos diversos parâmetros e variáveis que descrevem o comportamento dinâmico e cinemático do sistema ((CHLEBOUN et al., 2023)).

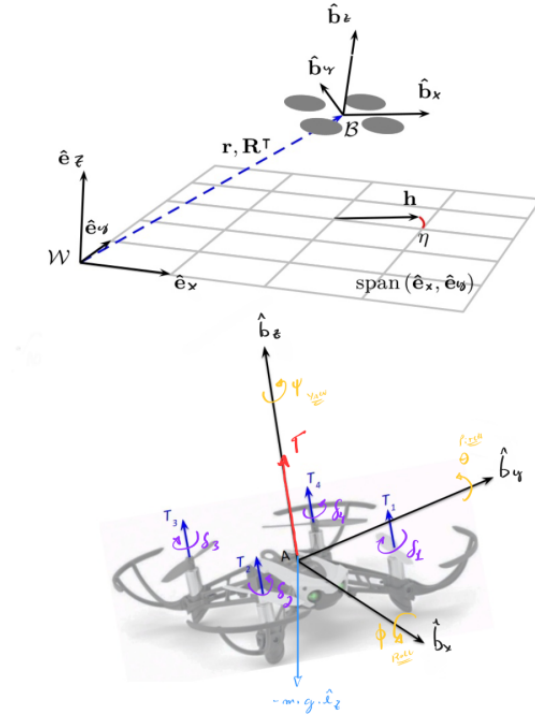


Figura 4 – Convenção dos eixos para o modelo de quadricóptero.

Fonte: Próprio Autor

Adotamos a convenção Tait-Bryan para ângulos, onde rotações em torno de x , y e z são roll (ϕ), pitch (θ) e yaw (ψ), respectivamente. A força responsável pelo movimento translacional do drone é o Thrust (empuxo) T . Em condições ideais, este impulso seria puramente vertical ao longo do eixo z . Contudo, devido aos efeitos aerodinâmicos, o empuxo tende a rotacionar em torno dos eixos x, y e z , como será discutido posteriormente. A Tabela 2 ilustra como as velocidades angulares dos rotores w_1, w_2, w_3 e w_4 que afetam o movimento do VANT, indicando aumentos ou diminuições na velocidade de cada motor.

Tipo de movimento	w_1	w_2	w_3	w_4
Thrust (T_z)	↑	↑	↑	↑
Roll (ϕ)	↓	↑	↑	↓
Pitch (θ)	↓	↓	↑	↑
Yaw (ψ)	↓	↑	↓	↑

Tabela 2 – O impacto da velocidade do rotor no movimento do quadricóptero.

As variáveis de estado do VANT definem sua posição, orientação e velocidade no espaço. Dessa forma, neste trabalho, as variáveis de estado do VANT, no frame do mundo, são convencionadas da seguinte maneira:

$$\mathbf{r} = [x \ y \ z]^T \quad (15)$$

$$\dot{\mathbf{r}} = [\dot{x} \ \dot{y} \ \dot{z}]^T \quad (16)$$

$$\ddot{\mathbf{r}} = [\ddot{x} \ \ddot{y} \ \ddot{z}]^T \quad (17)$$

$$\boldsymbol{\varphi} = [\phi \ \theta \ \psi]^T \quad (18)$$

$$\dot{\boldsymbol{\varphi}} = [\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T. \quad (19)$$

A seguir, detalhamos o significado de cada variável e equação:

- Posição no referencial do mundo ($\mathbf{r}$): é um vetor que define a localização do VANT no espaço em relação a um ponto de referência fixo, nos eixos x,y e z dados em metros [m].
- Velocidade no referencial do mundo ($\dot{\mathbf{r}}$): indica a velocidade linear do VANT em relação ao referencial do mundo, em metros por segundo [m/s].
- Aceleração no referencial do mundo ($\ddot{\mathbf{r}}$): representa a mudança na velocidade do VANT ao longo do tempo no referencial do mundo, em metros por segundo quadrado [m/s²].
- Orientação angular no referencial do mundo ($\boldsymbol{\varphi}$): define a orientação do VANT no espaço, representando a posição angular no referencial do mundo usando a convenção Tait-Bryan (roll, pitch, yaw), em radiano [rad].
- Velocidade angular no referencial do mundo ($\dot{\boldsymbol{\varphi}}$): indica a taxa de mudança da orientação do VANT em relação ao tempo, ou seja, a velocidade com que o VANT está girando em torno dos eixos de roll, pitch e yaw, em radianos por segundo [rad/s].

Além das equações do VANT em relação ao mundo precisamos considerar outras as equações do VANT no frame do corpo como sendo:

$$\mathbf{v} = [v_1 \ v_2 \ v_3]^T \quad (20)$$

$$\boldsymbol{\omega} = [\omega_1 \ \omega_2 \ \omega_3]^T \quad (21)$$

$$\boldsymbol{\delta} = [\delta_1 \ \delta_2 \ \delta_3 \ \delta_4]^T \quad (22)$$

$$\mathbf{T} = [T_x \ T_y \ T_z]^T \quad (23)$$

$$\boldsymbol{\tau} = [\tau_\phi \ \tau_\theta \ \tau_\psi]^T, \quad (24)$$

onde:

- Velocidade linear no referencial do VANT ($\mathbf{v}$): indica a velocidade linear do VANT em relação ao seu próprio corpo, onde v_1, v_2, v_3 representam, respectivamente, a velocidade linear nos eixos X, Y e Z do corpo do VANT, em metros por segundo [m/s].
- Velocidade angular no Referencial do VANT ($\boldsymbol{\omega}$): indica a taxa de rotação do VANT em relação ao seu próprio corpo, onde $\omega_1, \omega_2, \omega_3$ representam, respectivamente, a velocidade angular nos eixos X, Y e Z do corpo do VANT, em radianos por segundo [rad/s].
- Velocidades angulares dos motores ($\boldsymbol{\delta}$): Representa o vetor cujos elementos são as velocidades angulares de cada um dos quatro motores do VANT. os componentes $\delta_1, \delta_2, \delta_3$ e δ_4 representam, respectivamente, a velocidade angular do motor 1, 2, 3 e 4, em radianos por segundo [rad/s].
- Empuxo total no VANT ($\mathbf{T}$): é o vetor gerado pelo somatório dos empuxos dos quatro motores T_1, T_2, T_3 e T_4 , em Newton [N], sendo representado também como o vetor de empuxo nos eixos X, Y e Z, no frame do mundo, onde $\mathbf{T} = [T_x, T_y, T_z]^T$.
- Vetor de torques($\boldsymbol{\tau}$): momento total aplicado ao VANT em relação ao seu centro de massa, onde $\tau_\phi, \tau_\theta, \tau_\psi$ representam, respectivamente, o momento em torno do eixo X do corpo do VANT (roll), o momento em torno do eixo Y do corpo do VANT (pitch) e o momento em torno do eixo Z do corpo do VANT (yaw), em Newton metro [Nm].

Por fim, ressaltamos que, segundo a convenção de Tait-Bryan, as seguintes restrições são aplicadas às variáveis angulares:

$$\phi, \psi \rightarrow [0, 2\pi) \quad (25)$$

$$\theta \rightarrow \left(-\frac{\pi}{2}, \frac{\pi}{2}\right), \quad (26)$$

onde os ângulos de roll e yaw podem variar entre 0 e 2π e o ângulo de pitch é limitado entre $-\pi/2$ e $\pi/2$.

3.2.1 Cinemática

Na análise cinemática do VANT, é fundamental compreender o movimento de translacional e rotacional da aeronave. Abaixo, são apresentadas as equações que descrevem esses aspectos. A velocidade linear $\dot{\mathbf{r}}$ do VANT no referencial inercial é relacionada à velocidade no corpo $\mathbf{v}$ pela matriz de rotação $\mathbf{R}$, conforme a Equação (27):

$$\dot{\mathbf{r}} = \mathbf{R} \cdot \mathbf{v}. \quad (27)$$

Onde a matriz de rotação $\mathbf{R}$ é definida por:

$$\mathbf{R} = \begin{bmatrix} \cos \psi \cos \theta & \cos \psi \sin \theta \sin \phi - \sin \psi \cos \theta & \cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi \\ \sin \psi \cos \theta & \sin \psi \sin \theta \sin \phi + \cos \psi \cos \theta & \sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi \\ -\sin \theta & \cos \theta \sin \phi & \cos \theta \cos \phi \end{bmatrix}. \quad (28)$$

Esta matriz de rotação relaciona as velocidades no sistema de coordenadas do corpo ($\mathbf{v}$) com as velocidades no sistema de coordenadas inercial ($\dot{\mathbf{r}}$), sendo essencial para entender o movimento de translação do VANT.

Do mesmo modo, a relação entre a velocidade angular no referencial do mundo e a velocidade angular no referencial do robô pode ser expressa como:

$$\dot{\boldsymbol{\varphi}} = \boldsymbol{\Gamma} \cdot \boldsymbol{\omega}. \quad (29)$$

Para equação acima, o valor da matriz $\boldsymbol{\Gamma}$ é definida por:

$$\mathbf{\Gamma} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{bmatrix}. \quad (30)$$

Essa matriz $\mathbf{\Gamma}$ é chamada de matriz de acoplamento. Note que essa matriz é dependente apenas de θ e ϕ (ângulos de rotação e de guinada, respectivamente). Em situações onde os ângulos ϕ e θ são aproximadamente zero, essa matriz de acoplamento $\mathbf{\Gamma}$ pode ser aproximadamente igual a 1. Em outras palavras:

$$\mathbf{\Gamma} \approx 1 \quad \text{para} \quad \phi \text{ e } \theta \approx 0. \quad (31)$$

Essa simplificação é útil em situações onde os ângulos de rotação em torno dos eixos de rotação ϕ e θ são pequenos, proporcionando uma relação mais direta entre as variáveis. Entretanto não iremos considerar essa simplificação neste trabalho.

3.2.2 Dinâmica

Esta subseção apresenta as equações fundamentais que governam a dinâmica do quadricóptero. A dinâmica do quadricóptero é a área da robótica que estuda a relação entre as forças e torques que atuam sobre o veículo e seu movimento. A modelagem da dinâmica é crucial para o desenvolvimento de controladores eficazes e simulações precisas do comportamento do quadricóptero.

A primeira equação que descreve a dinâmica translacional do quadricóptero é:

$$\sum_i \mathbf{F}_i = m \cdot \ddot{\mathbf{r}}, \quad (32)$$

onde m é a massa do quadricóptero e $\ddot{\mathbf{r}}$ é a aceleração e $\mathbf{F}_i$ com $i = 1, 2, 3, \dots, n$ são as forças aplicadas no quadricóptero. Essa equação constitui a base para compreender a interação entre as forças aplicadas e o movimento do quadricóptero. Podemos expandir essa equação para considerar as forças individuais, como segue:

$$m \cdot \ddot{\mathbf{r}} = \mathbf{F}_t + \mathbf{F}_g + \mathbf{F}_{DA}, \quad (33)$$

onde $\mathbf{F}_t$ é a força de propulsão, $\mathbf{F}_g$ é a força gravitacional, e $(\mathbf{F}_{DA})$ é a força de arrasto.

A força de arrasto $\mathbf{F}_{DA}$ é uma soma das forças de arrasto $\mathbf{D}_i$ para cada componente do quadricóptero:

$$\mathbf{F}_{DA} = \sum_{i=1}^4 \mathbf{D}_i, \quad (34)$$

onde $\mathbf{D}_i$ é a força de arrasto no i -ésimo componente. Cada vetor $\mathbf{D}_i$ possui uma direção e um sentido específicos em relação ao componente do quadricóptero e ao movimento relativo do ar. A força de arrasto individual $\mathbf{D}_i$ é calculada como:

$$\mathbf{D}_i = \frac{1}{2} \rho C_D A_i (||\mathbf{v}_i||)^2 \quad (35)$$

onde temos que, ρ é a densidade do ar (kg/m^3), C_D é o coeficiente de arrasto do componente, A_i é a área de disco da hélice (m^2) e $||\mathbf{v}_i||$ é a magnitude da velocidade da hélice em relação ao ar (m/s).

Considerando a matriz de rotação $\mathbf{R}$, que depende dos ângulos de Euler, e o vetor unitário vertical $\hat{\mathbf{e}}_z = [0 \ 0 \ 1]^T$, a equação completa das forças que atuam no quadricóptero é:

$$m \cdot \ddot{\mathbf{r}} = \mathbf{T}\mathbf{R} - m \cdot g \cdot \hat{\mathbf{e}}_z - \mathbf{F}_{DA} \cdot \mathbf{R}, \quad (36)$$

onde m é a massa do quadricóptero (kg), $\ddot{\mathbf{r}}$ é a aceleração linear do centro de massa do quadricóptero (m/s^2), $\mathbf{T}$ é o empuxo total gerado pelos motores (N), g é a aceleração da gravidade e $\mathbf{F}_{DA}$ é a força de arrasto total (N).

A soma dos torques que atuam sobre o quadricóptero é igual ao momento de inércia multiplicado pela aceleração angular:

$$\sum_i \mathbf{M}_i = \mathbf{J} \cdot \ddot{\boldsymbol{\varphi}}, \quad (37)$$

onde $\mathbf{J}$ é a matriz diagonal de momento de inércia e $\ddot{\boldsymbol{\varphi}}$ é a aceleração angular (rad/s^2). Esta equação descreve como os torques totais aplicados ao quadricóptero se relacionam à sua aceleração angular.

A equação de Euler no sistema de coordenadas rotacional separa o torque total $\mathbf{J} \cdot \ddot{\boldsymbol{\varphi}}$ em componentes específicos:

$$\mathbf{J} \cdot \ddot{\boldsymbol{\varphi}} = \mathbf{M}_u + \mathbf{M}_t + \mathbf{M}_{gir}, \quad (38)$$

onde $\mathbf{M}_u$ é o torque devido à velocidade angular do corpo do quadricóptero que é o torque é gerado pela interação do corpo do quadricóptero com o ar, $\mathbf{M}_t$ é o torque do thrust e drag o qual é gerado pela diferença de empuxo entre os rotores, e $\mathbf{M}_{gir}$ é o torque do giroscópio gerado pela rotação dos rotores.

O torque $\mathbf{M}_u$ pode ser expresso da seguinte forma:

$$\mathbf{M}_u = \boldsymbol{\Gamma} \cdot (-\boldsymbol{\omega}^x \cdot \mathbf{J} \cdot \boldsymbol{\omega}), \quad (39)$$

onde $\boldsymbol{\Gamma}$ é a matriz de acoplamento, $\mathbf{J}$ é a matriz dos momentos de inércia, $\boldsymbol{\omega}^x$ é a matriz antissimétrica, $\boldsymbol{\omega}$ é o vetor de velocidade angular.

Enquanto o torque do empuxo e arrasto $\mathbf{M}_t$ é:

$$\mathbf{M}_t = \boldsymbol{\Gamma} \cdot \boldsymbol{\tau}, \quad (40)$$

onde $\boldsymbol{\Gamma}$ é a matriz de acoplamento e $\boldsymbol{\tau}$ é o vetor de torque do empuxo (thrust) e do arrasto (drag). O torque do thrust e drag é gerado pela diferença de empuxo entre os rotores. O torque depende da força de propulsão, da geometria do quadricóptero e da velocidade do ar.

O torque do giroscópio $\mathbf{M}_{gir}$ é dado por:

$$\mathbf{M}_{gir} = -\boldsymbol{\Gamma} \cdot \sum_{i=1}^4 (-1)^i \cdot \mathbf{J}_R(\boldsymbol{\omega}^x \cdot \hat{\mathbf{e}}_z), \quad (41)$$

onde temos que $\mathbf{J}_R$ é a matriz de inércia do rotor, $\boldsymbol{\omega}^x$ é a matriz antissimétrica que representa a orientação angular do quadricóptero, $\boldsymbol{\Gamma}$ é a matriz de acoplamento e $\hat{\mathbf{e}}_z$ é o vetor unitário vertical. O torque do giroscópio é gerado pela rotação dos rotores. A matriz de inércia do rotor $\mathbf{J}_R$ depende da geometria e da massa do rotor. O vetor unitário vertical $\hat{\mathbf{e}}_z$ representa a direção vertical. Por simplicidade, iremos desprezar o torque do giroscópio nesse trabalho. Dessa forma, a equação (38) fica sendo

$$\mathbf{J} \cdot \ddot{\boldsymbol{\varphi}} = \mathbf{M}_u + \mathbf{M}_t, \quad (42)$$

Ademais, a matriz antissimétrica ω^x é definida como:

$$\omega^x = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \quad (43)$$

Por fim, a matriz de inércia $\mathbf{J}$ é uma matriz diagonal dada por:

$$\mathbf{J} = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix}, \quad (44)$$

onde, J_x , J_y e J_z do corpo do quadricóptero são os momentos de inércia em torno dos eixos X, Y e Z, respectivamente.

3.2.3 Efeitos Aerodinâmicos

A dinâmica do voo de um quadricóptero é fortemente influenciada por uma série de efeitos aerodinâmicos (e.g., *balde flapping*, arrasto, etc.) e giroscópicos, que desempenham um papel crucial na estabilidade, desempenho e segurança da aeronave. Neste trabalho, a força provevinente dos efeitos aerodinâmicos contemplará apenas o arrasto, conforme equação (34).

O arrasto aerodinâmico é uma força que impacta o movimento de um corpo através de um fluido, como o ar. No contexto de VANTs multirotores, o arrasto influencia diretamente o desempenho da aeronave, afetando sua velocidade, altitude, consumo de energia, capacidade de manobra e estabilidade durante voo pairado e rastreamento de trajetória.

Embora as forças de arrasto sejam significativas, elas têm sido em grande parte negligenciadas no projeto de esquemas de controle, principalmente devido à sua natureza dissipativa que contribui para a estabilidade durante o voo pairado. No entanto, durante o rastreamento de trajetória, essas forças se tornam um distúrbio que, se não for considerado, pode levar a um desempenho de voo subótimo, como trajetórias imprecisas, oscilações indesejadas e maior consumo de energia. (OMARI et al., 2013).

3.2.3.1 Arrasto Induzido

O arrasto induzido está intrinsecamente relacionado às equações de *flapping* das lâminas, um fenômeno complexo que influencia significativamente a dinâmica do rotor. Este tipo de arrasto é gerado pela diferença de pressão entre as superfícies superior e inferior da lâmina, resultando na formação de vórtices nas pontas das lâminas. O arrasto induzido aumenta com o aumento do ângulo de ataque, da razão de impulso do rotor e da velocidade de voo, enquanto diminui com o aumento da envergadura da lâmina.

3.2.3.2 Arrasto Parasita

O arrasto parasita pode ser desmembrado em duas categorias distintas: arrasto de perfil e arrasto de interferência. O arrasto de perfil é gerado pelo atrito da superfície da lâmina com o ar e depende da forma do perfil aerodinâmico, da rugosidade da superfície e do ângulo de ataque. Perfis aerodinâmicos mais espessos, superfícies mais rugosas e maiores ângulos de ataque geram maior arrasto de perfil. Por outro lado, o arrasto de interferência surge da interação entre diferentes partes do VANT, como hélices, fuselagem e naceles. Protuberâncias, irregularidades na superfície e a interação entre as hélices e a fuselagem contribuem para o aumento do arrasto de interferência.

Muitos autores optam por modelar o arrasto parasita como uma única entidade, visto que está principalmente relacionado à forma geral do VANT. Mas neste trabalho, não incluiremos o Arrasto Parasita nas equações.

3.3 Conclusão

Neste capítulo, apresentamos uma análise detalhada da modelagem matemática do sistema, abordando tanto a dinâmica do quadricóptero [3.2.2](#) quanto a modelagem do VANT [3.2](#), ambos essenciais para o desenvolvimento de controladores eficazes. Através da modelagem do quadricóptero, discutimos sua dinâmica, cinemática e os efeitos aerodinâmicos que influenciam o comportamento do VANT durante o voo. Este modelo serve de base para a implementação de técnicas de

controle mais avançadas, permitindo simulações realistas e o desenvolvimento de estratégias robustas para o controle do quadricóptero.

4 Visão

A abordagem de detecção de objetos voadores baseada em visão é apresentada neste capítulo para rastreamento aéreo e ágil altamente dinâmico de objetos voadores. A abordagem usa os dados de entrada de um sensor RGB-D incorporado em um Veículo Aéreo Não Tripulado (VANT) autônomo. A imagem é alimentada em um algoritmo *You only look once* (YOLO) para detecção de objetos e, então, junto com os dados do VANT que está rastreando o objeto voador, é fundida por meio de uma Nuvem de Pontos para estimar o estado do objeto rastreado, ou seja, sua posição 3D e direção. Validamos nossa abordagem proposta em experimentos simulados e do mundo real, comparamos nossa abordagem com a estimativa de estado do VANT rastreado e provamos que nossa detecção é confiável. Nossa abordagem também pode recuperar o rastreamento em caso de reaparecimento do objeto no campo de visão do VANT.

4.1 Detecção de objetos voadores baseada em visão

Neste trabalho, utilizamos a versão 8 do YOLO (You Only Look Once), uma Rede Neural Convolutiva modificada, para detectar uma única classe de objetos. O YOLOv8 foi escolhido por sua capacidade de realizar detecção rápida e precisa, essencial para operações em tempo real, sem sacrificar a precisão (REDMON et al., 2016; REDMON; FARHADI, 2018). A CNN modificada foi integrada ao Sistema Operacional Robótico (ROS) (QUIGLEY et al., 2009) para a detecção e o rastreamento de alvos.

Assim, seja $\mathbf{I}_{RGB} \in \mathbb{R}^{H \times W \times C}$ a imagem de entrada, onde H é a altura da imagem, W é a largura da imagem, e C é o número de canais de cor (geralmente $C = 3$ para imagens RGB-D). A imagem $\mathbf{I}_{RGB}$ é primeiramente redimensionada para uma resolução padrão, ou seja, 1280×720 , utilizando uma interpolação bilinear da seguinte forma:

$$\mathbf{I}_{resized} = \text{resize}(\mathbf{I}_{RGB}, (1280, 720)). \quad (45)$$

Em seguida, o modelo YOLOv8 é utilizado para realizar a inferência na imagem redimensionada. Assim, seja M_{YOLOv8} o modelo treinado. A detecção de objetos na imagem pode ser representada como:

$$\mathbf{O} = M_{\text{YOLOv8}}(\mathbf{I}_{\text{resized}}, \text{conf} = 0.7, \text{classes} = 0), \quad (46)$$

onde $\text{conf} = 0.7$ é o limiar de confiança, indicando que apenas detecções com uma confiança de pelo menos 70% são consideradas, e $\text{classes} = 0$ refere-se à classe específica sendo detectada.

Assim, o resultado $\mathbf{O}$ consiste em um conjunto de N caixas delimitadoras (bounding boxes), cada uma representada por um vetor de quatro componentes:

$$\mathbf{b}_i = [u_i, v_i, w_i, h_i] \quad \text{com} \quad i = 1, 2, \dots, N,$$

onde u_i e v_i são respectivamente as coordenadas do canto superior esquerdo do retângulo na imagem da i -ésima caixa delimitadora, e w_i e h_i são respectivamente a largura e a altura da caixa delimitadora.

Para cada caixa delimitadora $\mathbf{b}_i$, um retângulo R_i é publicado contendo os valores que representam a posição e as dimensões do retângulo no espaço da imagem, de modo que:

$$R_i = \{u_i, v_i, w_i, h_i\}.$$

Esse retângulo é então encapsulado em uma mensagem ROS, que possui as regiões de interesse R_{ROI} e publicado no tópico respectivo do ROS. A Figura 5 mostra o retângulo em simulação e em experimentos no mundo real. A mensagem completa contendo todas as detecções é então representada como:

$$\mathbf{R}_{ROI} = \{R_1, R_2, \dots, R_N\}.$$

Essa mensagem é publicada para ser utilizada por outros nós na arquitetura de navegação baseada em ROS do VANT, permitindo que o sistema do VANT acesse a localização do drone detectado na imagem. Assim, o algoritmo proposto, embarcado no drone, integra detecção e rastreamento de alvos utilizando o modelo YOLOv8 e o framework MRS baseado em ROS. Este sistema consiste em duas implementações principais: uma responsável pelo pré-processamento de imagens

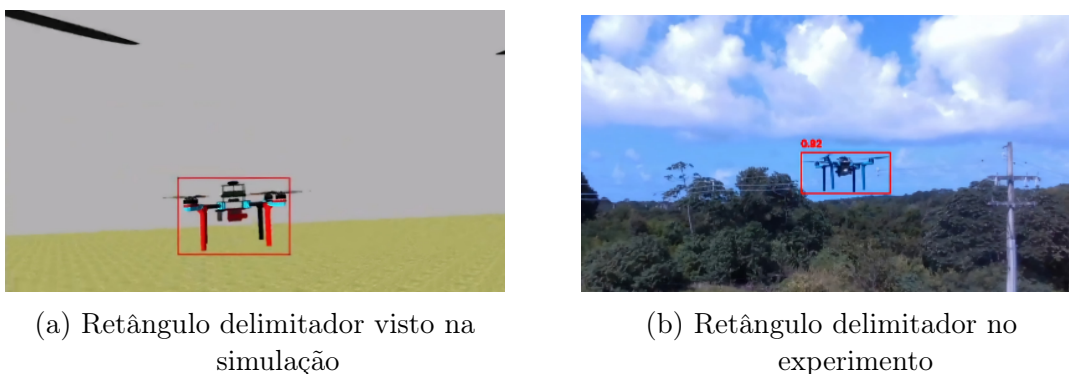


Figura 5 – Retângulo delimitador visto em simulação e experimento real

e inferência da rede neural, e outra que cuida do pós-processamento e integração com os demais sistemas do drone.

4.1.0.1 Treinamento

A detecção de drones usando inteligência artificial (IA) requer um modelo robusto e bem treinado, capaz de operar com precisão em uma variedade de condições. Nesta seção, detalhamos o processo de treinamento de nossa Rede Neural Convolucional (CNN) usando um conjunto de dados especializado para a detecção de VANTs.

Para treinar a CNN, criamos um conjunto de dados especificamente para detectar VANTs, que está disponível publicamente¹. Esse conjunto de dados foi composto por mais de quatro mil imagens rotuladas manualmente, capturadas em diferentes ambientes e condições de iluminação. As imagens foram extraídas de vídeos gravados tanto em simulações quanto em ambientes reais. Em seguida, catalogamos e rotulamos manualmente as imagens, garantindo que cada uma tivesse anotações precisas das posições dos drones.

Em resumo, foram realizados dois passos de treinamento: um usando 4.280 imagens do drone no ambiente simulado para treinar o modelo e executar os experimentos simulados, e outro com 5.000 imagens reais do drone para realizar os experimentos no robô real. O conjunto de dados foi dividido em três partes principais: uma para treinamento, outra para teste e uma para validação (ver Tabela

¹ <https://github.com/LASER-Robotics/train_target_detector>

3). A tabela a seguir resume a estrutura do conjunto de dados:

Tabela 3 – Distribuição do número de imagens para o treinamento de IA

Conjunto	Número de Imagens
Treinamento	2568
Teste	856
Validação	856

O processo de treinamento envolveu várias etapas, incluindo a definição dos hiperparâmetros, como a taxa de aprendizado, o número de épocas e o tamanho do lote (batch size). Abaixo estão os detalhes dos principais hiperparâmetros utilizados:

- Taxa de aprendizado: 0.01
- Número de épocas: 214 (robôs reais) / 300 (simulação)
- Tamanho do lote: 16
- Tamanho da imagem: 640x640
- Momentum: 0.9
- Otimizador: A CNN foi treinada utilizando o algoritmo Stochastic Gradient Descent com momentum (ZHANG, 2004)

Durante o treinamento, o modelo foi periodicamente avaliado usando o conjunto de teste. As métricas de avaliação incluíram a média de precisão (mAP), precisão e recall (ver Tabela 4). Essas métricas nos ajudaram a monitorar o desempenho do modelo e a fazer ajustes conforme necessário.

Tabela 4 – Métricas alcançadas com os melhores valores.

Métrica	Simulação	Experimentos com Robô Real
mAP	0.9950	0.9918
Precisão	0.99839	0.9851
Recall	1.0000	0.9865

O treinamento da CNN utilizando um conjunto de dados especificamente projetado para detecção de VANTs se mostrou eficaz na detecção de drones em

uma variedade de condições. Todo o processo de treinamento foi executado em uma placa gráfica GeForce RTX 3060.

4.1.1 Estimativa da Posição do Alvo

O bloco de Estimativa da Posição do Alvo (veja Fig. 2) recebe a imagem processada pelo bloco de Detecção de Alvo e calcula o centróide do objeto no referencial da câmera. Os retângulos na imagem recebida são regiões de interesse (ROIs) detectadas na imagem RGB. Cada retângulo R_i é representado pelas coordenadas do canto superior esquerdo e suas dimensões: (u_i, v_i, w_i, h_i) , onde u_i e v_i são as coordenadas do canto superior esquerdo do retângulo na imagem, e w_i e h_i são a largura e altura do retângulo, respectivamente.

Assim, para encontrar o centróide do retângulo, primeiramente considere $\mathbf{I}_{RGB}$ como a imagem colorida capturada pela câmera e $\mathbf{I}_D$ como os valores de profundidade para cada pixel, indicando a distância da câmera ao ponto na cena. Utilizando a imagem de profundidade, podemos calcular a coordenada de profundidade Z_i no referencial da câmera do ponto (u_i, v_i) como segue.

$$Z_i = \frac{\mathbf{I}_D(v_i, u_i)}{D_s}, \quad (47)$$

onde $D_s = 1000$ é a escala de profundidade usada para converter os valores de mm para metros.

Onde, podemos calcular as coordenadas (X_i, Y_i) do ponto respectivo no referencial da câmera utilizando o modelo de pinhole da câmera, dado pelas equações a seguir:

$$X_i = \frac{(u_i - c_x) \cdot Z_i}{f_x}, \quad (48)$$

$$Y_i = \frac{(v_i - c_y) \cdot Z_i}{f_y}. \quad (49)$$

Aqui, f_x e f_y são as distâncias focais da câmera em pixels, e (c_x, c_y) são as coordenadas do centro óptico da câmera. Esse processo é repetido para todos os pixels dentro do retângulo R_i , gerando assim uma nuvem de pontos 3D da ROI correspondente ao drone.

Finalmente, para um conjunto de n pontos (X_i, Y_i, Z_i) detectados dentro da ROI referentes ao eixo x, y e z , onde $i = 1 \dots n$, precisamos obter o centróide do retângulo no referencial da câmera ${}^{[c]}\hat{\xi}_{[G]} = [\bar{X}_c, \bar{Y}_c, \bar{Z}_c]^T$ é calculado como:

$$\begin{aligned}\bar{X}_c &= \frac{1}{n} \sum_{j=1}^n X_{i_j}, \\ \bar{Y}_c &= \frac{1}{n} \sum_{j=1}^n Y_{i_j}, \\ \bar{Z}_c &= \frac{1}{n} \sum_{j=1}^n Z_{i_j}.\end{aligned}\tag{50}$$

4.1.2 Navegação Guiada por Percepção

Agora, o bloco de Navegação Guiada por Percepção (veja Fig. 2) recebe o centróide do objeto ${}^{[c]}\hat{\xi}_{[G]}$ do bloco de Estimativa da Posição do Alvo, transforma suas coordenadas para o referencial do mundo que será utilizado como uma posição desejada ($\mathbf{r}_d$), determina os ângulos de orientação (α, β) que existem entre o VANT e o drone alvo (veja Fig. 6) e, finalmente, calcula o ângulo de direção desejado η_d .

Para transformar essas coordenadas para o Referencial do Mundo, utilizamos uma matriz de transformação $\mathbf{T}_{cw}$, com

$$\mathbf{T}_{cw} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}.\tag{51}$$

onde r_{ij} são os elementos da matriz de rotação $\mathbf{R}_{cw}$, e t_x, t_y, t_z são os elementos do vetor de translação $\mathbf{t}_{cw}$.

Finalmente, as coordenadas no referencial do mundo (X_w, Y_w, Z_w) são dadas por

$$\begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \mathbf{T}_{cw} \cdot \begin{bmatrix} \bar{X}_c \\ \bar{Y}_c \\ \bar{Z}_c \\ 1 \end{bmatrix}.\tag{52}$$



Figura 6 – Esquema da projeção de um VANT alvo e sua esfera delimitadora no plano de projeção da câmera $\mathcal{P}$. É mostrada uma visão em perspectiva da situação. O referencial do robô é definido pelo centro de massa do robô O_R e pelos vetores $\hat{b}_1$, $\hat{b}_2$ e $\hat{b}_3$. O referencial da câmera é definido pelos vetores $\hat{c}_1$ e $\hat{c}_2$. O referencial do mundo é definido pelos vetores $\hat{e}_1$, $\hat{e}_2$ e $\hat{e}_3$.

Ou seja,

$$\begin{aligned} X_w &= r_{11}\bar{X}_c + r_{12}\bar{Y}_c + r_{13}\bar{Z}_c + t_x, \\ Y_w &= r_{21}\bar{X}_c + r_{22}\bar{Y}_c + r_{23}\bar{Z}_c + t_y, \\ Z_w &= r_{31}\bar{X}_c + r_{32}\bar{Y}_c + r_{33}\bar{Z}_c + t_z. \end{aligned} \tag{53}$$

Note que a posição desejada é formada por essas coordenadas no referencial do mundo. Ou seja, $\mathbf{r}_d = [X_w, Y_w, Z_w]^T$. Uma matriz semelhante a matriz $\mathbf{T}_{cw}$ chamada $\mathbf{T}_{cr}$ permite a transformação do referencial da câmera para o referencial

do robô com cálculos semelhantes, sendo

$$\begin{bmatrix} X_R \\ Y_R \\ Z_R \\ 1 \end{bmatrix} = \mathbf{T}_{cr} \cdot \begin{bmatrix} \bar{X}_c \\ \bar{Y}_c \\ \bar{Z}_c \\ 1 \end{bmatrix}. \quad (54)$$

Essas coordenadas no referencial do robô serão importantes para calcular os ângulos α e β (veja Fig. 6), que por sua vez serão utilizados para calcular a direção desejada η_d .

No entanto, antes de calcular o ângulo de direção, precisamos desconsiderar os pontos discrepantes que podem aparecer durante a detecção. Para isso, utilizamos a primeira derivada do ponto estimado para filtrar os pontos discrepantes. Assim, chamemos $[\dot{x}, \dot{y}, \dot{z}]^T$ a primeira derivada das coordenadas do centróide do retângulo no referencial do robô $[X_R, Y_R, Z_R]^T$. Então, a velocidade do ponto pode ser calculada como

$$V = \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}. \quad (55)$$

Portanto, todos os pontos em que a velocidade estiver acima de um limiar V_{max} , onde o V_{max} é a velocidade máxima da câmera, não são considerados, e o processo de estimativa termina aqui.

Agora, assumamos que α é o ângulo entre $\hat{b}_1$ e um vetor $\mathbf{d}_y$ formado entre a origem do robô O_R e Y_c , que é a coordenada do VANT alvo em $\hat{c}_2$ no referencial da câmera do drone seguidor. Assumamos também que β é o ângulo entre esse mesmo vetor $\mathbf{d}_y$ e o vetor de distância entre o VANT alvo e o drone seguidor.

De acordo com a Fig. 6, podemos notar uma relação direta entre α e o ângulo de arfagem θ . De acordo com os ângulos de *Tait-Bryan* (comumente confundidos com ângulos de *Euler*), θ é o ângulo de inclinação em torno de $\hat{b}_2$, portanto $\alpha \equiv \theta$, e que, por sua vez, pode ser encontrado como segue

$$\theta = \arctan\left(\frac{Z_R}{X_R}\right) \quad (56)$$

De maneira semelhante, podemos notar uma relação direta entre β e o ângulo de guinada ψ . Assim, o ângulo de guinada ψ é o ângulo de inclinação em

torno de $\hat{b}_3$, portanto $\beta \equiv \psi$, e que, por sua vez, pode ser encontrado como segue

$$\psi = \arctan\left(\frac{Y_R}{X_R}\right) \quad (57)$$

Finalmente, precisamos calcular o ângulo de rotação ϕ , que é o ângulo de rotação em torno de $\hat{b}_1$. Esse ângulo está diretamente relacionado à relação entre as coordenadas Z_R e Y_R no referencial do robô e pode ser calculado como segue:

$$\phi = \arctan\left(\frac{Z_R}{Y_R}\right) - \pi \quad (58)$$

Para garantir que o ângulo ϕ esteja dentro do intervalo padrão $[-\pi, \pi]$, aplicamos a seguinte correção:

$$\text{Se } \phi < -\pi, \text{ então } \phi = \phi + 2\pi$$

Esse ajuste é feito para garantir que o ângulo esteja sempre dentro do intervalo desejado e seja representado corretamente.

Agora podemos calcular a direção desejada η_d , que é o ângulo formado pelo vetor de direção $\mathbf{h}$ (Veja a equação ??), que, por sua vez, é uma projeção de $\hat{b}_1$ no plano $\text{span}(\hat{e}_1, \hat{e}_2)$ (BACA et al., 2021). A direção, conforme definido posteriormente em nosso artigo, é uma alternativa mais intuitiva ao ângulo de *guinada*. Ela remove ambiguidades causadas por várias convenções dos ângulos de *Tait-Bryan* e fornece uma representação mais amigável para o 4º grau de liberdade controlável de um VANT multirrotor. Assim, a direção pode ser calculada como

$$\eta_d = \arctan\left(\frac{\sin \psi \cos \theta}{\cos \psi \cos \theta}\right) \quad (59)$$

O nosso algoritmo de detecção é dividido em duas etapas, a primeira etapa é detectar o alvo, utilizando um modelo pré-treinado do YOLOv8 para identificar aos alvos em um código em Python, e a segunda etapa, é utilizados os dados extraídos da YOLOv8 para obter a posição do alvo.

Este algoritmo explica o processo de detecção de alvos em C++, desde a configuração do nó ROS até a extração das coordenadas do alvo.

4.2 Resultados

Nesta seção, apresentamos e analisamos os resultados de experimentos conduzidos para avaliar o desempenho dos algoritmos propostos sob condições realistas. Para garantir uma avaliação abrangente, conduzimos experimentos em ambientes simulados e do mundo real.

4.2.1 Simulação

Para avaliar o desempenho do algoritmo de detecção de alvos, conduzimos várias simulações em um ambiente de simulação realista². O algoritmo foi integrado a um sistema exclusivo de detecção e rastreamento para mirar e perseguir outros veículos aéreos, permitindo uma avaliação aprofundada de suas capacidades sob condições controladas. Finalmente, a velocidade média de rastreamento na simulação foi de 1,0 m/s.

4.2.1.1 Simulação com um Único Alvo

Durante esta primeira simulação (veja a Figura 5a), o alvo possui uma trajetória predefinida, enquanto o VANT seguidor, equipado com o algoritmo de detecção, foi encarregado de identificar e rastrear o alvo (VANT perseguido) ao longo de todo o caminho. Isso forneceu uma avaliação sistemática da precisão do algoritmo em manter o alinhamento com o alvo em movimento.

Esses experimentos simulados nos permitiram validar nossa metodologia proposta antes de sua aplicação em cenários reais. Ao testar em um ambiente controlado, podemos identificar e mitigar problemas potenciais, garantindo uma transição mais segura e eficaz para aplicações do mundo real.

Assim, como podemos observar na Figura 7, o VANT foi capaz de detectar e manter o drone alvo dentro de seu campo de visão (Veja o Video 3). Esta mesma figura mostra as trajetórias reais tanto do VANT alvo (VANT2) quanto do VANT seguidor (VANT1), bem como as posições estimadas do alvo durante a simulação. A linha azul sólida representa a trajetória do VANT1, a linha vermelha representa as

² <https://gazebosim.org/home>

posições estimadas do VANT2 detectadas pelo algoritmo, e a linha verde representa a trajetória da verdade terrestre do alvo.

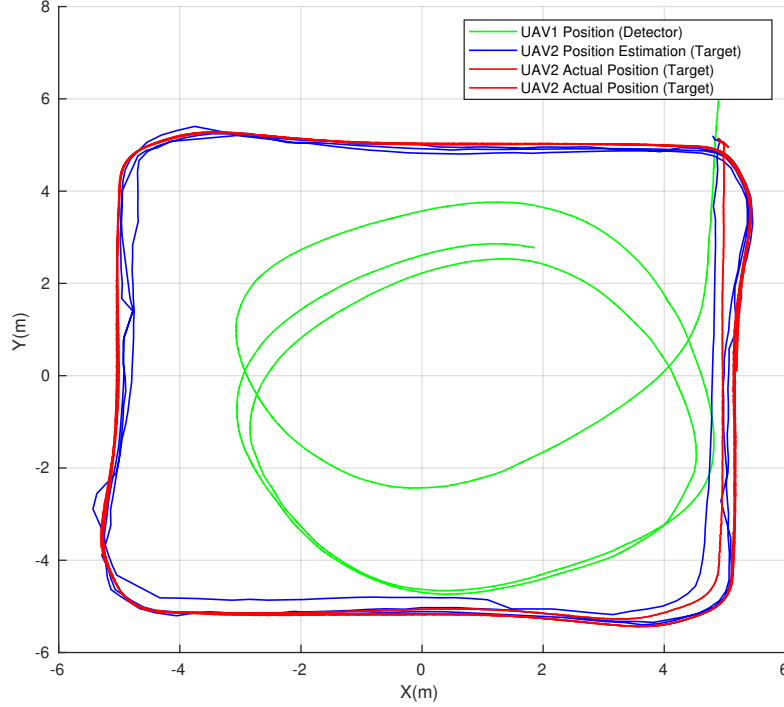


Figura 7 – Visão geral das trajetórias do VANT1 e do VANT2 alvo, e a trajetória estimada do alvo (VANT2) durante as simulações. A linha verde representa a trajetória do VANT1 que está rastreando o alvo. A linha azul mostra o caminho estimado do VANT2 alvo, e a linha vermelha representa a trajetória real do VANT2 alvo.

Além disso, o erro quadrático médio da posição (a distância euclidiana média entre as posições reais do alvo e as posições estimadas pelo algoritmo de detecção) nas coordenadas X, Y e Z é apresentado na Figura 8. O erro também é calculado para cada eixo, fornecendo uma visão abrangente do desempenho do algoritmo.

4.2.1.2 Simulação com Múltiplos Alvos

Para avaliar a capacidade do nosso sistema de lidar com múltiplos alvos em movimento simultaneamente, realizamos uma simulação envolvendo três drones. Como ilustrado na Figura 9, o **VANT1** (drone perseguidor) teve a missão

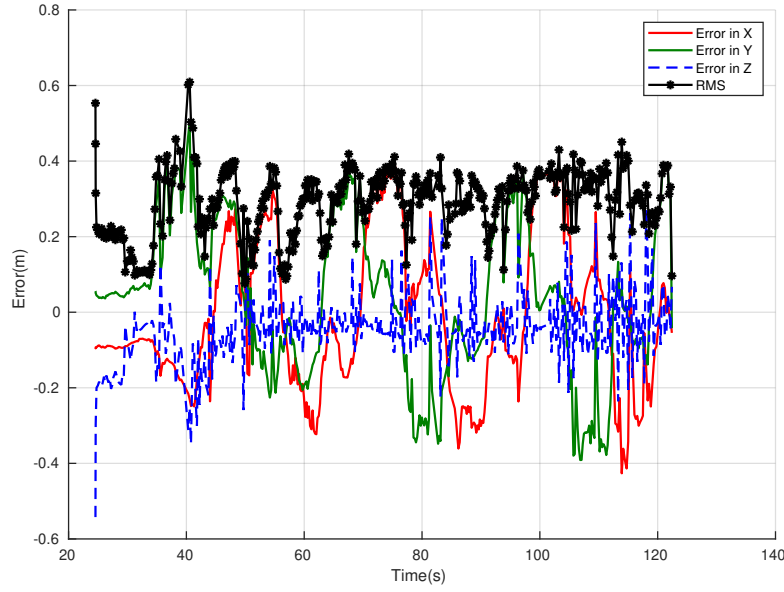


Figura 8 – Erro quadrático médio da posição nas coordenadas X, Y e Z durante as simulações (linha preta). A linha vermelha representa o erro no eixo X, a linha verde representa o erro no eixo Y e a linha tracejada azul representa o erro no eixo Z.

de rastrear o **VANT2** (alvo primário), enquanto um terceiro drone, **VANT3**, movendo-se ao longo de uma trajetória circular (representada pela linha preta).

Neste experimento, o objetivo foi avaliar a robustez do algoritmo de detecção ao lidar com múltiplos alvos em movimento. Conforme mostrado na Figura 9, o **VANT1** (drone de rastreamento) foi capaz de detectar e rastrear o **VANT2** (drone alvo) ao longo de sua trajetória predefinida. Em contraste, outro drone alvo **VANT3** estava na cena com uma trajetória diferente.

Durante a simulação, o algoritmo de detecção adotou dois critérios principais para priorizar o alvo correto:

- **Menor distância** entre o VANT1 (perseguidor) e o alvo identificado.
- **Consistência da detecção**, baseada na estabilidade da caixa delimitadora do VANT2. A caixa delimitadora é uma região retangular que envolve o alvo detectado, utilizada para identificar e rastrear objetos em imagens ou vídeos. Neste caso, a estabilidade da caixa delimitadora indica que o VANT2

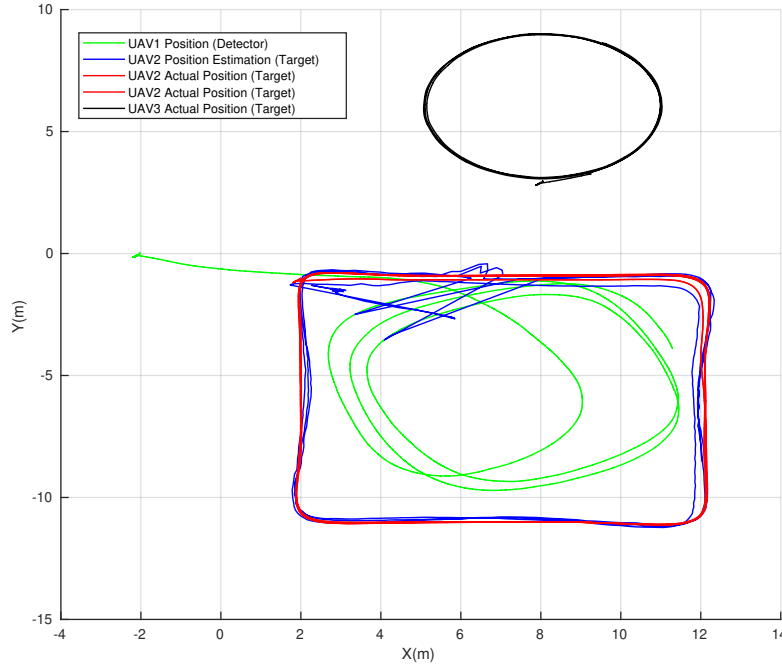


Figura 9 – Trajetórias dos VANTs durante a simulação com dois drones. A linha verde representa a trajetória do VANT1 (drone detector), enquanto a linha vermelha representa a trajetória real do VANT2 (alvo). A linha azul indica a trajetória estimada pelo algoritmo de detecção para o VANT2. A linha preta mostra a trajetória do segundo alvo (VANT3).

continua detectando o alvo de forma consistente, mantendo o desenho da caixa ao longo do tempo.

No Vídeo 3, é possível observar que, em determinados momentos, o **VANT1** detectava temporariamente o **VANT3**. Para evitar que isso comprometesse o rastreamento, o sistema aplicava o critério da menor distância, garantindo que o **VANT1** continuasse focado no **VANT2**. Essa estratégia foi especialmente crucial nos vértices da trajetória quadrada do **VANT2**, onde a proximidade momentânea do **VANT3** poderia induzir um erro no rastreamento. No entanto, o algoritmo demonstrou sua robustez ao se reajustar rapidamente, corrigindo a detecção e retomando o foco no alvo principal, garantindo a continuidade do rastreamento preciso.

Na Figura 10, apresentamos o erro quadrático médio (RMS) da posição do VANT2 em relação às suas coordenadas precisas (X, Y e Z). O gráfico revela picos

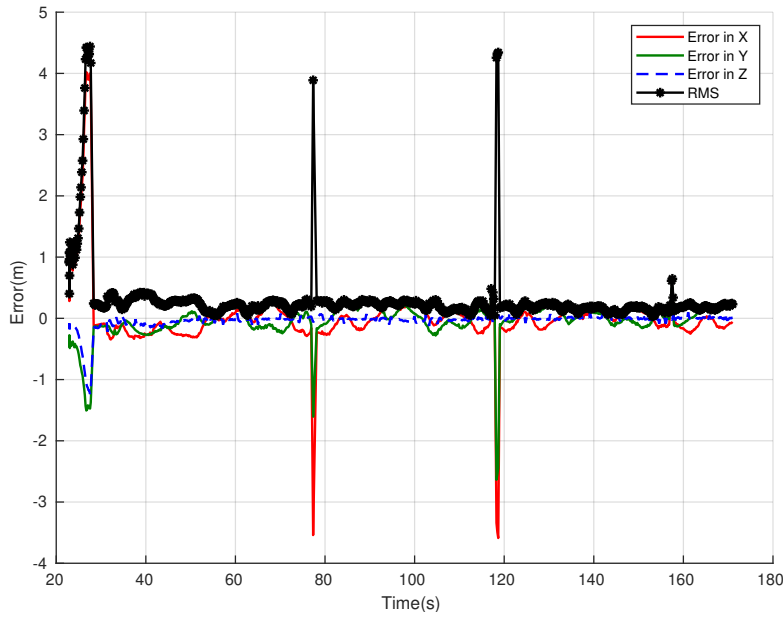


Figura 10 – Erro quadrático médio da raiz da posição nas coordenadas X, Y e Z durante as simulações com dois alvos (linha preta). A linha vermelha representa o erro no eixo X, a linha verde representa o erro no eixo Y e a linha tracejada azul representa o erro no eixo Z.

de erro quando o VANT1 detectou brevemente o segundo drone, principalmente quando o VANT2 estava nos cantos de sua trajetória. Esses picos refletem o ajuste da caixa delimitadora, que se adaptou temporariamente à presença do segundo drone no campo de visão. No entanto, o algoritmo rapidamente se concentrou no alvo correto, demonstrando a resiliência e a capacidade de recuperação do sistema de detecção.

Os resultados desta simulação demonstram que, mesmo em cenários com múltiplos alvos, o sistema pode manter a detecção confiável do alvo desejado, com o RMS permanecendo em níveis aceitáveis para aplicações em tempo real. Esses experimentos reforçam a viabilidade da implementação do sistema em robôs reais e indicam que, em cenários ideais, o algoritmo é capaz de rastrear um alvo em movimento de forma eficaz e precisa. O desempenho geral sugere que o sistema pode ser aplicado em ambientes naturais, onde a detecção robusta e a manutenção do foco em alvos específicos são essenciais.

Ambos os resultados demonstram que nossa abordagem é viável para ser

implementada em robôs reais e, em um cenário ideal, é capaz de detectar e rastrear um alvo móvel desejado. O retângulo delimitador gerado pelo bloco de detecção foi capaz de capturar consistentemente a posição do alvo, conforme evidenciado pelas discrepâncias mínimas mostradas em ambas as figuras. Essas descobertas apoiam a conclusão de que o algoritmo de detecção é robusto e confiável, justificando a transição para testes no mundo real para validação posterior.

4.2.2 Experimentos Reais

Após validar a metodologia por meio de diversas simulações, realizamos uma série de experimentos utilizando um VANT desenvolvido pelo nosso grupo do LASer inspirado no VANT do MRSGroup ([Hert et al., 2022](#)).

O primeiro experimento teve como objetivo avaliar a eficácia do algoritmo proposto na detecção e rastreamento do alvo em um ambiente real, onde ruídos visuais e variações nas características do alvo poderiam dificultar sua identificação. Para isso, utilizamos dois VANTs:

- VANT2(Alvo): no modo manual, sem seguir uma trajetória predefinida, simulando um alvo de comportamento imprevisível.
- VANT1(Seguidor): no modo manual, com a função de identificar e rastrear o VANT2 exclusivamente com base no algoritmo de detecção proposto.

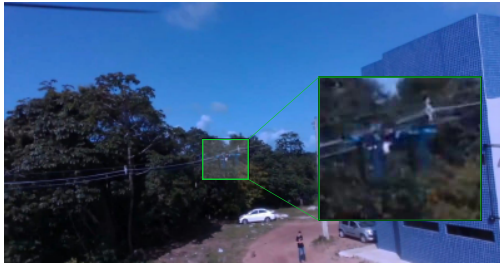
Esse experimento permitiu avaliar a robustez do algoritmo em condições realistas, considerando fatores como variações de iluminação, presença de obstáculos e outras interferências ambientais que poderiam comprometer a precisão da detecção. A partir dos resultados obtidos, foi possível comprovar que o algoritmo de detecção demonstrou eficácia mesmo em um ambiente não controlado, mantendo a capacidade de identificar e rastrear o alvo com consistência.

Em ambos experimentos utilizamos o seguinte VANT que possui os seguintes componentes listados na Subseção 1.4.1 (veja a Figura 1).

A velocidade média de voo nos experimentos reais do robô foi de 0,3 m/s. Além disso, para avaliar a robustez do algoritmo em condições adversas, conduzimos três tipos diferentes de experimentos em um cenário externo para validar a capacidade de detecção da nossa abordagem.

4.2.2.1 Experimentos de robôs reais ao ar livre

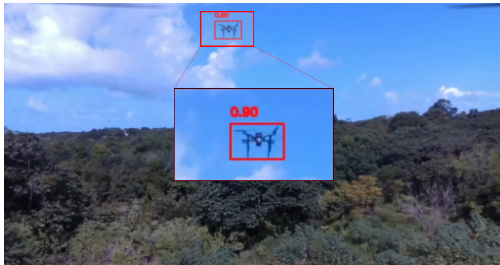
No primeiro experimento (veja a Figura 11), novamente, há duas situações em que o algoritmo falhou em detectar o drone nas imagens capturadas pela câmera (Figuras 11a e 11b). Em ambos os casos, a complexidade do fundo, especialmente devido à presença de cabos de energia, tornou a identificação do alvo um desafio significativo. No entanto, é importante destacar que, embora ambos os VANTs apresentassem muita variação de posição, o algoritmo de detecção conseguiu recuperar a identificação do drone quando o VANT alvo se moveu para uma posição mais favorável. Isso demonstra a capacidade do sistema de se adaptar a variações de contexto, recuperando a eficácia da detecção em condições mais favoráveis.



(a) Sem detecção.



(b) Sem detecção.



(c) Detecção bem-sucedida
(Confiança.: 0.9).



(d) Detecção bem-sucedida
(Confiança.: 0.87).

Figura 11 – Ambas as situações mostram a eficácia do algoritmo na detecção do drone durante o experimento.

Nesta mesma figura, também podemos notar que nossa abordagem foi capaz de detectar o VANT alvo, mesmo à distância, com uma confiança de 0,90, facilitada pelo fundo homogêneo de um céu ensolarado (veja a Figura 11c). Por outro lado, a Figura 11d apresenta uma situação mais desafiadora, onde o drone, igualmente distante, é detectado com uma confiança de 0,87 devido à presença de um fundo

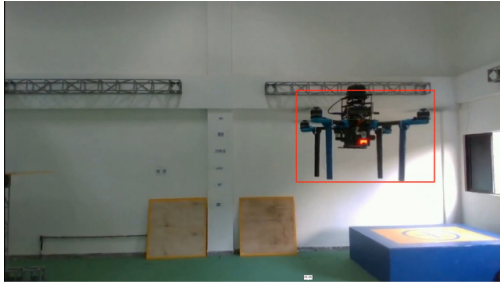
composto por várias árvores. No entanto, apesar da maior complexidade visual, o algoritmo demonstra uma notável capacidade de realizar a detecção, mesmo quando o alvo está significativamente distante.



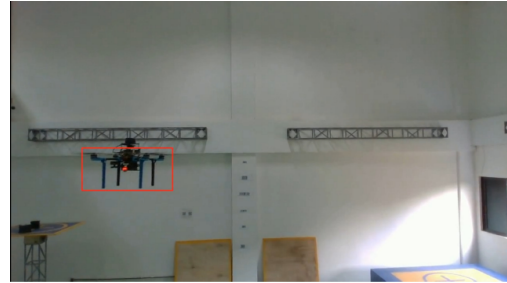
(a) Sem detecção.



(b) Sem detecção.



(c) Detecção bem-sucedida.



(d) Detecção bem-sucedida.

Figura 12 – Experimentos internos realizados utilizando dois VANTs.

4.2.2.2 Experimentos com robôs reais em ambiente interno sem rastreamento visual

Nosso segundo experimento foi realizado em uma arena interna para validar não apenas a abordagem de detecção, mas também a nossa abordagem de estimativa de posição. Durante este experimento, o principal VANT (VANT1) permanece em uma posição fixa enquanto um segundo VANT (VANT2) realiza um trajeto quadrado. Neste experimento, o VANT1 deve observar, detectar e rastrear a trajetória realizada pelo segundo VANT. Note na Figura 12 que agora temos tanto situações de detecção bem-sucedida quanto situações em que o algoritmo não foi capaz de detectar o drone alvo. Essas falhas de detecção ocorrem principalmente devido à oclusão parcial (Figura 12a). Além disso, outros fatores, como baixa qualidade de contraste entre o drone e o fundo, sombras que confundem o

algoritmo, ou a presença de áreas com iluminação inconsistente, podem agravar essa dificuldade de detecção (Figura 12b).

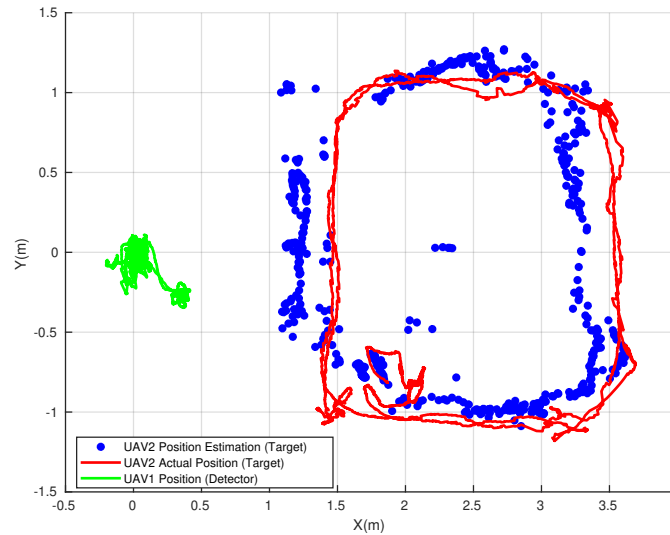


Figura 13 – Imagem de vista superior das trajetórias do drone detector e do drone alvo. A linha vermelha sólida representa a trajetória real do alvo. As cruzes azuis mostram as posições estimadas do drone alvo, e a linha verde representa a posição fixa do drone detector.

A Figura 13 apresenta uma vista superior das trajetórias percorridas pelo VANT1 e pelo drone alvo (VANT2) durante os experimentos. A linha vermelha sólida ilustra a trajetória real do drone alvo, enquanto as cruzes azuis indicam as posições estimadas pela nossa abordagem embarcada no VANT1. Note que, apesar das variações na trajetória estimada (indicadas pelas cruzes azuis), o VANT principal foi capaz de rastrear a posição do alvo, mesmo em algumas regiões de oclusão.

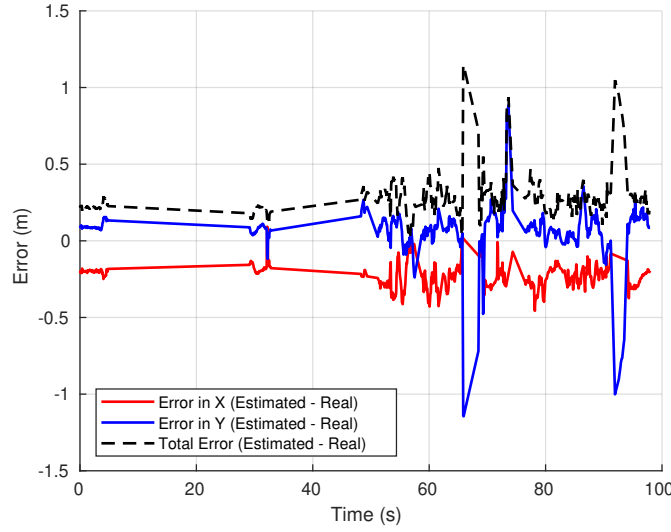


Figura 14 – Erros de posição durante o experimento com robôs reais, sem orientação.

O erro médio durante todo o experimento entre a estimativa e a trajetória real percorrida pelo VANT2 é apresentado na Figura 14. Nesta figura, os dois momentos de alto erro ocorrem devido a uma oclusão parcial, já que, neste experimento, o VANT principal permaneceu estacionário. Esses gráficos nos permitem avaliar qual a precisão do algoritmo de detecção e rastreamento, mostrando que, apesar de algumas flutuações, o sistema conseguiu manter um erro médio relativamente estável ao longo do tempo, demonstrando a eficácia do algoritmo utilizado.

4.2.2.3 Experimentos com robôs reais em ambiente interno com rastreamento visual

Nosso terceiro experimento também foi realizado em ambiente interno. No entanto, o foco principal foi fazer com que a câmera rastreasse continuamente um alvo, garantindo que ele permanecesse no campo de visão da câmera do VANT observador, independentemente do movimento do VANT alvo. O VANT alvo em questão segue uma trajetória quadrada, movendo-se para cima ou para baixo nas bordas do quadrado, validando que o detector consegue rastrear o alvo enquanto mantém sua posição em Z (altura). Além disso, os ângulos de guinada (que controlam a rotação horizontal da câmera) e de inclinação (que controlam a orientação vertical) não foram comprometidos, permitindo que o alvo permanecesse no campo

de visão da câmera o maior tempo possível. O objetivo deste trabalho não foi perseguir o alvo, mas sim rastreá-lo continuamente enquanto ele se move dentro do campo de visão da câmera.

A Figura 15 apresenta o gráfico comparativo entre a trajetória planejada e a posição real do VANT2 (alvo), assim como a posição estática do VANT1 (detector), que manteve o rastreamento do alvo. Na Figura 16, ilustramos os erros de rastreamento ao longo do tempo, destacando as diferenças entre as posições estimadas e reais do alvo. O erro total, apresentado em metros, demonstra que o rastreamento foi bem-sucedido e dentro da margem de tolerância esperada.

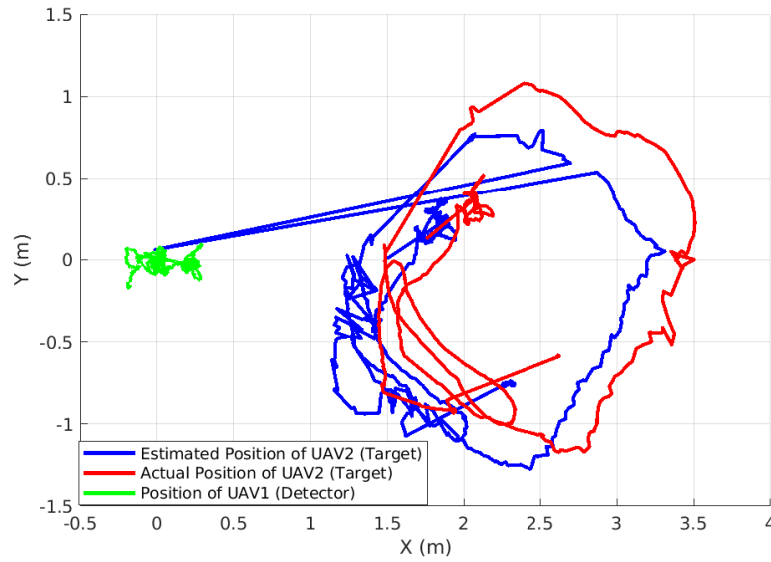


Figura 15 – Imagem de vista superior das trajetórias do drone detector e do drone alvo. A linha vermelha sólida representa a trajetória real do alvo. A linha azul mostra a trajetória estimada do drone alvo, e a linha verde representa a posição do drone detector enquanto altera sua orientação.

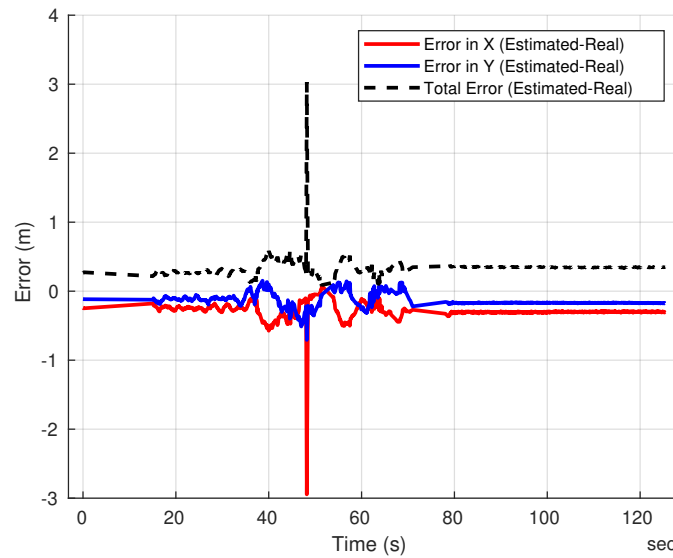


Figura 16 – Erros de posição durante o experimento com robôs reais, com mudança de orientação.

Finalmente, a Figura 17 apresenta o erro de orientação deste segundo experimento com robôs reais. Neste caso, o robô observador gira sua orientação em direção ao VANT alvo, minimizando o erro entre a orientação do robô observador e a orientação estimada dada pela Eq. (59). Podemos ver que o erro máximo é inferior a 34 graus (0,6 radianos), exceto por dois breves momentos em que o robô desaparece e reaparece no final do experimento.

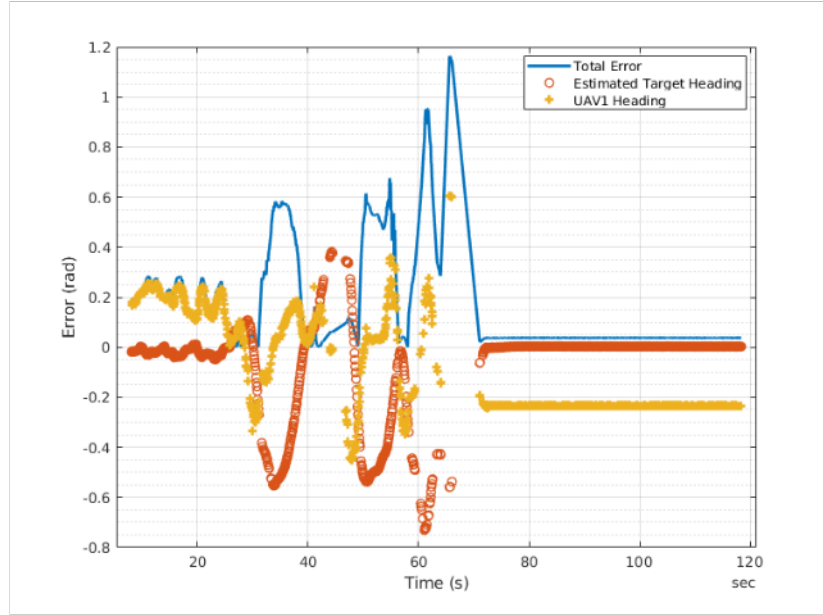


Figura 17 – Erros de orientação durante o experimento com robôs reais, com mudança de orientação.

4.2.2.4 Discussão dos Resultados

Os resultados demonstraram um sistema rápido e preciso que opera acima de 30 Hz. Em resumo, nossa abordagem leva aproximadamente 13,05 ms para realizar a detecção através do YOLOv8, 4,48 ms para a estimativa de posição, e 14,95 ms para sincronizar os dados de profundidade, RGB (cor) e a saída do YOLOv8. Isso resulta em um total de 32,48 ms, independente de o observador estar visualizando o alvo pela primeira vez ou após uma recuperação, o que se deve principalmente à taxa de quadros da câmera.

Após todas as simulações e execuções experimentais com robôs reais, calculamos os valores de RMSE, Precisão e Recall (veja a Tabela 5). Essas métricas nos ajudaram a monitorar o desempenho de nossa abordagem. Observe que o RMSE total em experimentos no mundo real foi menor do que o encontrado no estado da arte (RMSE de (VRBA; SASKA, 2020) = 3,76 m). No entanto, estudos adicionais devem ser realizados, pois nosso alcance de detecção é menor (=4m) do que o encontrado em (VRBA; SASKA, 2020) (=32m).

Outra observação é o RMSE da posição na segunda simulação (com dois VANTs), onde o valor do RMSE da posição foi duas vezes o RMSE da posição dos experimentos e da primeira simulação. Isso ocorre devido aos momentos em que dois VANTs são temporariamente detectados e, por uma fração de segundo, a detecção confunde o VANT rastreado com o segundo que aparece. Além disso, o valor do RMSE de orientação é consideravelmente maior na simulação, pois a velocidade do VANT observado é de 1m/s e a trajetória é maior, o que difere do experimento. Portanto, quanto maior a velocidade do alvo observado, mais difícil é rastreá-lo. Esta última observação é a base de nossos trabalhos futuros.

Tabela 5 – Métricas para Simulações e Experimentos com Robôs Reais.

Métricas	Simulação 1	Simulação 2	Experimento 1	Experimento 2
RMSE (Orientação)	0,9743	0,9457	-	0,2302
RMSE (Posição)	0,3046	0,6264	0,3038	0,3499
Recall	0,9466	0,9651	0,5532	0,9089
Precisão	0,9989	0,9173	0,7551	0,9067

Além disso, observe que a precisão de nossa abordagem, quando o procedimento de rastreamento está ativado (segundo experimento com robôs reais), é muito maior do que no caso de um VANT fixo (primeiro experimento com robôs reais). Esse mesmo comportamento é observado ao comparar o recall dos dois experimentos com robôs reais.

Finalmente, para complementar os resultados apresentados, vídeos das simulações e dos experimentos foram gravados para ilustrar o desempenho do algoritmo em diferentes cenários. Esses vídeos fornecem uma visão detalhada do comportamento do VANT detector e do VANT alvo durante as fases de detecção e rastreamento. Essas gravações estão disponíveis para visualização online e podem ser acessadas através do seguinte **link**³.

4.3 Conclusões

Este capítulo apresentou uma abordagem inovadora de detecção e rastreamento de objetos voadores baseada em visão, com foco na interceptação aérea

³ <<https://youtu.be/8h19DcP8Ufc>>

entre um VANT alvo e um VANT seguidor. Por meio de dados capturados por um sensor RGB-D embarcado em um VANT autônomo, as imagens foram processadas por um algoritmo de detecção de objetos baseado no YOLOv8 e integradas aos dados de navegação do VANT, possibilitando a estimativa da posição e direção relativa entre o VANT seguidor e o VANT alvo em tempo real.

As simulações realizadas demonstraram a confiabilidade, robustez e precisão da metodologia proposta, comprovando sua eficácia para rastrear objetos dentro do campo de visão do VANT. Em complemento, os resultados experimentais mostraram que o sistema proposto é capaz de detectar e rastrear drones de maneira robusta, com baixa latência, o que valida sua aplicabilidade prática em cenários reais de detecção e interceptação aérea. Esses resultados indicam que o sistema tem potencial para ser aplicado em situações que demandem alta precisão e tempo de resposta rápido em operações de vigilância, segurança e monitoramento com drones autônomos.

5 Controle de Postura para Vãos Ágeis

Neste capítulo, apresentamos a arquitetura de controle proposta. Aqui, discutimos o tipo de controle proposto, a metodologia de sintonia, e apresentamos os resultados experimentais para o voo com o drone em simulação. Este capítulo não apenas estabelece as bases matemáticas necessárias para a implementação do controlador adaptativo, mas também apresenta os resultados da implementação das estratégias de controle no contexto desta pesquisa.

5.1 Introdução

Como mencionado na seção 1.4.1 do capítulo 1, este trabalho propõe um sistema de navegação, cujos subsistemas em foco estão representados por blocos laranja e verde na Fig. 18. Neste capítulo, iremos focar no bloco chamado de **Full-state Controller** onde proporemos um controlador Geométrico Adaptativo para Rastreamento de Postura no $SO(3)$, eliminando assim, o controlador de postura da Pixhawk.

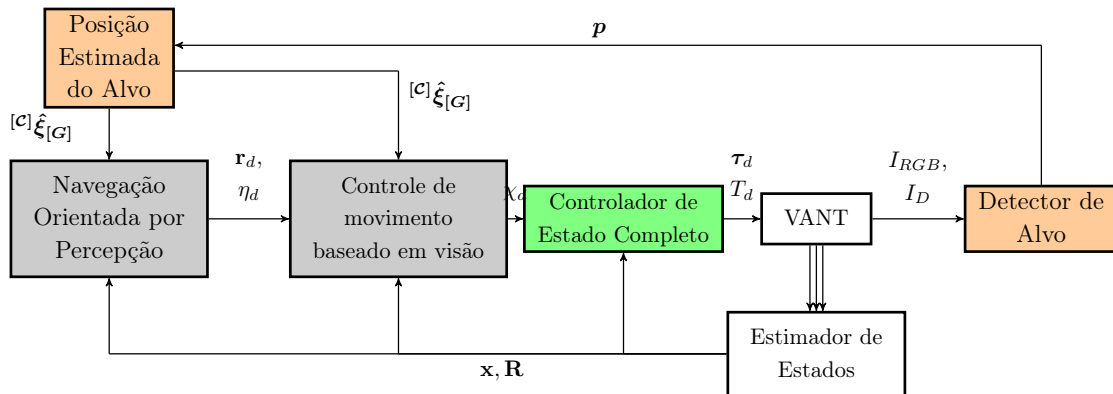


Figura 18 – Sistema de Navegação Proposto.

5.2 Controlador Geométrico de Postura

O Grupo Euclidiano é uma estrutura matemática que representa as transformações no espaço de um objeto. Onde as transformações realizadas neste espaço preservam a distancia euclidiana entre quaisquer dois pontos. Este grupo depende apenas da dimensão n do espaço onde pode ser denotado $E(n)$. No contexto deste trabalho, o corpo rígido a ser estudado será no espaço tridimensional.

Os controladores baseados em ângulos de Euler utilizam três ângulos (roll, pitch e yaw) para descrever a orientação de um corpo rígido no espaço tridimensional. Esses controladores são amplamente utilizados devido à sua simplicidade e intuição física. No entanto, eles apresentam singularidades matemáticas ao representar manobras rotacionais complexas de um VANT (Veículo Aéreo Não Tripulado) quadricóptero, como giros completos ou orientações em que dois eixos de rotação se alinham. Essas singularidades restringem fundamentalmente a capacidade do sistema de seguir trajetórias não triviais, ou seja, trajetórias que envolvem movimentos acrobáticos, mudanças bruscas de direção ou manobras que exigem alta precisão e estabilidade. (LEE et al., 2010).

Em comparação com outras abordagens de controle geométrico para a dinâmica de corpos rígidos, esta é distinta no sentido de que controla um VANT quadricóptero subatuado para estabilizar seis graus de liberdade translacionais e rotacionais usando quatro entradas de empuxo, enquanto rastreia assintoticamente quatro saídas, consistindo em sua posição e direção do cabeçote. Demonstramos que este controlador é particularmente útil para manobras complexas e acrobáticas de um VANT quadricóptero, como a recuperação após estar inicialmente de cabeça para baixo (LEE et al., 2010).

5.2.1 Modelo Dinâmico do VANT

Entretanto, antes de adentrarmos nas equações do controlador geométrico no $SO(3)$, vamos revisar a dinâmica do VANT dentro do espaço Euclidiano. Consideremos então o VANT da Fig. 19. A configuração deste VANT é definida pela localização do centro de massa e a atitude em relação ao referencial inercial. Portanto, a variedade de configuração é o grupo euclidiano especial $SE(3)$, que é o pro-

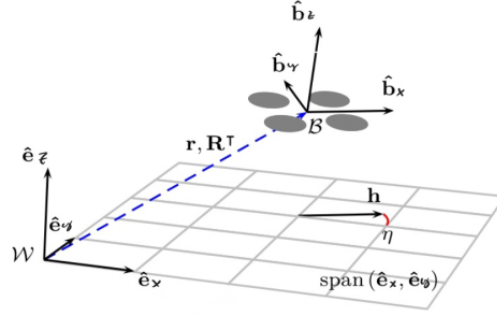


Figura 19 – Convenção dos eixos para o modelo de quadricóptero: A imagem descreve o frame do mundo $\mathcal{W} = \{\hat{\mathbf{e}}_x, \hat{\mathbf{e}}_y, \hat{\mathbf{e}}_z\}$ no qual a posição 3D e a orientação do corpo VANT são expressas. O frame do robô $\mathcal{B} = \{\hat{\mathbf{b}}_x, \hat{\mathbf{b}}_y, \hat{\mathbf{b}}_z\}$ se relaciona com $\mathcal{W}$ pela translação $\mathbf{r} = [x, y, z]^T$ e pela rotação $\mathbf{R}^T$. O vetor de direção do VANT, $\mathbf{h}$, que é uma projeção de $\hat{\mathbf{b}}_x$ para o plano $\text{span}(\hat{\mathbf{e}}_x, \hat{\mathbf{e}}_y)$, forma o ângulo de direção $\eta = \text{atan2}(\hat{\mathbf{b}}_x^T \hat{\mathbf{e}}_y, \hat{\mathbf{b}}_x^T \hat{\mathbf{e}}_x) = \text{atan2}(\mathbf{h}_y, \mathbf{h}_x)$.

Fonte: (BACA et al., 2021)

duto semidireto de $\mathbb{R}^3$ e do grupo ortogonal especial $SO(3) = \{\mathbf{R} \in \mathbb{R}^{3 \times 3} | \mathbf{R}^T \mathbf{R} = \mathbf{I}, \det \mathbf{R} = 1\}$. Ademais, também considerando a equação do somatório das forças (36), bem como a equação do momentum atuante no VANT (42), (39) e (40)), teremos resumidamente que o modelo dinâmico não-linear de movimento do VANT pode ser dividido em duas partes:

uma parte de translação

$$m\ddot{\mathbf{r}} = \mathbf{T}\mathbf{R}\hat{\mathbf{e}}_z - mg\hat{\mathbf{e}}_z - \mathbf{F}_{DA}, \quad (60)$$

e uma parte rotacional

$$\begin{aligned} \dot{\mathbf{R}} &= \mathbf{R}\boldsymbol{\omega}^x, \\ \boldsymbol{\tau} &= \mathbf{J}\dot{\boldsymbol{\omega}} + (-\boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega}), \end{aligned} \quad (61)$$

onde $\boldsymbol{\omega}^x$ é o tensor da velocidade angular, sob a condição $\boldsymbol{\omega}^x \mathbf{v} = \boldsymbol{\omega} \times \mathbf{v}, \forall \mathbf{v} \in \mathbb{R}^3$.

Durante o voo, o veículo experimenta aceleração gravitacional descendente com magnitude $g \in \mathbb{R}$ juntamente com a força de empuxo T criada coletivamente pelas hélices na direção de $\hat{\mathbf{b}}_z$. No entanto, pode-se considerar e estimar separadamente o azimuth do eixo $\hat{\mathbf{b}}_x$ no mundo como a *direção* do VANT. Sob a condição de $|\hat{\mathbf{e}}_z^T \hat{\mathbf{b}}_x| > 0$, definimos a *direção* como sendo:

$$\eta = \text{atan2} \left(\hat{\mathbf{b}}_x^\top \hat{\mathbf{e}}_y, \hat{\mathbf{b}}_x^\top \hat{\mathbf{e}}_x \right). \quad (62)$$

A direção η é uma alternativa mais intuitiva ao ângulo *yaw* (ψ) amplamente utilizado como um dos 4 graus de liberdade controláveis. É possível usar o *yaw*, mas com a suposição de que a inclinação do VANT ($\cos^{-1} \hat{\mathbf{b}}_z^\top \hat{\mathbf{e}}_z$) é baixa, quase horizontal. Aconselhamos contra o uso dos *ângulos de Tait-Bryan* (comumente confundidos com *ângulos de Euler*), devido ao número esmagador de convenções, que frequentemente levam a mal-entendidos. Geralmente, o ângulo *yaw* amplamente usado (como em *ângulos de Euler*, *ângulos de Tait-Bryan* (DIEBEL, 2006)) não tem significado direto com relação à orientação particular de nenhum dos eixos do corpo em nenhuma das convenções, uma vez que a orientação final também depende das duas rotações restantes (inclinação, rotação). Portanto, definimos o vetor de direção pelo eixo $\hat{\mathbf{b}}_x$ como:

$$\mathbf{h} = \begin{bmatrix} \mathbf{R}(1, 1) & \mathbf{R}(2, 1) & 0 \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{b}}_x^\top \hat{\mathbf{e}}_x & \hat{\mathbf{b}}_x^\top \hat{\mathbf{e}}_y & 0 \end{bmatrix} \quad (63)$$

e sua forma normalizada é tal que

$$\hat{\mathbf{h}} = \frac{\mathbf{h}}{|\mathbf{h}|} = \begin{bmatrix} \cos, \eta & \sin, \eta & 0 \end{bmatrix}. \quad (64)$$

5.2.2 Projeto do Controlador

Neste trabalho, desenvolvemos um controlador baseado no trabalho de (LEE et al., 2010) para minimizar o erro de controle em torno da referência de controle desejada χ_d ¹ e fornecer referência de controle de baixo nível ao sistema embarcado. A referência de controle de baixo nível produzida por um controlador dentro do nosso pipeline consiste nos torques desejados do corpo do VANT $\boldsymbol{\tau}_d \in \mathbb{R}^3$ e no empuxo desejado $\mathbf{T}_d \in [0, 1]$. Dessa forma, para um dado comando translacional $\mathbf{r}_d$ e uma direção desejada η_d , selecionamos o empuxo total desejado $\mathbf{T}_d$, e a rotação desejada $\mathbf{R}_d$ para estabilizar a dinâmica do VANT (LEE et al., 2010)..

A dinâmica translacional de um VANT é controlada pelo empuxo total $-\mathbf{T}\hat{\mathbf{R}}\hat{\mathbf{e}}_z$, onde a magnitude do empuxo total $\mathbf{T}$ é controlada diretamente, e a

¹ a referência χ é fornecida pelo bloco *Controle de movimento baseado em Visão* e que neste trabalho é apenas um controlador Proporcional-Derivativo

direção do empuxo total $-\mathbf{R}\hat{\mathbf{e}}_z$ é ao longo do terceiro eixo fixo do corpo $-\mathbf{b}_z$. Esse eixo representa a direção oposta ao eixo vertical do robô no referencial do corpo, que geralmente aponta para cima no sistema de coordenadas do próprio VANT. Dessa forma como mostrado na Equação (60), a resultante do controle de força será dado por $-\mathbf{T}\mathbf{R}\hat{\mathbf{e}}_z \in \mathbb{R}^3$. A magnitude do empuxo $\mathbf{T}$ pode ser arbitrariamente ajustada através na matriz de mixagem, mas a direção deverá ser $-\mathbf{R}\hat{\mathbf{e}}_z$ no frame do mundo, a qual é oposta ao terceiro eixo no frame do robô, $\mathbf{b}_z$. Dessa forma, para controlar a dinâmica translacional, a atitude deveria ser também controlada da mesma forma.

Nesse trabalho, nós assumimos que essa restrição é desprezível, podendo obter a resultante do controle de força através de um ajuste arbitrário. Em termos específicos, supomos que o termo $\mathbf{T}$ pode ser modificado para formar uma entrada de controle $\mathbf{F}_c$, na qual a saída deverá ser $-\mathbf{F}_c\mathbf{R}\hat{\mathbf{e}}_z \in \mathbb{R}^3$. Para projetarmos uma entrada de controle $\mathbf{F}_c$, deveremos primeiro considerar as variáveis de erro como sendo:

$$\begin{aligned}\mathbf{e}_p &= \mathbf{r} - \mathbf{r}_d, \\ \mathbf{e}_v &= \dot{\mathbf{r}} - \dot{\mathbf{r}}_d, \\ \mathbf{e}_a &= \ddot{\mathbf{r}} - \ddot{\mathbf{r}}_d,\end{aligned}\tag{65}$$

Diferentemente de (LEE et al., 2010), aqui assumimos que a referência χ_d enviada ao controle de postura é tal que

$$\chi_d = [\mathbf{r}_d, \dot{\mathbf{r}}_d, \ddot{\mathbf{r}}_d, \eta_d]^\top.\tag{66}$$

Dessa forma, teremos a lei de controle para o empuxo como sendo

$$\mathbf{F}_c = -k_p\mathbf{e}_p - k_v\mathbf{e}_v - k_a\mathbf{e}_a + mg\hat{\mathbf{e}}_z + \mathbf{F}_{DA} + m\ddot{\mathbf{r}},\tag{67}$$

onde k_p é o ganho associado aos erros de posição, k_v é o ganho associado ao erro de velocidade e k_a o ganho associado ao erro de aceleração.

Essa lei de controle combina Nonlinear Dynamic Inversion (NDI) e controle de erro. O NDI é utilizado para cancelar as dinâmicas não lineares do sistema, como a força gravitacional $mg\hat{\mathbf{e}}_z$ e a força de arrasto aerodinâmico $\mathbf{F}_{DA}$, além de impor a aceleração desejada $m\ddot{\mathbf{r}}$. Já o controle de erro $-k_p\mathbf{e}_p - k_v\mathbf{e}_v - k_a\mathbf{e}_a$ atua para

minimizar os desvios em relação à trajetória desejada, garantindo a convergência do sistema para o estado de referência.

Por fim, o empuxo desejado $\mathbf{T}_d$ será apenas a componente z da entrada de controle. Dessa forma,

$$\mathbf{T}_d = -\mathbf{F}_c \mathbf{R} \hat{\mathbf{e}}_z. \quad (68)$$

Ao entendermos a parte translacional do movimento de um VANT, é essencial também compreender o movimento de rotação, pois ambos estão intrinsecamente relacionados no controle de veículos aéreos. Enquanto a dinâmica translacional lida com a posição, velocidade e aceleração do VANT no espaço, a dinâmica rotacional descreve a orientação do veículo, representada pela matriz de rotação R .

Para garantir que o VANT siga a trajetória desejada, é necessário rastrear não apenas os erros de posição, velocidade e aceleração, mas também os erros de atitude (orientação) e suas respectivas derivadas. A atitude desejada é definida pela matriz de rotação R_d , e a atitude atual do VANT é representada por R . Para quantificar o erro de atitude, utilizamos uma função de erro no grupo de rotação $SO(3)$, que mede a discrepância entre R e R_d .

$$\Psi(\mathbf{R}, \mathbf{R}_d) = \frac{1}{2} \text{tr}[\mathbf{I} - \mathbf{R}_d^T \mathbf{R}]. \quad (69)$$

Essa é uma função positiva-definida localmente quando $\mathbf{R} = \mathbf{R}_d$ dentro da região onde a rotação entre os ângulos de $\mathbf{R}$ e $\mathbf{R}_d$ são menores que 180° . Ademais, quando a variação da matriz de rotação é expressa como $\alpha \mathbf{R} = \mathbf{R} \hat{\epsilon}$, para $\epsilon \in \mathbb{R}^3$, a derivada da função do erro é calculada como

$$\mathbf{D}_R \Psi(\mathbf{R}, \mathbf{R}_d) \cdot \mathbf{R} \hat{\epsilon} = \frac{1}{2} (\mathbf{R}_d^T \mathbf{R} - \mathbf{R}^T \mathbf{R}_d)^\vee \cdot \epsilon. \quad (70)$$

onde o sinal *vee* em $^\vee : SO(3) \rightarrow \mathbb{R}^3$ é a inversa do sinal $\hat{\cdot}$.

Dessa forma, o erro de atitude ($\mathbf{e}_R$) pode ser calculado como

$$\mathbf{e}_R = \frac{1}{2} (\mathbf{R}_d^T \mathbf{R} - \mathbf{R}^T \mathbf{R}_d)^\vee. \quad (71)$$

Por sua vez, os vetores tangente $\dot{\mathbf{R}} \in TrSO(3)$ e $\dot{\mathbf{R}}_d \in TrSO(3)$ não podem ser diretamente comparados, uma vez que eles estão em espaços tangenciais

diferentes. Sendo assim, para calcular o erro de taxa de variação da atitude ($\mathbf{e}_\Omega$), temos de acordo com (LEE et al., 2010) que

$$\mathbf{e}_\Omega = \boldsymbol{\omega} - \mathbf{R}^T \mathbf{R}_d \boldsymbol{\omega}_d. \quad (72)$$

E segundo (INVERNIZZI et al., 2020) podemos adicionar o termo integrativo, culminando numa lei de controle para calcular os torques desejados tal que

$$\boldsymbol{\tau}_d = -k_R \mathbf{e}_R - k_\omega \mathbf{e}_\Omega - k_{\omega_I} \mathbf{e}_{\dot{\Omega}} + \boldsymbol{\omega} \times (\mathbf{J} \boldsymbol{\omega}) + \mathbf{J}(\boldsymbol{\omega}^x \mathbf{R}^T \mathbf{R}_d \boldsymbol{\omega}_d - \mathbf{R}^T \mathbf{R}_d \dot{\boldsymbol{\omega}}_d), \quad (73)$$

onde onde k_R é o ganho associado ao erro de atitude, k_ω é o ganho associado ao erro de velocidade angular e k_{ω_I} o ganho associado a derivada do erro de velocidade angular.

5.2.2.1 Cálculo de $\mathbf{R}_d$

Para controlar a orientação de um VANT, é necessário definir uma matriz de orientação desejada $\mathbf{R}_d$. Essa matriz descreve como o veículo deve se orientar no espaço para alcançar os objetivos de controle. Para isso, utilizamos dois vetores principais: o vetor força desejado $\mathbf{f}_d$ (definido pela Equação ??) e o vetor de direção desejado $\hat{\mathbf{h}}_d$ (definido pela Equação ??).

A abordagem original para construir $\mathbf{R}_d$, proposta por (LEE et al., 2010) é viável, mas apresenta uma limitação importante: ela não mantém a direção η durante as manobras. Isso porque, quando compomos $\mathbf{R}_d$ com os vetores

$$\hat{\mathbf{b}}_{z_d} = \frac{\mathbf{f}_d}{\|\mathbf{f}_d\|}, \quad \hat{\mathbf{p}}_{y_d} = \frac{\hat{\mathbf{b}}_{z_d} \times \hat{\mathbf{h}}_d}{\|\hat{\mathbf{b}}_{z_d} \times \hat{\mathbf{h}}_d\|}, \quad \hat{\mathbf{p}}_{x_d} = \hat{\mathbf{p}}_{y_d} \times \hat{\mathbf{b}}_{z_d}, \quad (74)$$

onde $\hat{\mathbf{b}}_{z_d}$ é a direção da força desejada, $\hat{\mathbf{p}}_{x_d}$ e $\hat{\mathbf{p}}_{y_d}$ são vetores ortogonais que definem o plano perpendicular a $\hat{\mathbf{b}}_{z_d}$.

No entanto essa abordagem não preserva a direção η porque o azimuth do eixo $\hat{\mathbf{p}}_{x_d}$ geralmente não é igual ao azimuth de $\hat{\mathbf{h}}_d$. Isso ocorre porque $\hat{\mathbf{p}}_{x_d}$ é obtido como uma projeção ortogonal de $\hat{\mathbf{h}}_d$ no subespaço

$$\mathcal{L}_d = \text{span} \left(\hat{\mathbf{b}}_{z_d} \right)^\perp, \quad (75)$$

o que pode alterar a direção original.

Portanto, para resolver essa limitação, utilizaremos a forma proposta por (BACA et al., 2021), projetada explicitamente para facilitar o controle do ângulo de direção. Para isso, precisamos abordar o controle de torque por construir a matriz de orientação desejada como

$$\mathbf{R}_d = [\hat{\mathbf{b}}_{x_d}, \hat{\mathbf{b}}_{y_d}, \hat{\mathbf{b}}_{z_d}]. \quad (76)$$

Neste controlador proposto, o vetor força desejado $\mathbf{f}_d$ é encontrado através da equação (67) como sendo $\mathbf{F}_c$. Sendo assim, teremos que

$$\hat{\mathbf{b}}_{z_d} = \frac{\mathbf{F}_c}{\|\mathbf{F}_c\|}, \quad (77)$$

Podemos também encontrar $\hat{\mathbf{b}}_{x_d}$ como uma projeção oblíqua de $\hat{\mathbf{h}}_d$ na direção de $\hat{\mathbf{e}}_z$ para o subespaço $\mathcal{L}_d$. Esta projeção é construída como

$$\mathbf{b}_{x_d} = \mathbf{O}(\mathbf{P}^\top \mathbf{O})^{-1} \mathbf{P}^\top, \quad \hat{\mathbf{b}}_{x_d} = \frac{\mathbf{b}_{x_d}}{\|\mathbf{b}_{x_d}\|}, \quad (78)$$

onde $\mathbf{O} \in \mathbb{R}^{3 \times 2}$ é o projetor ortogonal para $\mathcal{L}_d$ (construído, por exemplo, como as duas primeiras colunas de $\mathbf{I} - \hat{\mathbf{b}}_{z_d} \hat{\mathbf{b}}_{z_d}^\top$). Ademais, $\mathbf{P} = [\hat{\mathbf{e}}_x, \hat{\mathbf{e}}_y]$ é a base ortogonal do plano x, y do mundo.

Por fim, o eixo $\hat{\mathbf{b}}_{y_d}$ é finalmente construído como

$$\hat{\mathbf{b}}_{y_d} = \hat{\mathbf{b}}_{z_d} \times \hat{\mathbf{b}}_{x_d}. \quad (79)$$

Essa abordagem garante que a direção η seja preservada durante as manobras desde que $\hat{\mathbf{b}}_{z_d} \nparallel \hat{\mathbf{h}}_d$.

5.3 Sintonia

A sintonia de controle refere-se ao processo de ajustar os parâmetros de um controlador para que ele atinja o desempenho desejado em um sistema. No nosso contexto, as leis de controle propostas incluem termos Proporcional (P), Integral (I) e Derivativo (D) para melhorar a estabilidade, amortecer oscilações, eliminar erros residuais ao longo do tempo, para aumentar a precisão.

Historicamente, vários métodos foram desenvolvidos para sintonizar controladores PID, sendo os mais comuns:

- Método de Ziegler-Nichols: um dos métodos mais tradicionais, que utiliza ensaios experimentais no sistema para determinar os valores dos parâmetros K_p , K_i , e K_d . Esse método pode ser feito por meio da resposta em malha fechada (usando oscilações contínuas) ou pela resposta em malha aberta (pelo método da curva de reação)([ÅSTRÖM; HäGGLUND, 1995](#)).
- Método da tentativa e erro: esse método envolve ajustar os parâmetros manualmente, com base na observação das respostas do sistema, até que o comportamento desejado seja atingido ([ZIEGLER; NICHOLS, 1942](#)).
- Método da resposta em frequência: baseia-se em ajustar os parâmetros do controlador analisando o comportamento do sistema no domínio da frequência.

Embora esses métodos funcionem bem para sistemas simples, lineares e bem modelados, eles apresentam limitações em sistemas mais complexos ([SEBORG et al., 2010](#)). Alguns dos principais desafios incluem:

- Ajuste manual: em muitos casos, os controladores são ajustados manualmente, o que pode ser um processo demorado e sujeito a erros. Além disso, não há garantia de que o ajuste encontrado seja o ótimo.
- Não linearidades: sistemas não lineares, como robôs, motores e processos químicos, respondem de maneira diferente a diferentes condições de operação. Métodos tradicionais, como Ziegler-Nichols, assumem linearidade ou pequenas não linearidades, tornando a sintonia menos eficaz nesses casos.
- Variações nos parâmetros do sistema: muitos sistemas sofrem variações ao longo do tempo, o que pode exigir re-sintonização frequente dos controladores.
- Sistemas multivariáveis: quando há múltiplas entradas e saídas (MIMO), a sintonia de controladores se torna significativamente mais complexa, pois os

parâmetros de um controlador podem influenciar o comportamento de outros subsistemas.

Para superar as limitações dos métodos tradicionais de sintonia, utilizamos o PSO (Particle Swarm Optimization) como uma estratégia de otimização capaz de ajustar automaticamente os ganhos do sistema de controle. A escolha do PSO foi motivada pela necessidade de lidar com um número elevado de ganhos (12 no total) e pela complexidade das não linearidades do sistema.

5.3.1 PSO

A Otimização por Enxame de Partículas (conhecida como Particle Swarm Optimization - PSO, em inglês) é um algoritmo heurístico, um método que busca soluções baseando-se em tentativas e aproximações, sem garantia de encontrar o ótimo global, inspirado no comportamento coletivo de bandos de pássaros. Este método foi desenvolvido por Eberhart e Kennedy em 1995 ([EBERHART; KENNEDY, 1995](#)), com o propósito de encontrar a melhor solução dentro de um espaço de busca. Para isso, os indivíduos de uma população cooperam entre si, definindo a trajetória que cada um deverá seguir dentro do espaço de busca para alcançar o objetivo.

Cada partícula tem uma posição e uma velocidade, e elas se movem pelo espaço de busca, em busca da solução ótima, sendo influenciadas pelas melhores soluções que elas já encontraram individualmente e também pela melhor solução encontrada por todo o grupo (ou enxame). Isso cria um equilíbrio entre exploração global, que busca novas áreas no espaço de busca, e exploração local, que refina as soluções já encontradas.

No PSO, cada partícula possui:

- Posição: representa uma solução candidata para o problema.
- Velocidade: define a direção e a magnitude do movimento da partícula no próximo passo.
- Melhor posição individual: a melhor posição que a partícula encontrou até agora.

- Melhor posição global: a melhor posição encontrada por qualquer partícula até agora.

As partículas se movem de acordo com as seguintes equações:

$$v_i(t+1) = w \cdot v_i(t) + c_1 \cdot r_1 \cdot (p_{\text{best},i} - x_i(t)) + c_2 \cdot r_2 \cdot (g_{\text{best}} - x_i(t)) \quad (80)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (81)$$

onde $v_i(t)$ é a velocidade da partícula i no tempo t , $x_i(t)$ é a posição, w é o fator de inércia, e c_1, c_2 são os fatores de atração para as melhores posições.

Abaixo podemos ver o algoritmo do PSO abaixo:

Algoritmo 1: Pseudocódigo do PSO

Inicializar um conjunto de N partículas com posições e velocidades aleatórias;

for cada partícula i **do**

 Avaliar a função objetivo $f(x_i)$ na posição da partícula;

 Definir $p_{\text{best},i}$ como a posição inicial da partícula;

end

Definir g_{best} como a melhor posição entre todas as partículas;

while critério de parada não for atingido **do**

for cada partícula i **do**

 Atualizar a velocidade:

$$v_i = w \cdot v_i + c_1 \cdot r_1 \cdot (p_{\text{best},i} - x_i) + c_2 \cdot r_2 \cdot (g_{\text{best}} - x_i);$$

 Atualizar a posição: $x_i = x_i + v_i$;

 Avaliar a função objetivo $f(x_i)$;

 Atualizar $p_{\text{best},i}$ se a nova posição for melhor;

 Atualizar g_{best} se a nova posição for a melhor globalmente;

end

end

Retornar g_{best} como a melhor solução encontrada;

Para sintonizar os ganhos desse trabalho, foram utilizados os seguintes parâmetros:

Tabela 6 – Parâmetros utilizados na sintonia dos ganhos

Parâmetro	Valor
Número de partículas (N)	20
Coefficiente de inércia (w)	0.7
Coefficiente cognitivo (c_1)	1.2
Coefficiente social (c_2)	1.3
Número máximo de iterações	50
Critério de parada	Função objetivo $< 1e - 5$

Utilizando os parâmetros listados na Tabela 6, obtemos os seguintes ganhos como resultados através do algoritmo PSO:

Tabela 7 – Valores dos ganhos obtidos pelo PSO

Parâmetro	Valor
kp	50
kv	19
ka	1
kr	0.00001
kw	0.003311
ka	0.00001
krp	0.00001
kwp	0.00544
kap	0.00001
kw_{yaw}	0.387753
kr_{yaw}	1
ka_{yaw}	0.0001

5.4 Resultados

Nesta seção, são apresentados os resultados das simulações realizadas para avaliar o desempenho do controlador geométrico baseado em $SE(3)$, aplicado ao modelo dinâmico do quadricóptero, conforme descrito na Seção 3.2. As simulações foram conduzidas no ambiente MATLAB, levando em consideração a dinâmica não linear do quadricóptero e os efeitos aerodinâmicos relevantes.

O controlador proposto foi validado utilizando a arquitetura do drone Mambo Parrot. Para fins comparativos, foi utilizado o controlador PID padrão do drone

Mambo, que já foi otimizado pela comunidade do MATLAB, e o controlador de postura $SE(3)$, otimizado utilizando o método PSO (Particle Swarm Optimization) para ajuste dos ganhos. Conforme na Figura 20 podemos ver a arquitetura de controle abaixo, onde no bloco amarelo que foi realizado a implementação do controle de postura geométrico $SE(3)$.

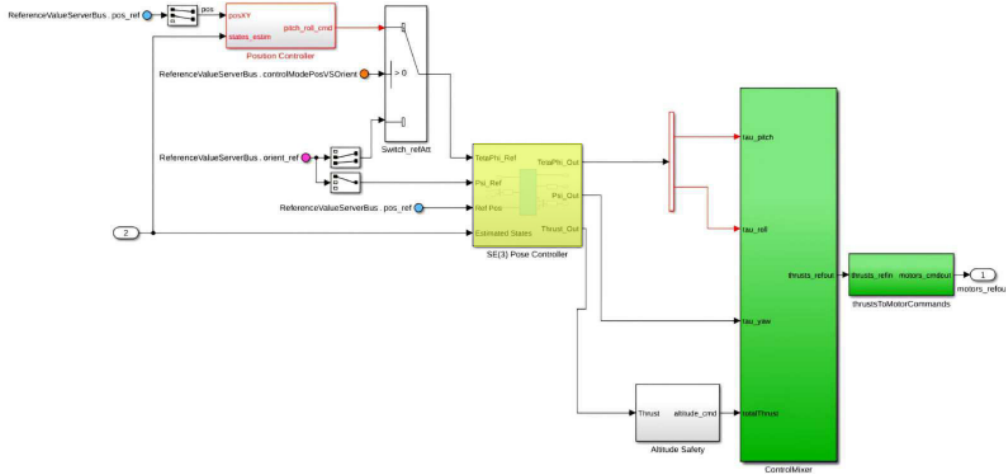


Figura 20 – Diagrama de Controle

Fonte: Próprio Autor

E os parâmetros que foram utilizados no controlador podem ser visualizados na tabela abaixo:

Tabela 8 – Parâmetros do VANT

J	$[0.0000582857, 0.0000716914, 0.0001] \text{ kg} \cdot \text{m}^2$
m	0.063 kg
d	-
Cd	6.9389×10^{-18}

No contexto dos trabalhos de (LEE et al., 2010) e (BACA et al., 2021), os controladores utilizam um único ganho para cada termo do PID (Proporcional, Integral e Derivativo) nas equações de força e torque desejados. Isso significa que, para cada termo (P, I, D), um único valor é aplicado a todos os eixos x , y e Z .

No entanto, em nosso trabalho, optamos por expandir essa abordagem. Na equação ??, não atribuímos um único valor para os ganhos k_R , k_ω e $k_{\omega I}$, como feito em (LEE et al., 2010) e (BACA et al., 2021). Em vez disso, atribuímos ganhos independentes para cada componente do vetor de posição (x, y, z) , totalizando 12 ganhos. Essa decisão foi tomada porque observamos que o uso de um único ganho para cada termo não era suficiente para lidar adequadamente com as não linearidades do sistema, resultando em desempenho insatisfatório.

$$\begin{aligned} k_R &= [kr, krp, kr_{yaw}] \\ k_\omega &= [k\omega, k\omega p, k\omega_{yaw}] \\ k_{\omega I} &= [ka, kap, ka_{yaw}] \end{aligned} \tag{82}$$

Isso possibilita ajustar de forma independente a resposta de cada eixo, levando em conta as diferentes dinâmicas e características de cada direção. Essa abordagem melhora a estabilidade do sistema, evitando instabilidades que podem surgir quando os eixos são tratados de maneira uniforme, além de otimizar o desempenho do controlador, tornando-o mais robusto a perturbações externas e garantindo um controle mais preciso, especialmente em manobras complexas. Ao permitir ajustes específicos, a separação desses ganhos contribui para um controle mais eficiente, minimizando os erros e melhorando a resposta do drone durante o voo.

A análise de desempenho foi realizada em duas trajetórias: quadrada e circular. Os resultados de RMSE (Root Mean Square Error) para cada controlador foram calculados e comparados. A seguir, são apresentados os resultados detalhados.

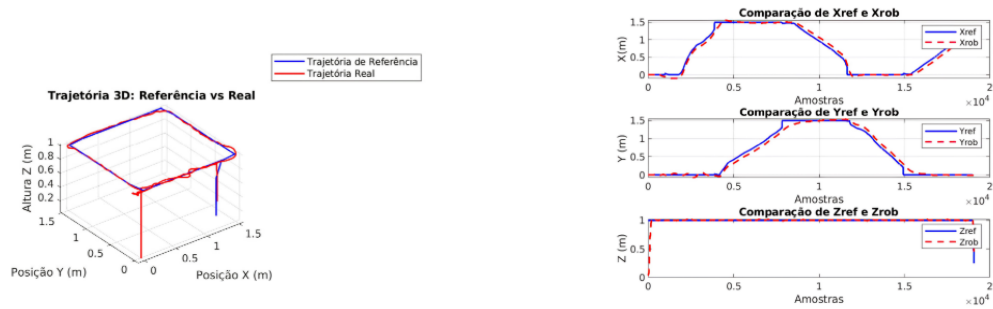
Na Tabela abaixo, indicam que o seguinte resultado: Que o controlador $SE(3)$ teve um desempenho superior em relação ao PID em ambas as trajetórias. A melhoria no RMSE reflete uma navegação mais precisa e eficiente com o controlador $SE(3)$, que contribui para a manutenção da postura do drone durante o voo.

Tabela 9 – Comparação RMSE

	PID	SE(3)
Trajetoória Quadrada	0.2193	0.2178
Trajetoória Circular	0.2087	0.2022

Isso mostra que o controlador $SE(3)$ se destaca ao garantir a postura do drone, facilitando o controle preciso das orientações, especialmente em trajetórias que exigem mudanças frequentes de direção. Essa manutenção da postura reduz o impacto de problemas de singularidade, que podem ocorrer em controladores convencionais, onde o drone pode perder a estabilidade em determinadas orientações.

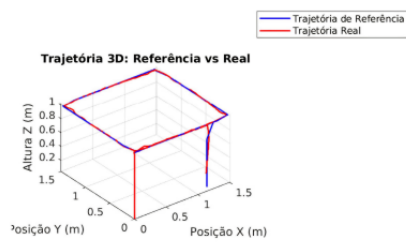
As Figuras 21, 22, 23 e 24 ilustram a comparação entre as simulações com o controlador $SE(3)$ e o controlador PID para as trajetórias quadrada e circular, respectivamente. As imagens incluem gráficos de trajetória 3D e a evolução da posição ao longo do tempo.



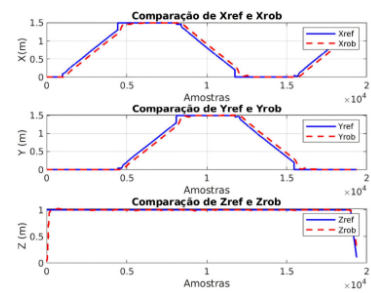
(a) Trajetória 3D (SE(3) - Quadrado)

(b) Posição (SE(3) - Quadrado)

Figura 21 – Trajetória 3D e RMSE da posição em SE(3) para o movimento em quadrado.

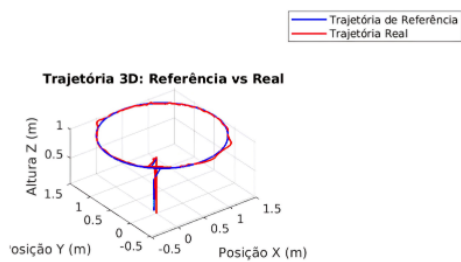


(a) Trajetória 3D (PID) - Quadrado

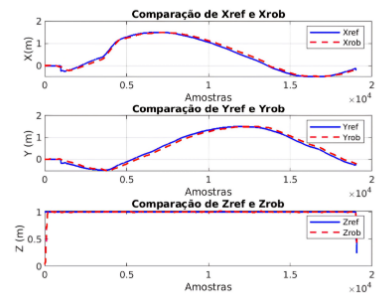


(b) Posição (PID - Quadrado)

Figura 22 – Trajetória 3D e RMSE da posição no PID para o movimento em quadrado.

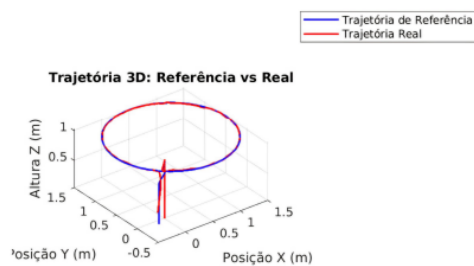


(a) Trajetória 3D (SE3) - Círculo

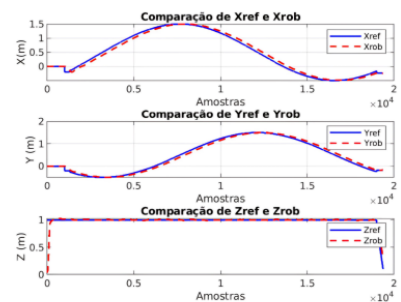


(b) Posição (SE3 - Círculo)

Figura 23 – Trajetória 3D e RMSE da posição no SE3 para o movimento circular.



(a) Trajetória 3D (PID) - Quadrado



(b) Posição (PID - Quadrado)

Figura 24 – Trajetória 3D e RMSE da posição no PID para o movimento circular.

5.5 Considerações Finais

Neste capítulo, apresentamos uma modelagem matemática detalhada do sistema, com foco nas dinâmicas do quadricóptero e no desenvolvimento de um controlador geométrico. Abordamos os fundamentos teóricos do controle geométrico no espaço $SE(3)$ e implementamos uma arquitetura de controle que visa aprimorar a estabilidade e precisão do quadricóptero, especialmente em cenários de voo ágil. O controlador $SE(3)$ otimizado com PSO mostrou-se eficaz para lidar com as singularidades e garantir um rastreamento robusto de postura, superando limitações comuns de controladores convencionais, como o PID.

Os resultados obtidos em simulação demonstraram que o controlador $SE(3)$ atinge um desempenho superior em termos de RMSE (Root Mean Square Error), uma métrica que quantifica a precisão do rastreamento de trajetória. Conforme apresentado na Tabela 9, o controlador $SE(3)$ obteve um RMSE de 0.2178 para a trajetória quadrada e 0.2022 para a trajetória circular, valores ligeiramente inferiores aos do controlador PID, que registrou 0.2193 e 0.2087, respectivamente. Essa melhoria, embora pequena em magnitude, é significativa em termos de estabilidade e precisão, especialmente em manobras complexas e trajetórias tridimensionais.

A superioridade do controlador $SE(3)$ pode ser atribuída à sua capacidade de lidar com as não linearidades do sistema de forma mais eficiente, graças à otimização dos ganhos via PSO e à estrutura geométrica que evita singularidades. Além disso, a separação dos ganhos por eixo (x, y, z) nos permitiu um ajuste mais refinado do comportamento do controlador em cada direção, resultando em um rastreamento de trajetória mais preciso e estável.

Essa abordagem nos permitiu melhorar a precisão do controle em voo, mas também oferece uma estrutura mais flexível para futuros trabalhos, como a incorporação de distúrbios externos (vento, turbulência) ou a extensão para cenários de voo em ambientes mais desafiadores.

6 Conclusão

Neste trabalho, investigamos e propusemos abordagens para o controle e rastreamento de alvos utilizando UAVs autônomos, com foco na implementação de técnicas de controle baseado em visão. Os principais resultados obtidos ao longo do estudo foram estruturados em torno de quatro capítulos-chave: a análise de trabalhos relacionados, a modelagem matemática do sistema, o desenvolvimento de um controlador geométrico de postura e a implementação de um sistema de rastreamento robusto.

O Capítulo 2 forneceu uma base sólida de conhecimento, revisando o estado da arte nas áreas de controle geométrico e sistemas de visão computacional aplicados à robótica. Através de uma análise concisa dos estudos mais recentes, destacamos a eficácia e a relevância dessas abordagens em cenários dinâmicos, como operações de UAVs, enfatizando, contudo, a falta de validação experimental em plataformas reais, o que abre espaço para futuras investigações e melhorias.

No Capítulo 3, apresentamos uma modelagem matemática detalhada do sistema, abordando as dinâmicas do quadrotor e a modelagem do alvo. Esta seção foi fundamental para estabelecer uma base teórica sólida para o desenvolvimento de controladores eficazes, considerando tanto os aspectos cinemáticos quanto aerodinâmicos que influenciam o comportamento do quadrotor. A modelagem do alvo também foi discutida, com foco na previsão de seu movimento e na estimativa de sua posição, crucial para o rastreamento preciso do alvo.

No Capítulo 4, desenvolvemos um sistema de rastreamento robusto de objetos voadores com base em visão, utilizando um UAV seguidor e um UAV alvo. O sistema foi capaz de detectar e rastrear drones em tempo real, com alta precisão e baixa latência, comprovando sua aplicabilidade prática para operações de vigilância e monitoramento. Os resultados experimentais validaram a eficácia da metodologia proposta, demonstrando seu potencial para aplicações em cenários reais de interceptação aérea.

Finalmente, o Capítulo 5 concentrou-se no desenvolvimento de um controlador geométrico para o quadrotor, explorando o controle no espaço $SE(3)$.

Através da otimização do controlador $SE(3)$ utilizando Particle Swarm Optimization (PSO), conseguimos superar limitações típicas de controladores convencionais, como o PID, alcançando um desempenho superior ao PID conforme podemos visualizar na Tabela 9. Os resultados das simulações demonstraram que o controlador $SE(3)$ foi eficaz em termos de precisão e estabilidade, com um desempenho significativamente superior em termos de RMSE, garantindo um rastreamento robusto de postura, mesmo em manobras complexas.

Em suma, este trabalho contribui significativamente para o avanço das técnicas de controle baseado em visão aplicadas a UAVs autônomos, apresentando uma abordagem eficaz para o rastreamento de alvos móveis e a implementação de controladores robustos e eficientes. O desenvolvimento e validação dos métodos propostos oferecem uma base sólida para futuras pesquisas, com destaque para a necessidade de mais experimentos em plataformas reais, visando melhorar ainda mais a robustez e a aplicabilidade dos sistemas desenvolvidos.

6.0.1 Trabalhos Futuros

Embora os resultados obtidos neste trabalho tenham demonstrado a eficácia dos métodos propostos para o controle e rastreamento de alvos com UAVs autônomos, existem várias áreas que podem ser exploradas em futuras investigações.

Uma direção importante para os trabalhos futuros é a validação em plataformas reais. Embora as simulações tenham mostrado um desempenho satisfatório, a implementação prática em UAVs reais é crucial para avaliar o desempenho dos controladores em condições de operação mais desafiadoras, como mudanças atmosféricas e interferências externas. Além disso, a integração com outros tipos de sensores, como LiDAR e radar, pode ser investigada para melhorar a robustez do sistema, especialmente em ambientes com visibilidade reduzida.

Um outro ponto complementar é o aperfeiçoamento do controlador geométrico baseado em visão utilizando leis adaptativas, explorando-as para lidar com não-linearidades e incertezas do sistema de maneira mais eficiente. A adaptação do controlador para diferentes configurações de UAV e para uma variedade maior de missões (como missões de busca e resgate ou patrulhamento) também seria uma contribuição relevante.

Essas direções sugerem várias áreas para futuras investigações, com o objetivo de melhorar ainda mais a performance e a aplicabilidade dos sistemas desenvolvidos, contribuindo para o avanço das tecnologias de UAVs autônomos em diversas aplicações.

Referências

EBERHART, R. and Kennedy, J.

ABBASI, Y.; MOOSAVIAN, S.; NOVINZADEH, A. Vision-based formation control of aerial robots in the presence of sensor failure. **Journal of Mechanical Science and Technology**, v. 31, p. 1413–1426, 11 2016.

ALGABRI, M.; MATHKOUR, H.; MEKHTICHE, M. A.; BENCHERIF, M. A.; ALSULAIMAN, M.; ARAFAH, M. A.; GHALEB, H. Wireless vision-based fuzzy controllers for moving object tracking using a quadcopter. **International Journal of Distributed Sensor Networks**, v. 13, n. 4, p. 1550147717705549, 2017. Disponível em: <<https://doi.org/10.1177/1550147717705549>>.

ALSHAHIR, A.; ALBEKAIRI, M.; BERRIRI, K.; MEKKI, H.; KAANICHE, K.; ALSHAHR, S.; ALSHAMMARI, B. A.; SAHBANI, A. Quadrotor uav dynamic visual servoing based on differential flatness theory. **Applied Sciences**, v. 13, n. 12, 2023. ISSN 2076-3417. Disponível em: <<https://www.mdpi.com/2076-3417/13/12/7005>>.

ANDERSON, R. T.; CHOWDHARY, G.; JOHNSON, E. N. Comparison of rbf and shl neural network based adaptive control. **J. Intell. Robotics Syst.**, Kluwer Academic Publishers, USA, v. 54, n. 1–3, p. 183–199, mar 2009. ISSN 0921-0296. Disponível em: <<https://doi.org/10.1007/s10846-008-9262-1>>.

ANDERSON, S. B. Technical Memorandum, **Historical overview of V/STOL aircraft technology**. 1981. Available at ntrs.nasa.gov.

ARAAR, O.; AOUF, N. Practical servoing of vtol uavs from object of unknown 3d geometry: new hybrid approach. **The Imaging Science Journal**, v. 63, p. 1743131X15Y.000, 07 2015.

ARAAR, O.; AOUF, N.; VITANOV, I. Vision based autonomous landing of multirotor uav on moving platform. **Journal of Intelligent & Robotic Systems**, v. 85, 02 2017.

ARIF, A.; WANG, H.; LIU, Z.; CASTAÑEDA, H.; WANG, Y. Adaptive visual servo control law for finite-time tracking to land quadrotor on moving platform using virtual reticle algorithm. **Robotics and Autonomous Systems**, v. 141, p. 103764, 2021. ISSN 0921-8890. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S092188902100049X>>.

ASL, H. J.; BOLANDI, H. Robust vision-based control of an underactuated flying robot tracking a moving target. **Transactions of the Institute of Measurement and Control**, v. 36, n. 3, p. 411–424, 2014. Disponível em: <<https://doi.org/10.1177/0142331213502747>>.

ASL, H. J.; YOON, J. Robust image-based control of the quadrotor unmanned aerial vehicle. **Nonlinear Dynamics**, v. 85, 08 2016.

BACA, T.; PETRLIK, M.; VRBA, M.; SPURNY, V.; PENICKA, R.; HERT, D.; SASKA, M. The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-world Deployment, and Education with Autonomous Unmanned Aerial Vehicles. **Journal of Intelligent & Robotic Systems**, Springer, v. 102, n. 26, p. 1–28, May 2021. Disponível em: <<https://link.springer.com/article/10.1007/s10846-021-01383-5>>.

BACHRACH, A.; HE, R.; ROY, N. Autonomous flight in unknown indoor environments. **International Journal of Micro Air Vehicles**, v. 1, n. 4, p. 217–228, 2009. Disponível em: <<https://doi.org/10.1260/175682909790291492>>.

BALDIVIESO, T.; BOENTE, A.; BATISTA, T.; ROSA, P. Aplicação de mini vants em modelos 3d em escala real para preservação cultural de construções históricas. In: **Anais do III Workshop Brasileiro de Cidades Inteligentes**. Porto Alegre, RS, Brasil: SBC, 2022. p. 168–175. ISSN 0000-0000. Disponível em: <<https://sol.sbc.org.br/index.php/wbci/article/view/20455>>.

BARROS, C.; FREITAS, E. D.; BRAGA, A. R.; BOMFIM, I. G.; GOMES, D. Aplicando redes neurais convolucionais em imagens para reconhecimento automatizado de abelhas melíferas (*apis mellifera* l.). In: **Anais do XII Workshop de Computação Aplicada à Gestão do Meio Ambiente e Recursos Naturais**. Porto Alegre, RS, Brasil: SBC, 2021. p. 19–28. ISSN 2595-6124. Disponível em: <<https://sol.sbc.org.br/index.php/wcama/article/view/15733>>.

BHARGAVAPURI, M.; SHASTRY, A. K.; SINHA, H.; SAHOO, S. R.; KOTHARI, M. Vision-based autonomous tracking and landing of a fully-actuated rotorcraft. **Control Engineering Practice**, v. 89, p. 113–129, 2019. ISSN 0967-0661. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0967066118306415>>.

BIN, X.; SEN, Y.; XU, Z. Control of a quadrotor helicopter using the compass (beidou) system and on-board vision system. **Optik**, v. 127, n. 17, p. 6829–6838, 2016. ISSN 0030-4026. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0030402616304557>>.

BISHEBAN, M.; LEE, T. Geometric adaptive control with neural networks for a quadrotor in wind fields. **IEEE Transactions on Control Systems Technology**, v. 29, n. 4, p. 1533–1548, 2021.

BOUABDALLAH, S.; MURRIERI, P.; SIEGWART, R. Design and control of an indoor micro quadrotor. In: **IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004.** [S.l.: s.n.], 2004. v. 5, p. 4393–4398 Vol.5.

BOUAISS, O.; MECHGOUG, R.; AHMED, A. taleb. Visual soft landing of an autonomous quadrotor on a moving pad using a combined fuzzy velocity control with model predictive control. **Signal, Image and Video Processing**, v. 17, 04 2022.

BRAS, F. L.; HAMEL, T.; MAHONY, R.; TREIL, A. Output feedback observation and control for visual servoing of vtol uavs. **International Journal of Robust and Nonlinear Control**, v. 21, n. 9, p. 1008–1030, 2011. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.1638>>.

BRAS, F. L.; MAHONY, R.; HAMEL, T. Output based observation and control for visual servoing of vtol uav's. **IFAC Proceedings Volumes**, v. 41, n. 2, p. 8558–8563, 2008. ISSN 1474-6670. 17th IFAC World Congress. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1474667016403265>>.

BRAS, F. L.; MAHONY, R.; HAMEL, T.; BINETTI, P. Dynamic image-based visual servo control for an aerial robot: Theory and experiments. **International Journal of Optomechatronics**, v. 2, p. 296–325, 07 2008.

CAI, J.; XIAN, B. Robust hierarchical geometry control for the multiple uavs aerial transportation system with a suspended payload. **Nonlinear Dynamics**, v. 112, n. 6, p. 4551–4571, 2024. ISSN 1573-269X. Disponível em: <<https://doi.org/10.1007/s11071-024-09277-5>>.

CALDERÓN, M.; AGUILAR, W. G.; MERIZALDE, D. Visual-based real-time detection using neural networks and micro-uavs for military operations. In: ROCHA, Á.; PAREDES-CALDERÓN, M.; GUARDA, T. (Ed.). **Developments and Advances in Defense and Security**. Singapore: Springer Nature Singapore, 2020. p. 55–64. ISBN 978-981-15-4875-8.

CALISE, A.; RYSDYK, R. Nonlinear adaptive flight control using neural networks. **IEEE Control Systems Magazine**, v. 18, n. 6, p. 14–25, 1998.

CASTANON, D. A.; WOHPLETZ, J. M. Model predictive control for stochastic resource allocation. **IEEE Transactions on Automatic Control**, v. 54, n. 8, p. 1739–1750, 2009.

CEREN, Z.; ALTUG, E. Image based and hybrid visual servo control of an unmanned aerial vehicle. **Journal of Intelligent and Robotic Systems**, v. 65, p. 325–344, 09 2012.

CESETTI, A.; FRONTONI, E.; MANCINI, A.; ZINGARETTI, P.; LONGHI, S. A vision-based guidance system for uav navigation and safe landing using natural landmarks. **Journal of Intelligent and Robotic Systems**, v. 57, p. 233–257, 01 2010.

CHAUMETTE, F.; HUTCHINSON, S. Visual servo control, part i: Basic approaches. **IEEE Robotics and Automation Magazine**, IEEE, v. 13, n. 4, p. 82–90, 2006.

CHEN, C.; CHEN, S.; HU, G.; CHEN, B.; CHEN, P.; SU, K. An auto-landing strategy based on pan-tilt based visual servoing for unmanned aerial vehicle in gnss-denied environments. **Aerospace Science and Technology**, v. 116, p. 106891, 2021. ISSN 1270-9638. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1270963821004016>>.

CHEN, C.-W.; HUNG, H.-A.; YANG, P.-H.; CHENG, T.-H. Visual servoing of a moving target by an unmanned aerial vehicle. **Sensors**, v. 21, n. 17, 2021. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/21/17/5708>>.

CHLEBOUN, J.; AMORIM, T.; NASCIMENTO, A. M.; NASCIMENTO, T. P. An improved spanning tree-based algorithm for coverage of large areas using multi-uav systems. **Drones**, v. 7, n. 1, 2023. ISSN 2504-446X. Disponível em: <<https://www.mdpi.com/2504-446X/7/1/9>>.

CHOWDHARY, G.; JOHNSON, E.; MAGREE, D.; WU, A.; SHEIN, A. Gps-denied indoor and outdoor monocular vision aided navigation and control of unmanned aircraft. **Journal of Field Robotics**, v. 30, 05 2013.

CHUANG, H.-M.; HE, D.; NAMIKI, A. Autonomous target tracking of uav using high-speed visual feedback. **Applied Sciences**, v. 9, n. 21, 2019. ISSN 2076-3417. Disponível em: <<https://www.mdpi.com/2076-3417/9/21/4552>>.

CORKE, P. **Robotics, Vision and Control: Fundamental Algorithms in MATLAB® Second, Completely Revised**. [S.l.]: Springer, 2017.

CROON, G.; WAGTER, C. D.; SEIDL, T. Enhancing optical-flow-based control by learning visual appearance cues for flying robots. **Nature Machine Intelligence**, v. 3, p. 33–41, 01 2021.

DELBENE, A.; BAGLIETTO, M.; SIMETTI, E. Visual servoed autonomous landing of an uav on a catamaran in a marine environment. **Sensors**, v. 22, n. 9, 2022. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/22/9/3544>>.

DIEBEL, J. Representing attitude: Euler angles, unit quaternions, and rotation vectors. **Matrix**, v. 58, n. 15-16, p. 1–35, 2006.

DUARTE, M.; NARENDRA, K. Combined direct and indirect approach to adaptive control. **IEEE Transactions on Automatic Control**, v. 34, n. 10, p. 1071–1075, 1989.

DUONG, T.; ATANASOV, N. Adaptive control of $se(3)$ hamiltonian dynamics with learned disturbance features. **IEEE Control Systems Letters**, v. 6, p. 2773–2778, 2022.

DYDEK, Z. T.; ANNASWAMY, A. M.; LAVRETSKY, E. Adaptive control and the nasa x-15-3 flight revisited. **IEEE Control Systems Magazine**, v. 30, n. 3, p. 32–48, 2010.

EBERHART, R.; KENNEDY, J. A new optimizer using particle swarm theory. In: **MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science**. [S.l.: s.n.], 1995. p. 39–43.

FERNANDO, T.; CHANDIRAMANI, J.; LEE, T.; GUTIERREZ, H. Robust adaptive geometric tracking controls on $so(3)$ with an application to the attitude dynamics of a quadrotor uav. In: **2011 50th IEEE Conference on Decision and Control and European Control Conference**. [S.l.: s.n.], 2011. p. 7380–7385.

FRANCHI, A.; MASONE, C.; GRABE, V.; RYLL, M.; BÜLTHOFF, H. H.; GIORDANO, P. R. Modeling and control of uav bearing formations with bilateral high-level steering. **The International Journal of Robotics Research**, v. 31, n. 12, p. 1504–1525, 2012. Disponível em: <<https://doi.org/10.1177/0278364912462493>>.

GHASEMI, A.; PARIVASH, F.; EBRAHIMIAN, S. Autonomous landing of a quadrotor on a moving platform using vision-based fofpid control. **Robotica**, v. 40, n. 5, p. 1431–1449, 2022.

GHOMMAM, J.; FETHALLA, N.; SAAD, M. Quadrotor circumnavigation of an unknown moving target using camera vision-based measurements. **IET Control Theory & Applications**, v. 10, n. 15, p. 1874–1887, 2016. Disponível em: <<https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-cta.2015.1246>>.

GIBSON, T. E.; CRESPO, L. G.; ANNASWAMY, A. M. Adaptive control of hypersonic vehicles in the presence of modeling uncertainties. In: **2009 American Control Conference**. [S.l.: s.n.], 2009. p. 3178–3183.

GOMEZ-BALDERAS, J.; FLORES, G.; CARRILLO, L. R. G.; LOZANO, R. Tracking a ground moving target with a quadrotor using switching control. **Journal of Intelligent & Robotic Systems**, v. 70, 04 2013.

GOMEZ-BALDERAS, J.-E.; GARCIA, P. C.; GUERRERO, J.; LOZANO, R. Vision based tracking for a quadrotor using vanishing points. **Journal of Intelligent and Robotic Systems**, v. 65, p. 361–371, 07 2012.

GONZALEZ-GARCIA, A.; MIRANDA-MOYA, A.; CASTAÑEDA, H. Robust visual tracking control based on adaptive sliding mode strategy: Quadrotor uav - catamaran usv heterogeneous system. In: **2021 International Conference on Unmanned Aircraft Systems (ICUAS)**. [S.l.: s.n.], 2021. p. 666–672.

GOODARZI, F. A.; LEE, D.; LEE, T. Geometric adaptive tracking control of a quadrotor unmanned aerial vehicle on $se(3)$ for agile maneuvers. **Journal of Dynamic Systems, Measurement, and Control**, ASME International, v. 137, n. 9, jun. 2015. ISSN 1528-9028. Disponível em: <<http://dx.doi.org/10.1115/1.4030419>>.

GROVES, K.; SIGTHORSSON, D.; SERRANI, A.; YURKOVICH, S.; BOLENDER, M.; DOMAN, D. Reference command tracking for a linearized model of an air-breathing hypersonic vehicle. In: **AIAA Guidance, Navigation, and Control Conference and Exhibit**. [s.n.]. Disponível em: <<https://arc.aiaa.org/doi/abs/10.2514/6.2005-6144>>.

GUENARD, N.; HAMEL, T.; MAHONY, R. A practical visual servo control for an unmanned aerial vehicle. **IEEE Transactions on Robotics**, v. 24, n. 2, p. 331–340, 2008.

GUENARD, N.; HAMEL, T.; MAHONY, R. A practical visual servo control for an unmanned aerial vehicle. **IEEE Transactions on Robotics**, v. 24, n. 2, p. 331–340, 2008.

GUPTA, P. M.; PAIRET, E.; NASCIMENTO, T.; SASKA, M. Landing a uav in harsh winds and turbulent open waters. **IEEE Robotics and Automation Letters**, v. 8, n. 2, p. 744–751, 2023.

HAM, Y.; HAN, K.; LIN, J.; GOLPARVAR-FARD, M. Visual monitoring of civil infrastructure systems via camera-equipped unmanned aerial vehicles (uavs): a review of related works. **Visualization in Engineering**, v. 4, 12 2016.

HASHIM, H. A.; ELTOUKHY, A. E.; ODRY, A. Observer-based controller for vtol-uavs tracking using direct vision-aided inertial navigation measurements. **ISA Transactions**, v. 137, p. 133–143, 2023. ISSN 0019-0578. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0019057822006450>>.

HE, W.; YUAN, L. Fixed-time controller of visual quadrotor for tracking a moving target. **Journal of Vibration and Control**, v. 0, n. 0, p. 10775463231200914, 0. Disponível em: <<https://doi.org/10.1177/10775463231200914>>.

Hert, D.; Baca, T.; Petracek, P.; Kratky, V.; Spurny, V.; Petrlik, M.; Vrba, M.; Zaitlik, D.; Stoudek, P.; Walter, V.; Stepan, P.; Horyna, J.; Pritzl, V.; Silano, G.; Bonilla Licea, D.; Stibinger, P.; Penicka, R.; Nascimento, T.; Saska, M. MRS Modular UAV Hardware Platforms for Supporting Research in Real-World Outdoor and Indoor Environments. In: IEEE. **2022 International Conference on Unmanned Aircraft Systems (ICUAS)**. [S.l.], 2022. p. 1264–1273.

HOFFMANN, G.; HUANG, H.; WASLANDER, S.; TOMLIN, C. Quadrotor helicopter flight dynamics and control: Theory and experiment. 08 2007.

HUGO, R.; SERGIO, S.; ROGELIO, L.; RYAD, B. Real-time visual servoing control of a four-rotor rotorcraft. **IFAC Proceedings Volumes**, v. 40, n. 13, p. 209–214, 2007. ISSN 1474-6670. 9th IFAC Workshop on Adaptation and Learning in Control and Signal Processing. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1474667015320188>>.

HUTCHINSON, S.; HAGER, G. D.; CORKE, P. I. A tutorial on visual servo control. **IEEE transactions on robotics and automation**, IEEE, v. 12, n. 5, p. 651–670, 1996.

INVERNIZZI, D.; PANZA, S.; LOVERA, M. Robust tuning of geometric attitude controllers for multirotor unmanned aerial vehicles. **Journal of Guidance, Control, and Dynamics**, v. 43, n. 7, p. 1332–1343, 2020.

IOANNOU, P.; TSAKALIS, K. A robust direct adaptive controller. **IEEE Transactions on Automatic Control**, v. 31, n. 11, p. 1033–1043, 1986.

IOANNOU, P. A.; TSAKALIS, K. A robust direct adaptive controller. **IEEE Transactions on Automatic Control**, IEEE, v. 31, n. 11, p. 1033–1043, Nov 1986.

JENKINS, D. R. **Hypersonics before the shuttle: A concise history of the X-15 research airplane**. NASA, 2000. v. 18. (Monographs in Aerospace History, v. 18). Disponível em: <<https://ntrs.nasa.gov>>.

JIANG, T.; LIN, D.; SONG, T. Vision-based autonomous landing of a quadrotor using a gimbaled camera. **Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering**, v. 233, n. 14, p. 5093–5106, 2019. Disponível em: <<https://doi.org/10.1177/0954410019837777>>.

JURADO, F.; PALACIOS, G.; FLORES, F.; BECERRA, H. M. Vision-based trajectory tracking system for an emulated quadrotor uav. **Asian Journal of Control**, v. 16, n. 3, p. 729–741, 2014. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/asjc.886>>.

KANELLAKIS, C.; KARVELIS, P.; MANSOURI, S. S.; AGHA-MOHAMMADI, A. akbar; NIKOLAKOPOULOS, G. Vision-driven nmpe for autonomous aerial navigation in subterranean environments**this work has been funded by the european unions horizon 2020 research and innovation programme under the grant agreement no. 730302 sims. **IFAC-PapersOnLine**, v. 53, n. 2, p. 9288–9294, 2020. ISSN 2405-8963. 21st IFAC World Congress. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2405896320330536>>.

KARACA, Y.; CICEK, M.; TATLI, O.; SAHIN, A.; PASLI, S.; BESER, M. F.; TUREDI, S. The potential use of unmanned aircraft systems (drones) in mountain search and rescue operations. **The American Journal of Emergency Medicine**, v. 36, n. 4, p. 583–588, 2018. ISSN 0735-6757. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0735675717307507>>.

KARASON, S.; ANNASWAMY, A. Adaptive control in the presence of input constraints. **IEEE Transactions on Automatic Control**, v. 39, n. 11, p. 2325–2330, 1994.

KEIPOUR, A.; PEREIRA, G. A. S.; BONATTI, R.; GARG, R.; RASTOGI, P.; DUBEY, G.; SCHERER, S. Visual servoing approach to autonomous uav landing on a moving vehicle. **Sensors**, v. 22, n. 17, 2022. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/22/17/6549>>.

KENDOUL, F.; NONAMI, K.; FANTONI, I.; LOZANO, R. An adaptive vision-based autopilot for mini flying machines guidance, navigation and control. **Autonomous Robots**, v. 27, 10 2009.

KIM, B. S.; CALISE, A. J. Nonlinear flight control using neural networks. **Journal of Guidance, Control, and Dynamics**, v. 20, n. 1, p. 26–33, 1997. Disponível em: <<https://doi.org/10.2514/2.4029>>.

KIM, H.; HONG, A.; PARK, J. Vision-based line tracking for an indoor micro quadrotor. **Journal of International Council on Electrical Engineering**, v. 3, 07 2013.

KOTARU, P.; EDMONSON, R.; SREENATH, K. **Geometric L1 Adaptive Attitude Control for a Quadrotor Unmanned Aerial Vehicle**. 2020. Disponível em: <<https://arxiv.org/abs/1910.07730>>.

LAVRETSKY, E. Combined/composite model reference adaptive control. In: . [S.l.: s.n.], 2009. v. 54, n. 11, p. 2692–2697.

LEE, H. Trajectory tracking control of multirotors from modelling to experiments: A survey. **International Journal of Control, Automation and Systems**, v. 15, 12 2016.

LEE, T. Geometric adaptive control for aerial transportation of a rigid body. **arXiv: Optimization and Control**, 2015. Disponível em: <<https://api.semanticscholar.org/CorpusID:119014435>>.

LEE, T.; LEOK, M.; MCCLAMROCH, N. H. **Control of Complex Maneuvers for a Quadrotor UAV using Geometric Methods on SE(3)**. 2011. Disponível em: <<https://arxiv.org/abs/1003.2005>>.

LEE, T. et al. Geometric tracking control of a quadrotor UAV on SE(3). In: IEEE. **2010 IEEE Conference on Decision and Control**. [S.l.], 2010. p. 5420–5425.

LEISHMAN, J. G. **Principles of Helicopter Aerodynamics**. Cambridge, United Kingdom: Cambridge University Press, 2006.

LI, J.; XIE, H.; LOW, K. H.; YONG, J.; LI, B. Image-based visual servoing of rotorcrafts to planar visual targets of arbitrary orientation. **IEEE Robotics and Automation Letters**, v. 6, n. 4, p. 7861–7868, 2021.

LI, J.; YE, D. H.; CHUNG, T.; KOLSCH, M.; WACHS, J.; BOUMAN, C. Multi-target detection and tracking from a single camera in unmanned aerial vehicles (uavs). In: **2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**. [S.l.: s.n.], 2016. p. 4992–4997.

LI, S.; OZO, M. M.; De Wagter, C.; de Croon, G. C. Autonomous drone race: A computationally efficient vision-based navigation and control strategy. **Robotics and Autonomous Systems**, v. 133, p. 103621, 2020. ISSN 0921-8890. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0921889020304619>>.

LIN, F.; LUM, K.-Y.; CHEN, B.; LEE, T. Development of a vision-based ground target detection and tracking system for a small unmanned helicopter. **Science in China Series F: Information Sciences**, v. 52, p. 2201–2215, 11 2009.

LIN, J.; MIAO, Z.; ZHONG, H.; PENG, W.; WANG, Y.; FIERRO, R. Adaptive image-based leader–follower formation control of mobile robots with visibility constraints. **IEEE Transactions on Industrial Electronics**, v. 68, n. 7, p. 6010–6019, 2021.

LIN, J.; WANG, Y.; MIAO, Z.; ZHONG, H.; FIERRO, R. Low-complexity control for vision-based landing of quadrotor uav on unknown moving platform. **IEEE Transactions on Industrial Informatics**, v. 18, n. 8, p. 5348–5358, 2022.

LIN, J.; WANG, Y.; MIAO, Z.; FAN, S.; WANG, H. Robust observer-based visual servo control for quadrotors tracking unknown moving targets. **IEEE/ASME Transactions on Mechatronics**, v. 28, n. 3, p. 1268–1279, 2023.

LIN, S.; GARRATT, M.; LAMBERT, A. Monocular vision-based real-time target recognition and tracking for autonomously landing an uav in a cluttered shipboard environment. **Autonomous Robots**, v. 41, 04 2017.

LIU, M.; WANG, X.; ZHOU, A.; FU, X.; MA, Y.; PIAO, C. Uav-yolo: Small object detection on unmanned aerial vehicle perspective. **Sensors**, v. 20, n. 8, 2020. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/20/8/2238>>.

LIU, N.; SHAO, X.; YANG, W. Integral barrier lyapunov function based saturated dynamic surface control for vision-based quadrotors via back-stepping. **IEEE Access**, v. 6, p. 63292–63304, 2018.

LIU, X.; LI, H.; XUE, J.; ZENG, T.; ZHAO, X. Location and tracking of environmental pollution sources under multi-uav vision based on target motion model. **Soft Computing**, v. 27, p. 1–15, 03 2023.

LIU, Y.; MENG, Z.; ZOU, Y.; CAO, M. Visual object tracking and servoing control of a nano-scale quadrotor: System, algorithms, and experiments. **IEEE/CAA Journal of Automatica Sinica**, v. 8, n. 2, p. 344–360, 2021.

LOPEZ-FRANCO, C.; GOMEZ-AVILA, J.; ALANIS, A. Y.; ARANA-DANIEL, N.; VILLASEÑOR, C. Visual servoing for an autonomous hexarotor using a neural network based pid controller. **Sensors**, v. 17, n. 8, 2017. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/17/8/1865>>.

LYU, Y.; CAO, M.; YUAN, S.; XIE, L. Vision-based plane estimation and following for building inspection with autonomous uav. **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, v. 53, n. 12, p. 7475–7488, 2023.

MADRUGA, S. P.; TAVARES, A. H. B. M.; LUIZ, S. O. D.; NASCIMENTO, T. P. do; LIMA, A. M. N. Aerodynamic effects compensation on multi-rotor uavs

based on a neural network control allocation approach. **IEEE/CAA Journal of Automatica Sinica**, v. 9, n. 2, p. 295–312, 2022.

MARANTOS, P.; KARRAS, G.; VLANTIS, P.; KYRIAKOPOULOS, K. Vision-based autonomous landing control for unmanned helicopters. **Journal of Intelligent & Robotic Systems**, v. 92, p. 1–14, 09 2018.

MARTINEZ, C.; MONDRAGÓN, I.; CAMPOY, P.; SANCHEZ-LOPEZ, J. L.; OLIVARES-MENDEZ, M. A hierarchical tracking strategy for vision-based applications on-board uavs. **Journal of Intelligent and Robotic Systems**, v. 72, 03 2013.

MCKERROW, P. Modelling the draganflyer four-rotor helicopter. In: **IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004.** [S.l.: s.n.], 2004. v. 4, p. 3596–3601 Vol.4.

MEBARKI, R.; LIPPIELLO, V.; SICILIANO, B. Nonlinear visual control of unmanned aerial vehicles in gps-denied environments. **IEEE Transactions on Robotics**, v. 31, n. 4, p. 1004–1017, 2015.

MEHTA, S.; TON, C.; RYSZ, M.; KAN, Z.; DOUCETTE, E.; CURTIS, J. New approach to visual servo control using terminal constraints. **Journal of the Franklin Institute**, v. 356, n. 10, p. 5001–5026, 2019. ISSN 0016-0032. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0016003219303072>>.

MEJIAS, C.; MONDRAGÓN, I.; OLIVARES-MENDEZ, M.; CAMPOY, P.; MEJIAS, L.; MONDRAGÓN, I.; OLIVARES, M.; MARTINEZ, C. Unmanned aerial vehicles uavs attitude, height, motion estimation and control using visual systems. **Auton. Robots**, v. 29, 07 2010.

MEJIAS, L.; CAMPOY, P.; SARIPALLI, S.; SUKHATME, G. A visual servoing approach for tracking features in urban areas using an autonomous helicopter. In: . [S.l.: s.n.], 2006. v. 2006, p. 2503 – 2508.

MEJIAS, L.; CAMPOY, P.; SARIPALLI, S.; SUKHATME, G. A visual servoing approach for tracking features in urban areas using an autonomous helicopter. In: **Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.** [S.l.: s.n.], 2006. p. 2503–2508.

METNI, N.; HAMEL, T. A uav for bridge inspection: Visual servoing control law with orientation limits. **Automation in Construction**, v. 17, n. 1, p. 3–10, 2007. ISSN 0926-5805. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0926580507000052>>.

METNI, N.; HAMEL, T.; DERKX, F. A uav for bridge's inspection: visual servoing control law with orientation limits. **IFAC Proceedings Volumes**, v. 37, n. 8, p. 454–459, 2004. ISSN 1474-6670. IFAC/EURON Symposium on Intelligent Autonomous Vehicles, Lisbon, Portugal, 5-7 July 2004. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1474667017320189>>.

MICHAEL, N.; FINK, J.; KUMAR, V. Cooperative manipulation and transportation with aerial robots. In: . [S.l.: s.n.], 2009.

MIRANDA-MOYA, A.; CASTAÑEDA, H.; GORDILLO, J.; WANG, H. Ibvs based on adaptive sliding mode control for a quadrotor target tracking under perturbations. **Mechatronics**, v. 88, p. 102909, 2022. ISSN 0957-4158. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0957415822001271>>.

MISHRA, S.; SYED, D. F.; PLOUGHE, M.; ZHANG, W. Autonomous vision-guided object collection from water surfaces with a customized multirotor. **IEEE/ASME Transactions on Mechatronics**, v. 26, n. 4, p. 1914–1922, 2021.

MOHAMMADI, A.; FENG, Y.; ZHANG, C.; RAWASHDEH, S.; BAEK, S. Vision-based autonomous landing using an mpc-controlled micro uav on a moving platform. In: **2020 International Conference on Unmanned Aircraft Systems (ICUAS)**. [S.l.: s.n.], 2020. p. 771–780.

MURATET, L.; DONCIEUX, S.; BRIERE, Y.; MEYER, J.-A. A contribution to vision-based autonomous helicopter flight in urban environments. **Robotics and Autonomous Systems**, v. 50, n. 4, p. 195–209, 2005. ISSN 0921-8890. Biomimetic Robotics. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0921889004001745>>.

MáTHÉ, K.; BUŞONIU, L. Vision and control for uavs: A survey of general methods and of inexpensive platforms for infrastructure inspection. **Sensors**, v. 15, n. 7, p. 14887–14916, 2015. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/15/7/14887>>.

NABAVI, Y.; ASADI, D.; AHMADI, K. Automatic landing control of a multi-rotor uav using a monocular camera. **Journal of Intelligent & Robotic Systems**, v. 105, 07 2022.

NGUYEN, T.; MANN, G.; GOSINE, R.; VARDY, A. Appearance-based visual-teach-and-repeat navigation technique for micro aerial vehicle. **Journal of Intelligent & Robotic Systems**, v. 84, 12 2016.

NIU, G.; YANG, Q.; GAO, Y.; PUN, M.-O. Vision-based autonomous landing for unmanned aerial and ground vehicles cooperative systems. **IEEE Robotics and Automation Letters**, v. 7, n. 3, p. 6234–6241, 2022.

OGREN, P.; FIORELLI, E.; LEONARD, N. Cooperative control of mobile sensor networks: adaptive gradient climbing in a distributed environment. **IEEE Transactions on Automatic Control**, v. 49, n. 8, p. 1292–1302, 2004.

OLIVARES-MENDEZ, M. A.; FU, C.; LUDIVIG, P.; BISSYANDÉ, T. F.; KANNAN, S.; ZURAD, M.; ANNAIYAN, A.; VOOS, H.; CAMPOY, P. Towards an autonomous vision-based unmanned aerial system against wildlife poachers. **Sensors**, v. 15, n. 12, p. 31362–31391, 2015. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/15/12/29861>>.

OMARI, S.; HUA, M.-D.; DUCARD, G.; HAMEL, T. Nonlinear control of vtol uavs incorporating flapping dynamics. In: . [S.l.: s.n.], 2013.

PACHTER, M.; CHANDLER, P. R.; MEARS, M. Reconfigurable tracking control with saturation. **Journal of Guidance, Control, and Dynamics**, v. 18, n. 5, p. 1016–1022, 1995. Disponível em: <<https://doi.org/10.2514/3.21499>>.

PARK, B.; OH, H. Vision-based obstacle avoidance for uavs via imitation learning with sequential neural networks. **International Journal of Aeronautical and Space Sciences**, v. 21, 02 2020.

PEREIRA, J.; LEITE, V.; RAFFO, G. Nonlinear model predictive control on se(3) for quadrotor aggressive maneuvers. **Journal of Intelligent Robotic Systems**, v. 101, 03 2021.

QIN, C.; YU, Q.; GO, H. S. H.; LIU, H. H.-T. Perception-aware image-based visual servoing of aggressive quadrotor uavs. **IEEE/ASME Transactions on Mechatronics**, v. 28, n. 4, p. 2020–2028, 2023.

QIU, Z.; HU, S.; LIANG, X. Model predictive control for uncalibrated and constrained image-based visual servoing without joint velocity measurements. **IEEE Access**, v. 7, p. 73540–73554, 2019.

QUIGLEY, M.; CONLEY, K.; GERKEY, B.; FAUST, J.; FOOTE, T.; LEIBS, J.; WHEELER, R.; NG, A. Y. et al. Ros: an open-source robot operating system. In: KOBE, JAPAN. **ICRA workshop on open source software**. [S.l.], 2009. v. 3, n. 3.2, p. 5.

RAKOTOMAMONJY, T.; TRUONG, Q. H. Helicopter ship landing using visual servoing on a moving platform 11this work has been accomplished within the frame of the research project containct, funded by onera general scientific grants. **IFAC-PapersOnLine**, v. 50, n. 1, p. 10507–10512, 2017. ISSN 2405-8963. 20th IFAC World Congress. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2405896317317901>>.

RECALDE, L. F.; VARELA, J.; GUEVARA, B. S.; ANDALUZ, V.; GIMENEZ, J.; GANDOLFO, D. Constrained visual servoing of quadrotors based on model predictive control. **IFAC-PapersOnLine**, v. 55, n. 37, p. 353–360, 2022. ISSN 2405-8963. 2nd Modeling, Estimation and Control Conference MECC 2022. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S240589632202852X>>.

REDMON, J.; DIVVALA, S.; GIRSHICK, R.; FARHADI, A. **You Only Look Once: Unified, Real-Time Object Detection**. 2016. Disponível em: <<https://arxiv.org/abs/1506.02640>>.

REDMON, J.; FARHADI, A. Yolov3: An incremental improvement. **arXiv**, 2018.

ROBERTS, R.; BARAJAS, M.; RODRIGUEZ-LEAL, E.; GORDILLO, J. L. Haptic feedback and visual servoing of teleoperated unmanned aerial vehicle for obstacle awareness and avoidance. **International Journal of Advanced Robotic Systems**, v. 14, n. 4, p. 1729881417716365, 2017. Disponível em: <<https://doi.org/10.1177/1729881417716365>>.

ROZANTSEV, A.; LEPETIT, V.; FUA, P. Flying objects detection from a single moving camera. In: **2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2015. p. 4128–4136.

SALES, A.; MIRA, P.; FILHO, J.; GARCIA, W.; NASCIMENTO, A. M.; AMORIM, T.; NASCIMENTO, T. Vision-based k-nearest neighbor approach for multiple search and landing with energy constraints. In: **2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE)**. [S.l.: s.n.], 2023. p. 71–76.

SANTANA, L.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. Navigation and cooperative control using the ar.drone quadrotor. **Journal of Intelligent & Robotic Systems**, v. 84, p. 1–24, 12 2016.

SAPKOTA, K. R.; ROELOFSEN, S.; ROZANTSEV, A.; LEPETIT, V.; GILLET, D.; FUA, P.; MARTINOLI, A. Vision-based unmanned aerial vehicle detection and tracking for sense and avoid systems. In: **2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**. [S.l.: s.n.], 2016. p. 1556–1561.

SAQIB, M.; KHAN, S. D.; SHARMA, N.; BLUMENSTEIN, M. A study on detecting drones using deep convolutional neural networks. In: **2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)**. [S.l.: s.n.], 2017. p. 1–5.

- SAQIB, M.; KHAN, S. D.; SHARMA, N.; BLUMENSTEIN, M. A study on detecting drones using deep convolutional neural networks. In: **2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)**. [S.l.: s.n.], 2017. p. 1–5.
- SCHUMANN, A.; SOMMER, L.; KLATTE, J.; SCHUCHERT, T.; BEYERER, J. Deep cross-domain flying object classification for robust uav detection. In: **2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)**. [S.l.: s.n.], 2017. p. 1–6.
- SEBORG, D. E.; EDGAR, T. F.; MELLICHAMP, D. A.; DOYLE, F. J. **Process dynamics and control**. 3rd. ed. Hoboken, NJ: John Wiley & Sons, 2010.
- SEPULCHRE, R.; PALEY, D. A.; LEONARD, N. E. Stabilization of planar collective motion with limited communication. **IEEE Transactions on Automatic Control**, v. 53, n. 3, p. 706–719, 2008.
- SERRA, P.; CUNHA, R.; HAMEL, T.; CABECINHAS, D.; SILVESTRE, C. Landing of a quadrotor on a moving target using dynamic image-based visual servo control. **IEEE Transactions on Robotics**, v. 32, n. 6, p. 1524–1535, 2016.
- SHAKERNIA, O.; MA, Y.; JOHN, T.; SASTRY, S. Landing an unmanned air vehicle: Vision based motion estimation and nonlinear control. **Asian Journal of Control**, v. 1, 11 1999.
- SHAO, L.; NAGATA, F.; OCHI, H.; OTSUKA, A.; IKEDA, T.; WATANABE, K.; HABIB, M. Visual feedback control of quadrotor by object detection in movies. **Artificial Life and Robotics**, v. 25, 06 2020.
- SHIN, Y.; CALISE, A. J.; JOHNSON, M. D. Adaptive control of advanced fighter aircraft in nonlinear flight regimes. **Journal of Guidance, Control, and Dynamics**, v. 31, n. 5, p. 1464–1477, 2008. Disponível em: <<https://doi.org/10.2514/1.30213>>.
- SIMHA, A.; TALLAM, M.; SHANKAR, H.; MURALISHANKAR, R.; H.N.L.N., S. Adaptive attitude control of the spherical drone on so(3). In: **2016 IEEE Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)**. [S.l.: s.n.], 2016. p. 90–94.
- SLOTINE, J.-J. E.; LI, W. Composite adaptive control of robot manipulators. **Automatica**, v. 25, n. 4, p. 509–519, 1989. ISSN 0005-1098. Disponível em: <<https://www.sciencedirect.com/science/article/pii/0005109889900940>>.
- SLOTINE, J.-J. E.; LI, W. **Applied Nonlinear Control**. Englewood Cliffs, NJ: Prentice Hall, 1991.

SMRCKA, D.; BACA, T.; NASCIMENTO, T.; SASKA, M. Admittance force-based uav-wall stabilization and press exertion for documentation and inspection of historical buildings. In: IEEE. **2021 International Conference on Unmanned Aircraft Systems (ICUAS)**. [S.l.], 2021. p. 552–559.

Stasinchuk, Y.; Vrba, M.; Petrlik, M.; Baca, T.; Spurny, V.; Hert, D.; Zaitlik, D.; Nascimento, T.; Saska, M. A Multi-UAV System for Detection and Elimination of Multiple Targets. In: IEEE. **2021 IEEE International Conference on Robotics and Automation (ICRA)**. [S.l.], 2021. p. 555–561.

TANG, S.; WÜEST, V.; KUMAR, V. Aggressive flight with suspended payloads using vision-based control. **IEEE Robotics and Automation Letters**, v. 3, n. 2, p. 1152–1159, 2018.

TAO, G.; KOKOTOVIC, P. Adaptive control of plants with unknown hystereses. **IEEE Transactions on Automatic Control**, v. 40, n. 2, p. 200–212, 1995.

THUILOT, B.; MARTINET, P.; CORDESSES, L.; GALLICE, J. Position based visual servoing: keeping the object in the field of vision. In: **Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)**. [S.l.: s.n.], 2002. v. 2, p. 1624–1629 vol.2.

TONG, P.; YANG, X.; YANG, Y.; LIU, W.; WU, P. Multi-uav collaborative absolute vision positioning and navigation: A survey and discussion. **Drones**, v. 7, n. 4, 2023. ISSN 2504-446X. Disponível em: <<https://www.mdpi.com/2504-446X/7/4/261>>.

TRUONG, Q. H.; RAKOTOMAMONJY, T.; TAGHIZAD, A.; BIANNIC, J.-M. Vision-based control for helicopter ship landing with handling qualities constraints. **IFAC-PapersOnLine**, v. 49, n. 17, p. 118–123, 2016. ISSN 2405-8963. 20th IFAC Symposium on Automatic Control in AerospaceACA 2016. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2405896316314938>>.

VRBA, M.; SASKA, M. Marker-less micro aerial vehicle detection and localization using convolutional neural networks. **IEEE Robotics and Automation Letters**, v. 5, n. 2, p. 2459–2466, 2020.

WANG, G.; QIN, J.; LIU, Q.; MA, Q.; ZHANG, C. Image-based visual servoing of quadrotors to arbitrary flight targets. **IEEE Robotics and Automation Letters**, v. 8, n. 4, p. 2022–2029, 2023.

WANG, Z.; HUANG, W.; YAN, Y. Vision-based bio-inspired guidance law for small aerial vehicle. **Chinese Journal of Aeronautics**, v. 28, n. 1, p. 260–267,

2015. ISSN 1000-9361. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1000936114002313>>.

WEI, P.; LIANG, R.; MICHELMORE, A.; KONG, Z. Vision-based 2d navigation of unmanned aerial vehicles in riverine environments with imitation learning. **Journal of Intelligent & Robotic Systems**, v. 104, 03 2022.

WEISS, L.; SANDERSON, A.; NEUMAN, C. Dynamic sensor-based control of robots with visual feedback. **IEEE Journal on Robotics and Automation**, v. 3, n. 5, p. 404–417, 1987.

WISE, K.; LAVRETSKY, E.; ZIMMERMAN, J.; FRANCIS, J.; DIXON, D.; WHITEHEAD, B. Adaptive flight control of a sensor guided munition. In: **AIAA Guidance, Navigation, and Control Conference and Exhibit**. [s.n.]. Disponível em: <<https://arc.aiaa.org/doi/abs/10.2514/6.2005-6385>>.

XIE, H.; FINK, G.; LYNCH, A. F.; JAGERSAND, M. Adaptive visual servoing of uavs using a virtual camera. **IEEE Transactions on Aerospace and Electronic Systems**, v. 52, n. 5, p. 2529–2538, 2016.

XIE, N.; YU, Y.; SUN, C. An image-based decoupling controller of quadrotor for moving target tracking. **Journal of Control and Decision**, v. 10, p. 1–12, 07 2022.

YANG, S.; SCHERER, S.; ZELL, A. An onboard monocular vision system for autonomous takeoff, hovering and landing of a micro aerial vehicle. **Journal of Intelligent & Robotic Systems**, v. 69, 09 2013.

YILDIZ, Y.; ANNASWAMY, A.; KOLMANOVSKY, I. V.; YANAKIEV, D. Adaptive posicast controller for time-delay systems with relative degree n_2 . **Automatica**, v. 46, n. 2, p. 279–289, 2010. ISSN 0005-1098. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0005109809005329>>.

YU, Z.; NONAMI, K.; SHIN, J.; CELESTINO, D. 3d vision based landing control of a small scale autonomous helicopter. **International Journal of Advanced Robotic Systems**, v. 4(1), 03 2007.

ZARUDZKI, M.; SHIN, H.-S.; LEE, C.-H. An image based visual servoing approach for multi-target tracking using an quad-tilt rotor uav. In: **2017 International Conference on Unmanned Aircraft Systems (ICUAS)**. [S.l.: s.n.], 2017. p. 781–790.

ZHANG, K.; SHI, Y.; SHENG, H. Robust nonlinear model predictive control based visual servoing of quadrotor uavs. **IEEE/ASME Transactions on Mechatronics**, v. 26, n. 2, p. 700–708, 2021.

ZHANG, S.; JI, H. Vision-based tracking control of quadrotor for a moving target with nonlinear angular velocity observer. **Asian Journal of Control**, v. 25, 03 2022.

ZHANG, T. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In: **Proceedings of the Twenty-First International Conference on Machine Learning**. New York, NY, USA: Association for Computing Machinery, 2004. (ICML '04), p. 116. ISBN 1581138385. Disponível em: <<https://doi.org/10.1145/1015330.1015332>>.

ZHANG, X.; FANG, Y.; ZHANG, X.; SHEN, P.; JIANG, J.; CHEN, X. Attitude-constrained time-optimal trajectory planning for rotorcrafts: Theory and application to visual servoing. **IEEE/ASME Transactions on Mechatronics**, v. 25, n. 4, p. 1912–1921, 2020.

ZHAO, B.; TANG, Y.; WU, C.; DU, W. Vision-based tracking control of quadrotor with backstepping sliding mode control. **IEEE Access**, v. 6, p. 72439–72448, 2018.

ZHAO, G.; CHEN, G.; CHEN, J.; HUA, C. Finite-time control for image-based visual servoing of a quadrotor using nonsingular fast terminal sliding mode. **International Journal of Control, Automation and Systems**, v. 18, 04 2020.

ZHENG, D.; WANG, H.; CHEN, W.; WANG, Y. Planning and tracking in image space for image-based visual servoing of a quadrotor. **IEEE Transactions on Industrial Electronics**, v. 65, n. 4, p. 3376–3385, 2018.

ZHENG, D.; WANG, H.; WANG, J.; CHEN, S.; CHEN, W.; LIANG, X. Image-based visual servoing of a quadrotor using virtual camera approach. **IEEE/ASME Transactions on Mechatronics**, v. 22, n. 2, p. 972–982, 2017.

ZHENG, D.; WANG, H.; WANG, J.; ZHANG, X.; CHEN, W. Toward visibility guaranteed visual servoing control of quadrotor uavs. **IEEE/ASME Transactions on Mechatronics**, v. 24, n. 3, p. 1087–1095, 2019.

ZHU, B.; ZHANG, B.; GE, Q. A nonlinear model predictive control based control method to quadrotor landing on moving platform. **Cognitive Computation and Systems**, v. 5, n. 2, p. 118–131, 2023. Disponível em: <<https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/ccs2.12081>>.

ZIEGLER, J. G.; NICHOLS, N. B. Optimum settings for automatic controllers. **Transactions of the ASME**, v. 64, p. 759–768, 1942.

ÅSTRÖM, K. J.; HÄGGLUND, T. **PID controllers: theory, design, and tuning**. Research Triangle Park, NC: Instrument Society of America, 1995.