



# ESCOLA NAVAL



*ta sainte & bi-faire*

**Frederico Guilherme Raposo Baptista Alves  
Rasinhãs**

**Localização e navegação em ambientes com  
disrupção de sinal GNSS, utilizando LIDAR e  
odometria visual**

**Dissertação para obtenção do Grau de Mestre em  
Ciências Militares Navais, na especialidade de Engenharia  
Naval Ramo de Armas e Eletrônica**



Alfeite

2025





# ESCOLA NAVAL

*talant de bi-faire*



**Frederico Guilherme Raposo Baptista Alves  
Rasinh**

*Localização e navegação em ambientes com interrupção de  
sinal GNSS, utilizando LIDAR e odometria visual*

Dissertação para obtenção do Grau de Mestre em  
Ciências Militares Navais, na especialidade de Engenharia Naval Ramo  
de Armas e Eletrónica

**Orientação de:** Bruno Duarte Damas

**Co-orientação de:** Vítor Rodrigues Viegas

*O Aluno Mestrando,*

*O Orientador,*

---

GMAR EN-AEL Alves Rasinh

---

Bruno Damas

Alfeite

2025



“Standing on the shoulders of Giants”

Isaac Newton



Dedico este trabalho ao meu avô.



# Agradecimentos

Este trabalho marca o culminar do meu trajeto na Escola Naval, um percurso desafiante e repleto de crescimento pessoal, mas concluído com a certeza inabalável de ter sido a melhor decisão da minha vida. Gostaria de expressar a minha profunda gratidão a todas as pessoas que, de forma direta ou indireta, influenciaram, apoiaram e enriqueceram este caminho.

Um especial e sentido agradecimento ao meu orientador, Professor Bruno Damas, pelo seu apoio incansável, dedicação permanente e pela orientação exemplar e inspiradora ao longo deste trajeto. Ao meu co-orientador, Professor Vítor Viegas, agradeço pelas valiosas contribuições e conselhos pertinentes para a concretização deste projeto.

À minha família, agradeço do fundo do coração pelo apoio incondicional, constante e generoso que sempre me proporcionaram, especialmente nesta fase tão importante da minha vida.

À minha mãe e ao meu padrasto, dedico uma palavra especial de gratidão: foram vocês que moldaram quem me tornei. Sem o vosso amor, exemplo e incentivo, nada disto teria sido possível. Obrigado por serem o alicerce sobre o qual assenta o homem que sou hoje.

Aos meus amigos Pedro, Luana e Inês, pela amizade sincera, constante e pura; por todo o apoio inabalável e pelos bons momentos vividos ao longo de todos estes anos. Obrigado, acima de tudo, por me fazerem sorrir.

À Beatriz, pela forma como entraste na minha vida, tornando-te rapidamente um porto seguro. Obrigado pelo carinho altruísta, presença constante e pelo apoio incondicional neste trajeto e em todos os que ainda virão.

Aos meus camaradas do curso Fernão de Magalhães, pela amizade e camaradagem demonstrados ao longo da nossa jornada, com um especial reconhecimento aos “engenheiros” nesta reta final.

A todos vós, deixo o meu mais sincero e profundo agradecimento!



# Resumo

Um dos desafios mais prementes no domínio da robótica é o desenvolvimento de sistemas autônomos capazes de estimar continuamente a sua posição e orientação sem depender de referências externas, especialmente face às vulnerabilidades associadas aos Sistemas Globais de Navegação por Satélite (GNSS). Esta necessidade é particularmente evidente em conflitos atuais na Europa e Médio Oriente, onde veículos não tripulados desempenham um papel crescente e essencial. Neste contexto, a presente dissertação centra-se no desenvolvimento de um sistema de localização autônomo robusto, aplicado a um veículo terrestre não tripulado ou *Unmanned Ground Vehicle* (UGV), o Leo Rover. O objetivo principal consiste em estimar com precisão e robustez a trajetória do veículo utilizando exclusivamente sensores integrados, nomeadamente, um sensor LIDAR, uma câmara, *encoders* nas rodas e uma *Inertial Measurement Unit* (IMU) de baixo custo. A abordagem adotada baseia-se numa fusão sensorial *loosely-coupled*, onde a pose é inicialmente estimada independentemente através de algoritmos específicos para cada sensor: a odometria visual recorre ao método de fluxo ótico Kanade-Lucas-Tomasi (KLT), sendo a escala relativa determinada pelo ajuste de pontos identificados ao plano do solo; enquanto a odometria LIDAR é obtida através do algoritmo *Iterative Closest Point* (ICP). Estas estimativas são posteriormente fundidas com medições provenientes das rodas e da IMU através de um *Error-State Extended Kalman Filter* (ESKF), numa estrutura modular flexível que facilita a integração ou exclusão de sensores adicionais. Para validar a abordagem proposta, foram realizados testes experimentais em ambientes interiores e exteriores, com diferentes níveis de estruturação, visibilidade e condições de iluminação, simulando cenários operacionais realistas. As trajetórias estimadas pelo sistema foram comparadas com medições de referência obtidas a partir de sensores externos, nomeadamente um recetor GNSS e um sistema de posicionamento em ambientes interiores (IPS). Esta comparação permitiu avaliar, de forma rigorosa, a precisão e a robustez do sistema desenvolvido.

**Palavras-chave:** Fusão Sensorial, Odometria, Odometria Visual, Odometria LIDAR, Error State Extended Kalman Filter



# Abstract

One of the most pressing challenges in the field of robotics is the development of autonomous systems capable of continuously estimating their position and orientation without relying on external references, particularly in light of the vulnerabilities associated with Global Navigation Satellite Systems (GNSS). This need is especially evident in current conflicts in Europe and the Middle East, where unmanned vehicles are playing an increasingly vital role. In this context, the present dissertation focuses on the development of a robust autonomous localization system applied to an unmanned ground vehicle (UGV), the Leo Rover. The main objective is to accurately and reliably estimate the vehicle's trajectory using only onboard sensors, namely a LIDAR sensor, a camera, wheel encoders and a low-cost Inertial Measurement Unit (IMU). The adopted approach relies on a loosely coupled sensor fusion strategy, where the pose is initially estimated independently using sensor-specific algorithms: visual odometry is based on the Kanade-Lucas-Tomasi (KLT) optical flow method, with relative scale determined by fitting identified points to the ground plane, while LIDAR odometry is computed using the Iterative Closest Point (ICP) algorithm. These independent estimates are then fused with wheel and IMU data through an Error-State Extended Kalman Filter (ESKF), in a flexible modular architecture that facilitates the integration or removal of additional sensors. To validate the proposed approach, experimental tests were conducted in both indoor and outdoor environments, under varying levels of structural complexity, visibility, and lighting conditions, simulating realistic operational scenarios. The estimated trajectories were compared against reference data obtained from external sensors, specifically a GNSS receiver and an Indoor Positioning System (IPS). This comparison enabled a rigorous assessment of the accuracy and robustness of the developed system.

**Keywords:** Sensor Fusion, Visual Odometry, LIDAR Odometry, Error State Extended Kalman Filter



# Declaração

Declaro que recorri a ferramentas de inteligência artificial generativa para correção ortográfica e semântica do texto.

# Índice

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Formulação do Problema . . . . .	2
1.2	Objetivos . . . . .	3
1.3	Estrutura da Dissertação . . . . .	4
<b>2</b>	<b>Enquadramento Teórico</b>	<b>5</b>
2.1	Sensores . . . . .	5
2.1.1	Unidades de Medida Inercial . . . . .	6
2.1.2	<i>Light Detection and Ranging</i> (LIDAR) . . . . .	6
2.1.3	Câmeras RGB . . . . .	8
2.2	Visão Computacional . . . . .	10
2.3	Representações de Posição e Orientação . . . . .	13
2.4	Odometria . . . . .	16
2.4.1	Odometria Inercial . . . . .	18
2.4.2	Odometria Visual . . . . .	19
2.4.3	Odometria Lidar . . . . .	22
2.5	Fusão Sensorial . . . . .	22
2.5.1	Filtros de Kalman . . . . .	23
2.5.2	Extended Kalman Filter . . . . .	25
2.5.3	Error-State Extended Kalman Filter . . . . .	26
<b>3</b>	<b>Estado da Arte</b>	<b>31</b>
3.1	Odometria LIDAR . . . . .	32
3.2	Odometria LIDAR-Inercial . . . . .	33
3.3	Odometria Visual-Inercial . . . . .	34
3.4	Odometria Visual-Inercial-LIDAR . . . . .	36
3.5	Desafios Identificados . . . . .	37
<b>4</b>	<b>Metodologia</b>	<b>39</b>
4.1	Leo Rover . . . . .	39

4.1.1	Características Físicas e Arquitetura de Hardware . . . . .	40
4.1.2	Arquitetura de Software . . . . .	41
4.2	Integração de Sensores no Sistema . . . . .	42
4.2.1	Integração do LIDAR . . . . .	42
4.2.2	Integração do recetor GNSS . . . . .	44
4.2.3	Integração do <i>Indoor Positioning System</i> . . . . .	45
4.3	Abordagem . . . . .	47
4.3.1	Caracterização da IMU . . . . .	50
4.3.2	Algoritmo VO . . . . .	51
4.3.3	Algoritmo LIDAR . . . . .	56
4.4	Error-State Extended Kalman Filter . . . . .	58
4.4.1	Rodas . . . . .	60
4.4.2	LIDAR e VO . . . . .	62
4.4.3	GNSS . . . . .	64
<b>5</b>	<b>Análise de Resultados</b>	<b>67</b>
5.1	Métricas Utilizadas . . . . .	67
5.2	Descrição dos Cenários Experimentais . . . . .	69
5.2.1	Trajeto 1 . . . . .	69
5.2.2	Trajeto 2 . . . . .	74
5.2.3	Trajeto 3 . . . . .	77
5.2.4	Trajeto 4 . . . . .	81
5.2.5	Trajeto 5 . . . . .	88
5.3	Considerações Adicionais . . . . .	91
<b>6</b>	<b>Conclusões</b>	<b>95</b>
6.1	Trabalho Futuro . . . . .	97
	<b>Bibliografia</b>	<b>99</b>
	<b>Anexos</b>	<b>107</b>
<b>I</b>	<b>Classificação e Exemplo de Sensores</b>	<b>107</b>
<b>II</b>	<b>Referencial LIDAR</b>	<b>109</b>
<b>III</b>	<b>Comparação de Trajetórias</b>	<b>111</b>
III.1	Trajeto 1 . . . . .	111
III.2	Trajeto 2 . . . . .	112
III.3	Trajeto 3 . . . . .	113

III.4 Trajeto 5 . . . . .	114
---------------------------	-----

# Lista de Figuras

1.1	Leo Rover utilizado na aquisição do <i>dataset</i> . . . . .	3
2.1	Funcionamento Simplificado de um LIDAR . . . . .	8
2.2	Captura de Imagem digital . . . . .	9
2.3	Filtro de Bayer . . . . .	9
2.4	Exemplificação da detecção de pontos de interesse recorrendo a FAST . . . . .	12
2.5	Modelo simplificado de veículo de duas rodas . . . . .	17
2.6	Diagrama simplificado de um SOI . . . . .	19
2.7	Exemplos de métodos de fusão sensorial . . . . .	22
4.1	Leo Rover . . . . .	39
4.2	Estrutura ROS do Leo Rover . . . . .	41
4.3	RPLIDAR-A3 . . . . .	43
4.4	RPLIDAR-A3 Integrado . . . . .	44
4.5	Módulo UBlox EVK-M8N . . . . .	44
4.6	Marvelmind Starter Set Super-MP-3D . . . . .	45
4.7	Representação do espaço na aplicação <i>Dashboard</i> . . . . .	46
4.8	Leo Rover após integração dos sensores . . . . .	46
4.9	Esquema Representativo da Abordagem Utilizada . . . . .	48
4.10	Referencial Leo Rover . . . . .	48
4.11	Medições com o rover estacionário . . . . .	50
4.12	Região de interesse (ROI) . . . . .	53
4.13	Representação geométrica da normal ao plano do solo e da distância $d$ no referencial da câmara . . . . .	56
5.1	Cenário experimental do Trajeto 1 . . . . .	70
5.2	Trajeto 1 — Estimativas por métodos isolados . . . . .	70
5.3	Trajeto 1 — Estimativa apenas com a IMU . . . . .	72
5.4	Trajeto 1 — Comparação entre as melhores estimativas . . . . .	73
5.5	Trajeto 2 — Comparação entre variantes da fusão IMU+LIDAR . . . . .	75
5.6	Trajeto 2 — Estimativas com ESKF . . . . .	76
5.7	Cenário experimental do Trajeto 3 . . . . .	78

5.8	<i>Frame</i> ilustrativo da detecção de <i>features</i> . . . . .	79
5.9	Trajeto 3 — Estimativas ESKF . . . . .	81
5.10	Comparação das diferentes estimativas obtidas pela odometria de rodas . . . . .	82
5.11	Estimativas com métodos isolados no Trajeto 4 . . . . .	83
5.12	Trajeto 5 — Estimativas ESKF . . . . .	85
5.13	Comparação de diferentes estimativas — IMU+ODO . . . . .	86
5.14	Comparação de diferentes estimativas — IMU+ODO+VO . . . . .	87
5.15	Comparação de diferentes estimativas — TOTAL . . . . .	87
5.16	Zona do Trajeto 5 . . . . .	88
5.17	<i>Frame</i> ilustrativo da detecção de <i>features</i> , no Trajeto 5 . . . . .	89
5.18	Trajeto 5 — Estimativas ESKF . . . . .	91
II.1	Referencial Lidar . . . . .	109
III.1	Trajeto 1 — Estimação da pose com o filtro ESKF e diferentes sensores de correção . . . . .	111
III.2	Trajeto 2 — Estimação da pose com os métodos isolados . . . . .	112
III.3	Trajeto 2 — Estimativa da pose com a combinação IMU+VO . . . . .	112
III.4	Trajeto 3 — Estimação da pose com os métodos isolados . . . . .	113
III.5	Trajeto 3 — Estimação da pose com os métodos isolados, sem distorção introduzida pelo LIDAR . . . . .	113
III.6	Estimativas com métodos isolados no Trajeto 5 . . . . .	114
III.7	Estimativas com métodos isolados no Trajeto 5 . . . . .	114

# Lista de Tabelas

5.1	Trajeto 1 — Avaliação dos Métodos Isolados . . . . .	71
5.2	Trajeto 1 — Avaliação dos Métodos Compostos com ESKF . . . . .	72
5.3	Trajeto 2 — Avaliação dos Métodos Isolados . . . . .	74
5.4	Trajeto 2 — Avaliação dos Métodos Compostos com ESKF . . . . .	77
5.5	Trajeto 3 — Avaliação dos Métodos Isolados . . . . .	79
5.6	Trajeto 3 — Avaliação dos Métodos Compostos com ESKF . . . . .	80
5.7	Trajeto 4 — Avaliação dos Métodos Isolados . . . . .	83
5.8	Trajeto com GNSS — Avaliação dos Métodos Compostos com ESKF . . . . .	84
5.9	Trajeto 5 — Avaliação dos Métodos Isolados . . . . .	89
5.10	Trajeto 5 — Avaliação dos Métodos Compostos com ESKF . . . . .	90
I.1	Classificação e Exemplos de Sensores . . . . .	107



# Lista de Siglas e Acrónimos

<b>AHRS</b>	<b>A</b> ttitude <b>H</b> eading <b>R</b> eference <b>S</b> ystem
<b>APE</b>	<b>A</b> bsolute <b>P</b> ose <b>E</b> rror
<b>CINAV</b>	<b>C</b> entro <b>I</b> nvestigação <b>N</b> aval
<b>CSV</b>	<b>C</b> omma- <b>S</b> eparated <b>V</b> alues
<b>EKF</b>	<b>E</b> xtended <b>K</b> alman <b>F</b> ilter
<b>ESKF</b>	<b>E</b> rror <b>E</b> state <b>E</b> xtended <b>K</b> alman <b>F</b> ilter
<b>FOV</b>	<b>F</b> ield <b>O</b> f <b>V</b> iew
<b>GNSS</b>	<b>G</b> lobal <b>N</b> avigation <b>S</b> atellite <b>S</b> ystem
<b>ICP</b>	<b>I</b> terative <b>C</b> losest <b>P</b> oint
<b>IMU</b>	<b>I</b> nertial <b>M</b> easurement <b>U</b> nit
<b>INS</b>	<b>I</b> nertial <b>N</b> avigation <b>S</b> ystem
<b>IO</b>	<b>I</b> nertial <b>O</b> dometry
<b>IPS</b>	<b>I</b> ndoor <b>P</b> ositioning <b>S</b> ystem
<b>KF</b>	<b>K</b> alman <b>F</b> ilter
<b>KLT</b>	<b>K</b> anade- <b>L</b> ucas- <b>T</b> omasi
<b>LIDAR</b>	<b>L</b> ight <b>D</b> etection <b>A</b> nd <b>R</b> anging
<b>MEMS</b>	<b>M</b> icro <b>E</b> lectro <b>M</b> echanical <b>S</b> ystems
<b>MSCKF</b>	<b>M</b> ulti- <b>S</b> tate <b>C</b> onstraint <b>K</b> alman <b>F</b> ilter
<b>NATO</b>	<b>N</b> orth <b>A</b> tlantic <b>T</b> reaty <b>O</b> rganization
<b>ORB</b>	<b>O</b> riented <b>F</b> AST and <b>R</b> otated <b>B</b> RIEF
<b>PE</b>	<b>P</b> erceção <b>E</b> xterna (Exteroception)
<b>PI</b>	<b>P</b> erceção <b>I</b> nterna (Proprioception)
<b>PNT</b>	<b>P</b> ositioning, <b>N</b> avigation, and <b>T</b> iming
<b>RANSAC</b>	<b>R</b> andom <b>S</b> ample <b>C</b> onsensus
<b>RMSE</b>	<b>R</b> oot <b>M</b> ean <b>S</b> quare <b>E</b> rror
<b>RFID</b>	<b>R</b> adio <b>F</b> requency <b>I</b> dentification
<b>ROS</b>	<b>R</b> obot <b>O</b> perating <b>S</b> ystem
<b>RPE</b>	<b>R</b> elative <b>P</b> ose <b>E</b> rror
<b>SIFT</b>	<b>S</b> cale <b>I</b> nvariant <b>F</b> eature <b>T</b> ransform
<b>SLAM</b>	<b>S</b> imultaneous <b>L</b> ocalization <b>A</b> nd <b>M</b> apping

<b>STO</b>	<b>Science and Technology Organization</b>
<b>SURF</b>	<b>Speeded Up Robust Features</b>
<b>TOF</b>	<b>Time Of Flight</b>
<b>UAV</b>	<b>Unmanned Aerial Vehicle</b>
<b>UGV</b>	<b>Unmanned Ground Vehicle</b>
<b>UUV</b>	<b>Unmanned Underwater Vehicle</b>
<b>USV</b>	<b>Unmanned Surface Vessel</b>
<b>UWB</b>	<b>Ultra Wide Band</b>
<b>VIO</b>	<b>Visual Inertial Odometry</b>
<b>VO</b>	<b>Visual Odometry</b>
<b>VINS</b>	<b>Visual-Inertial Navigation System</b>

# Capítulo 1

## Introdução

Um dos desafios mais prementes e emergentes no domínio da robótica é o desenvolvimento de sistemas de localização autônoma, ou seja, sistemas capazes de determinar, de forma contínua e precisa, a sua posição e orientação sem depender de referências externas. Esta capacidade é fundamental para permitir que plataformas robóticas realizem tarefas complexas, como navegação, seguimento de alvos e detecção e evasão de obstáculos.

A tendência atual aponta para uma crescente robotização dos conflitos armados, com um número cada vez mais elevado de plataformas não tripuladas a operar e cooperar no teatro de operações. Esta tendência tem sido verificada amplamente nos atuais conflitos na Europa e Médio Oriente, nomeadamente no conflito entre a Federação Russa e a Ucrânia, em que os *Unmanned Aerial Vehicles* (UAVs) de baixo custo desempenham um papel preponderante.

Atualmente, os *Global Navigation Satellite Systems* (GNSS) representam a principal fonte de informação para a estimação da posição, permitindo alcançar precisão na ordem dos centímetros e garantindo cobertura global (Egea-Roca et al. 2022). No entanto, a fiabilidade do sinal GNSS varia significativamente consoante o ambiente, sendo particularmente vulnerável em teatros operacionais onde interferências intencionais, como *jamming* e *spoofing*, são empregues para comprometer as capacidades de forças opositoras.

O problema em causa tem sido abordado em diversos contextos estratégicos e tecnológicos, incluindo o grupo de trabalho SET-309, da *Science and Technology Organization* (STO), da *North Atlantic Treaty Organization* (NATO). Este grupo reconhece que a NATO necessita de dados PNT (*Positioning, Navigation, and Timing*) robustos, resilientes e fidedignos, em tempo real para garantir o sucesso de missões, conduzidas em ambientes onde os sinais GNSS são sujeitos a interferências quer ambientais, quer intencionais.

O desenvolvimento de sistemas autónomos fiáveis é uma crescente área de interesse para a Marinha Portuguesa, permitindo a extensão da capacidade operacional através do emprego de veículos aéreos (UAVs), veículos terrestres (UGVs), veículos de superfície (USVs) e veículos subaquáticos (UUVs) não tripulados (Gouveia e Melo 2019).

## 1.1 Formulação do Problema

A presente dissertação incide sobre o desenvolvimento e a avaliação de métodos de localização alternativos ao GNSS, aplicados a um veículo terrestre não tripulado (UGV). O foco desta centra-se na exploração de técnicas de odometria baseadas em sensores visuais, nomeadamente uma câmara, sensores *Light Detection and Ranging* (LIDAR), sensores de medições inercias (*Inertial Measurement Unit*-IMU) e sensores presentes nas rodas do veículo.

O problema fundamental consiste em estimar, com precisão e robustez, o estado do veículo no referencial global, utilizando unicamente medições provenientes de sensores integrados, sem referência de sinais externos. Estas medições são expressas no referencial do corpo do mesmo, sendo combinadas de forma a inferir a pose global (posição e orientação) da plataforma no mundo.

Os métodos de odometria considerados, quando utilizados de forma isolada, tendem a apresentar estimativas enviesadas e cumulativamente imprecisas ao longo do tempo. Por conseguinte, torna-se essencial a fusão da informação proveniente das diferentes fontes, de forma a criar uma estimativa da trajetória mais exata.

A plataforma utilizada é o Leo Rover, um veículo terrestre não tripulado de quatro rodas, com arquitetura modular e controlo de direção por velocidade diferencial, apto a operar em ambientes interiores e exteriores. Apresenta dimensões aproximadas de  $424 \times 445 \times 303$  mm, um peso de 7 kg, e é capaz de atingir uma velocidade linear máxima de 0.4 m/s e uma velocidade angular máxima de 1 rad/s. Utiliza, como suporte nativo, o *middleware Robot Operating System* (ROS). Esta plataforma encontra-se, de origem, equipada com a câmara, IMU e encoders nas rodas, necessários no decurso da dissertação, sendo o sensor LIDAR integrado posteriormente. A Figura 1.1 ilustra a configuração final do Leo Rover com todos os sensores instalados.



FIGURA 1.1: Leo Rover utilizado na aquisição do *dataset*

## 1.2 Objetivos

O principal objetivo da presente dissertação consiste no desenvolvimento, integração e validação de um sistema de localização robusto, baseado na fusão de múltiplas fontes sensoriais, aplicado a uma plataforma robótica terrestre. Para tal, foi delineada uma metodologia composta pelas seguintes etapas:

- Recolher um *dataset* com medições provenientes dos sensores integrados no veículo, abrangendo cenários representativos de diferentes condições operacionais:
  - Ambientes interiores e exteriores;
  - Espaços estruturados e não estruturados;
  - Condições de visibilidade variadas.
- Implementar métodos de odometria visual e por LIDAR, capazes de estimar de forma autónoma o estado do veículo com base nos dados adquiridos.
- Integrar as estimativas obtidas através de um algoritmo de fusão sensorial baseado num *Error-State Extended Kalman Filter* (ESKF), segundo um esquema de *late fusion* que combina medições diretas dos sensores com resultados dos métodos de odometria.
- Validar a precisão do sistema proposto através da comparação com trajetórias de referência (*ground truth*) obtidas por sistemas externos de posicionamento.

Alguns dos principais desafios enfrentados neste trabalho relacionam-se com a necessidade de estimar com exatidão os *bias* associados à IMU, o que requer a medições absolutas, como as fornecidas por sistemas GNSS. Na ausência deste tipo de informação, os erros acumulados tornam-se dificilmente observáveis e propagam-se ao longo do tempo.

Adicionalmente, a qualidade das medições obtidas a partir do LIDAR e da câmera está fortemente dependente das condições ambientais. O LIDAR tende a apresentar limitações em ambientes pouco estruturados, campos abertos ou zonas com vegetação densa. Por outro lado, os métodos de odometria visual são particularmente sensíveis a variações nas condições de iluminação, apresentando um desempenho reduzido em cenários com baixa luminosidade ou escassez de *features* visuais.

### 1.3 Estrutura da Dissertação

A dissertação encontra-se organizada em seis capítulos, conforme descrito:

- O Capítulo 2, apresenta o enquadramento teórico aos conceitos abordados no decorrer da dissertação, com foco nos sensores, métodos de odometria e fusão sensorial.
- No Capítulo 3, é feita uma revisão da literatura, aferindo o estado da arte, nos temas abordados no capítulo anterior, alicerçando as escolhas feitas ao longo da dissertação.
- No Capítulo 4, é descrita a plataforma utilizada, os cenários de teste e a metodologia empregue.
- No Capítulo 5, são apresentados os processos e resultados experimentais, analisando detalhadamente o desempenho do método implementado.
- No Capítulo 6, são apresentadas as conclusões aferidas e abordadas futuras linhas de investigação.

# Capítulo 2

## Enquadramento Teórico

O presente capítulo tem como objetivo apresentar o enquadramento teórico necessário à compreensão dos métodos abordados ao longo da dissertação. A Secção 2.1 descreve os sensores utilizados, a sua constituição e os respetivos princípios de funcionamento. A Secção 2.2 introduz os conceitos fundamentais do processamento de imagem, com especial ênfase na deteção de características visuais e no cálculo de fluxo ótico, essenciais para a odometria visual. Na Secção 2.3 são abordadas as representações fundamentais de posição e orientação utilizadas. Na Secção 2.4 são analisados os principais tipos de odometria. Por fim, a Secção 2.5 aborda as principais técnicas de fusão sensorial baseadas no Filtro de Kalman.

### 2.1 Sensores

A presente secção pretende apresentar e detalhar vários sensores, e o seu modo de funcionamento. De acordo com Siciliano e Khatib (2016), podemos classificar sensores pelo que medem e como o medem. Assim, para aferição do próprio estado os autores referem-se a sensores de “proprioception” ou de percepção interna (PI), para aferição das condições ambientais que rodeiam o sistema, referem-se a sensores de “exteroception” ou de percepção externa (PE). Os sensores podem também ser categorizados como ativos (A) ou passivos (P) e ainda agrupados de acordo com o tipo de informação que disponibilizam. A Tabela I.1 presente no Anexo I, apresenta vários exemplos, alguns dos quais são explorados com maior detalhe ao longo da dissertação, devido à sua relevância.

### 2.1.1 Unidades de Medida Inercial

As *Inertial Measurement Units* (IMU), são dispositivos compostos por acelerómetros e giroscópios, projetados para fornecer dados relativos a acelerações lineares e velocidades angulares, exercidas no sistema em que estão integrados (Siciliano e Khatib 2016). Para medir essas grandezas em três dimensões, é necessário que tanto os acelerómetros, como os giroscópios operem em três eixos ortogonais entre si. Estes sensores fornecem, respetivamente, medições das acelerações lineares nas direções  $x$ ,  $y$  e  $z$ , bem como das velocidades angulares em torno dos mesmos eixos. Adicionalmente, para complementar a estimativa inicial da orientação do sistema, é possível integrar magnetómetros na IMU. Estes, fazendo recurso do campo magnético terrestre, permitem estimar um azimute inicial em relação ao Norte magnético, tendo por norma uma aplicação mais pronunciada no domínio da aeronáutica (Alteriis et al. 2021).

Atualmente, a vasta maioria das IMU, são implementadas sob a forma de circuitos integrados que se designam por *Microelectromechanical Systems* (MEMS), sendo estes com um comprimento compreendido entre 1  $\mu\text{m}$  e 1 mm e que combinam componentes eletrónicos e mecânicos (Gad-el-Hak 2001).

### 2.1.2 *Light Detection and Ranging* (LIDAR)

Os sensores *Light Detection and Ranging* (LIDAR), são dispositivos que se baseiam na reflexão de um feixe luminoso para determinar a distância a um objeto. Podem ter como principio de funcionamento dois meios de operação distintos: tempo de voo (*Time of Flight* - TOF) ou modulação quer em amplitude, quer em frequência do sinal emitido. Os primeiros são denominados métodos diretos de deteção, enquanto os segundos, uma vez que recorrem ao efeito de *Doppler* para determinação de distância e velocidade, são denominados métodos indiretos (Y. Li e Ibanez-Guzman 2020).

Nos métodos diretos, a estimativa da distância baseia-se na medição do tempo de voo de um feixe luminoso emitido por um transmissor, refletido num alvo e posteriormente detetado por um recetor, tipicamente um foto-detetor. A distância ao objeto é determinada a partir da multiplicação do tempo de voo pela velocidade da luz no meio de propagação, conforme expresso por:

$$2d = c \times t, \quad (2.1)$$

em que  $d$  representa a distância ao alvo,  $c$  a velocidade da luz no meio, e  $t$  o tempo de voo. Os feixes *laser* são usualmente utilizados como fonte luminosa devido à sua elevada colimação e reduzida largura espectral (Behroozpour et al. 2017).

A partir da equação 2.1, é possível inferir que erros na medição do tempo, conduzem a erros proporcionais na distância. Essa medição é feita pela detecção do pico da potência do sinal refletido. Reflexões múltiplas e ruído ambiental podem assim enviesar a medição. Com o aumento da distância a potência de sinal recebido irá diminuir, o que faz com que a probabilidade de erro aumente. Como medida para mitigar estes fatores é normal efetuar a média de várias medições. O alcance típico dos dispositivos utilizados em robótica situado entre os 10-100 m, com uma precisão entre os 5-10 mm (Siciliano e Khatib 2016).

Nos veículos autônomos, os sensores LIDAR são, por norma, utilizados para obter informação que permita reconstruir o panorama tridimensional do ambiente envolvente. Para isso, é necessário que os feixes emitidos cubram a totalidade do campo de visão desejado, ou *Field of View* (FOV), o qual pode ser bidimensional (2D) ou tridimensional (3D). Os LIDAR 2D realizam varrimentos num único plano, geralmente horizontal. Por sua vez, os LIDAR 3D captam informação em múltiplos planos.

A aquisição dos dados pode ser feita por iluminação simultânea de todos os pontos do FOV (*flash LIDAR*), por varrimento sequencial do feixe emissor, ou por uma combinação de ambas as abordagens (Behroozpour et al. 2017). Os feixes utilizados situam-se, geralmente, na região do infravermelho próximo, entre 780 e 3000 nm (*ISO 20473:2007 - Optics and photonics – Spectral bands* 2007). Atualmente, a solução mais comum em sistemas embarcados para veículos autônomos baseia-se num mecanismo rotativo com espelhos ou refletores, capaz de garantir um FOV de 360° (Y. Li e Ibanez-Guzman 2020).

A saída de um LIDAR é um conjunto de pontos, ou nuvem de pontos, que correspondem ao ambiente “varrido” cuja intensidade é correspondente à energia recebida pelo foto-detetor. A Figura 2.1, ilustra o princípio simplificado de funcionamento de um LIDAR, genérico.

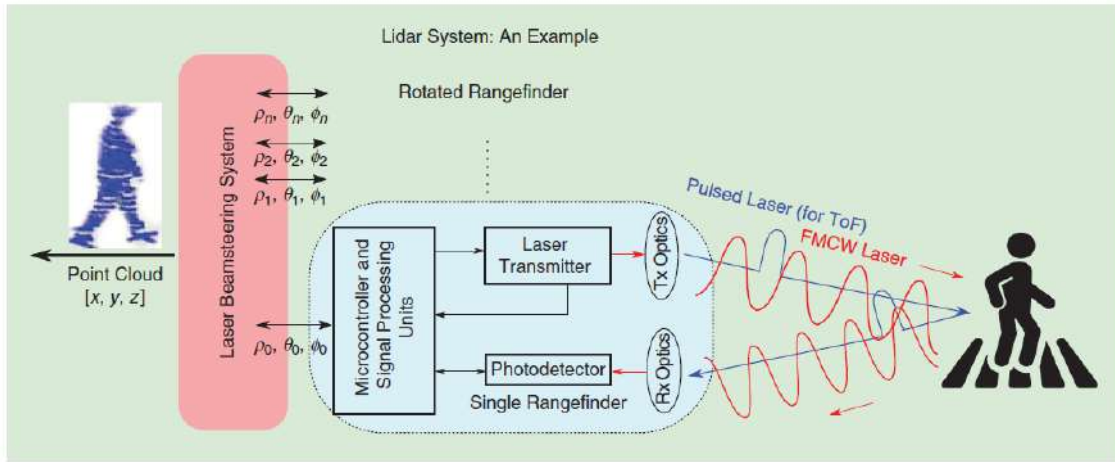


FIGURA 2.1: Funcionamento Simplificado de um LIDAR

Fonte: Y. Li e Ibanez-Guzman (2020)

A nuvem de pontos resultante serve como base para a odometria LIDAR. Na secção 3.1, será discutido em mais detalhe o processamento destas nuvens, bem como a integração da informação obtida no sistema.

### 2.1.3 Câmeras RGB

As câmeras RGB (*Red Green Blue*) são dispositivos cujo objetivo é captar imagens coloridas, em que o princípio de funcionamento baseia-se na deteção de radiação visível nas gamas de frequências correspondentes às três cores primárias: o vermelho, o verde e o azul.

A deteção ocorre quando a luz atravessa a lente da câmara e atinge os foto-detetores, que convertem a energia da luz incidente em corrente elétrica. Na maioria das aplicações, esta corrente é proporcional à intensidade luminosa incidente (Plataniotis e Venetsanopoulos 2000). Assim, como é possível descrever aproximadamente todas as cores no espectro da radiação visível através uma combinação de diferentes intensidades das três cores primárias, é viável replicar a cor do objeto real, digitalmente.

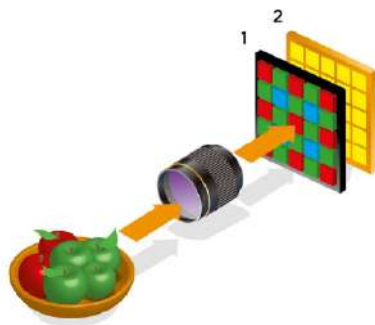


FIGURA 2.2: Captura de Imagem digital

Fonte : Adaptado de Canon (2025)

A figura 2.2 ilustra sucintamente a fase inicial deste processo. O componente 1 é a representação de um filtro, utilizado na maioria das aplicações, denominado por filtro de cor de *Bayer* (Bayer 1976). Este garante a melhor distribuição de foto-sensores sensíveis à incidência dos fotões, no respetivo comprimento de onda, garantido a mais fiel representação das cores reais. Na figura 2.3, podemos ver a das cores no filtro sendo o G, uma das componentes que apenas permite a passagem de luz verde, B a luz azul e R a vermelha. Os foto-detetores estão representados com o número 2, na mesma imagem.

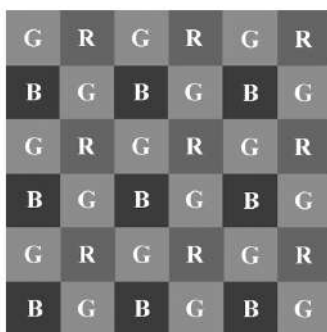


FIGURA 2.3: Filtro de Bayer

Fonte : Lukac et al. (2005)

Segundo Lukac et al. (2005), seguida da deteção e da conversão da intensidade luminosa em corrente elétrica, o sinal é convertido através de um conversor analógico-digital (A/D). Este sinal é então tratado por um circuito integrado dedicado, que organiza a informação num conjunto finito de elementos discretos denominados *pixels*, cada um com uma posição definida e com uma intensidade associada a cada uma das três cores primárias. Este processo permite gerar uma imagem digital representável no formato desejado.

## 2.2 Visão Computacional

A visão computacional clássica consiste no desenvolvimento de algoritmos e técnicas para a extração de informação a partir de imagens digitais e vídeos. Este campo depreende a utilização de métodos como o processamento de imagem, extração de pontos de interesse (*features*), reconstrução tridimensional e correspondência de *features* entre *frames* (Hartley e Zisserman 2003; Szeliski 2022).

O processamento de imagem corresponde então à manipulação sistemática de *pixels*, com o objetivo de maximizar a extração de informação desta. Para tal, são aplicadas diversas operações, nomeadamente o realce, a restauração, a filtragem e a suavização.

Entre as técnicas fundamentais deste processamento destacam-se a filtragem e a suavização, que visam principalmente reduzir o ruído e destacar elementos significativos. Para tal, são frequentemente utilizados filtros lineares como o filtro gaussiano, que reduz variações abruptas de intensidade ao aplicar uma convolução com uma função gaussiana (Gonzalez e Woods 2008). Em contraste, os filtros de alta frequência, como o de Sobel e Laplaciano, são aplicados para realçar detalhes estruturais relevantes. Os filtros de Sobel aplicam convoluções direcionais para realçar gradientes horizontais e verticais, permitindo a deteção de cantos e arestas. Já os filtros Laplacianos utilizam a segunda derivada para identificar zonas de transição abrupta de intensidade, realçando contornos em todas as direções (Szeliski 2022). Por norma, estes filtros são implementados recorrendo a *kernels*, que são matrizes (tipicamente  $3 \times 3$  ou  $5 \times 5$ ) aplicadas sobre a imagem através das operações de convolução. Cada elemento do *kernel* atua como um peso atribuído aos *pixels* vizinhos, determinado pela tipologia de filtro aplicada.

Por sua vez, o CLAHE (*Contrast Limited Adaptive Histogram Equalization*) permite uma melhoria adicional do contraste, subdividindo a imagem em vários segmentos e adaptando o contraste em cada um destes separadamente. Cada região é denominada de *tile* e para cada um é calculado o histograma do nível de intensidade de cinzento. Para limitar a amplificação de ruído, é imposto um limiar superior: os valores que o ultrapassam são considerados excesso e redistribuídos uniformemente por todos os *bins* do histograma, incluindo os originalmente saturados. Os *bins* correspondem a intervalos discretos que agrupam contagens de *pixels* por faixa de intensidade. Este processo assegura que o histograma final não amplifica desproporcionalmente qualquer nível de intensidade, mitigando o risco de realce excessivo

de ruído. Com o histograma redistribuído, calcula-se a função de distribuição cumulativa (CDF), que permite mapear os valores de intensidade da imagem original para os novos valores ajustados. Esta CDF define a transformação de intensidade a aplicar em cada *tile*. Por fim, todos os *tiles* processados são recombinaados através de interpolação, garantindo transições suaves entre regiões adjacentes e produzindo uma imagem global com contraste melhorado. (Pizer et al. 1987)

À fase de aplicação de filtros dá-se, por norma, o nome de pré-processamento. Concluída esta etapa, segue-se a fase de deteção e descrição de *features*, cuja identificação é fundamental para tarefas posteriores como a correspondência dos mesmos entre imagens ou a estimação de movimento. As *features* mais frequentemente extraídas são cantos, uma vez que apresentam variações de intensidade, isto é, gradientes, em múltiplas direções, o que os torna particularmente robustos e distintos.

Entre os algoritmos mais utilizados para deteção de cantos destaca-se o detetor de Harris (Harris, Stephens et al. 1988), que avalia a estrutura do gradiente numa vizinhança através da matriz de auto-correlação  $M$ , definida por:

$$M = \sum_{x,y} \omega(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (2.2)$$

onde  $I_x$  e  $I_y$  são os gradientes nas direções horizontal e vertical, e  $\omega(x,y)$  é uma função que pondera os valores locais. A probabilidade de um ponto ser um canto é então estimada pelo valor  $R$ , dado por:

$$R = \det(M) - k \cdot (\text{trace}(M))^2, \quad (2.3)$$

sendo  $\det(M)$  o determinante da matriz, e  $\text{trace}(M)$  a soma dos elementos da diagonal principal e  $k$  um parâmetro empírico que controla a sensibilidade do critério de Harris à deteção de cantos. Se  $R$  for positivo e elevado, o ponto é classificado com uma elevada probabilidade de ser um canto; se for negativo, assume-se ser uma aresta; se for próximo de zero, a região é considerada plana. O método de Shi-Tomasi (Shi et al. 1994) segue uma abordagem semelhante, mas utiliza diretamente o menor valor obtido da matriz como critério, eliminando a necessidade do parâmetro  $k$ . Para além destes, existem descritores robustos e invariantes a transformações geométricas e variações de iluminação, como o SIFT (*Scale-Invariant Feature Transform*) (Lowe 1999) e o SURF (*Speeded-Up Robust Features*) (Bay et al. 2008), que detetam pontos em múltiplas escalas e atribuem orientações dominantes.

Destaca-se também o FAST (*Features from Accelerated Segment Test*) (Rosten e Drummond 2006), especialmente concebido para aplicações em tempo real devido à sua elevada eficiência computacional. Este algoritmo baseia-se na análise da intensidade dos *pixels* ao longo de uma circunferência de 16 elementos em torno de um *pixel* central  $p$ . Considerando um limiar  $t$ , o ponto  $p$  é classificado como canto se existirem pelo menos 12 *pixels* contíguos ao longo da circunferência que sejam todos mais claros que  $I_p + t$  ou todos mais escuros que  $I_p - t$ , onde  $I_p$  representa a intensidade do *pixel* central. Para acelerar o processo, aplica-se um teste rápido preliminar sobre quatro posições fixas da circunferência (posições 1, 9, 5 e 13), permitindo excluir rapidamente *pixels* que não satisfaçam o critério. Os candidatos que passam neste teste são posteriormente avaliados segundo o critério completo. Por fim, aplica-se supressão não-máxima para eliminar múltiplos cantos próximos e garantir que apenas os mais significativos são retidos. O ORB (*Oriented FAST and Rotated BRIEF*) (Rublee et al. 2011), por sua vez, combina o FAST com um descritor binário orientado, permitindo maior robustez à rotação entre *frames*.

A Figura 2.4 ilustra um *frame* do *dataset* experimental. À esquerda, apresenta-se a imagem convertida para tons de cinzento e processada com CLAHE; à direita, a mesma imagem com o mesmo pré-processamento e *features* detetadas pelo algoritmo FAST.

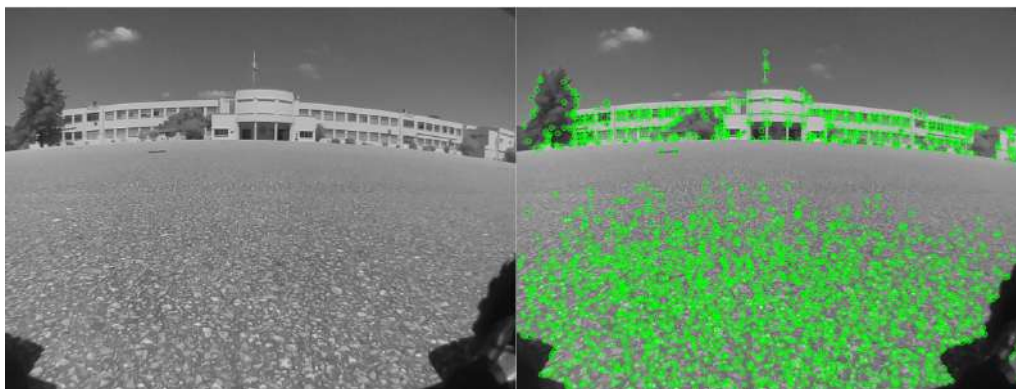


FIGURA 2.4: Exemplificação da deteção de pontos de interesse recorrendo a FAST

Atualmente a ferramenta mais utilizada para a implementação destes métodos é a biblioteca *Open Source Computer Vision Library* (OpenCV), que disponibiliza um vasto conjunto de funções e algoritmos de processamento e análise de imagem, amplamente empregues na comunidade científica e na indústria.

## 2.3 Representações de Posição e Orientação

O problema central na estimação do movimento de um corpo rígido consiste em descrever, de forma precisa, onde este se encontra e como está orientado no espaço, a cada instante. Esta descrição só pode ser feita relativamente a um outro sistema de referência, o que motiva a introdução formal de referenciais e as ferramentas necessárias para exprimir transformações entre estes.

De acordo com Siciliano e Khatib (2016), um referencial  $i$  é constituído por uma origem,  $O_i$ , e por um conjunto de três vetores mutuamente ortogonais,  $(\hat{x}_i, \hat{y}_i, \hat{z}_i)$ , todos fixos relativamente ao corpo em questão. Neste contexto, a posição de um qualquer ponto no espaço tridimensional,  $P$ , pode ser representada relativamente a um sistema de coordenadas ou referencial,  $i$ , através de um vetor ( $3 \times 1$ ) da forma:

$${}^iP = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}. \quad (2.4)$$

Para além da posição no espaço, é igualmente necessário representar a orientação do corpo. No domínio da robótica, as rotações podem ser descritas através de diversas representações, nomeadamente matrizes de rotação, representações eixo-ângulo, quaterniões e ângulos de Euler. No contexto da presente dissertação, o enfoque será colocado nas matrizes de rotação e nos quaterniões.

As matrizes de rotação permitem descrever a orientação de um referencial relativamente a outro. Suponha-se que o referencial  $i$  corresponde ao referencial do mundo, e o referencial  $j$  está acoplado ao corpo, com orientação arbitrária. A orientação de  $j$ , relativamente a  $i$ , pode então ser expressa através da matriz de rotação  ${}^iR_j$ , cujos elementos correspondem aos produtos escalares entre os vetores dos dois referenciais. Esta matriz é dada por:

$${}^i\mathbf{R}_j = \begin{pmatrix} \hat{x}_j \cdot \hat{x}_i & \hat{y}_j \cdot \hat{x}_i & \hat{z}_j \cdot \hat{x}_i \\ \hat{x}_j \cdot \hat{y}_i & \hat{y}_j \cdot \hat{y}_i & \hat{z}_j \cdot \hat{y}_i \\ \hat{x}_j \cdot \hat{z}_i & \hat{y}_j \cdot \hat{z}_i & \hat{z}_j \cdot \hat{z}_i \end{pmatrix}. \quad (2.5)$$

Cada linha da matriz representa a projeção de um vetor do referencial  $j$  sobre os eixos do referencial  $i$ , permitindo assim converter coordenadas expressas em  $j$  para coordenadas em  $i$  através de multiplicações matriciais.

De forma mais simplificada, uma rotação elementar em torno de um dos eixos do referencial pode ser expressa através de uma matriz de rotação,  $R(\theta)$ , que depende do ângulo  $\theta$  e do eixo de rotação. As rotações em torno dos eixos  $\hat{z}_j$ ,  $\hat{y}_j$  e  $\hat{x}_j$  são dadas, respetivamente, pelas seguintes matrizes:

$$R_Z(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad (2.6)$$

$$R_Y(\theta) = \begin{pmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{pmatrix}, \quad (2.7)$$

$$R_X(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{pmatrix}. \quad (2.8)$$

Torna-se então possível representar a posição e orientação de um qualquer ponto relativamente a um referencial arbitrário utilizando uma matriz de transformação. Esta, construída em coordenadas homogêneas, é composta por duas componentes: uma matriz de rotação  ${}^i_jR \in R^{3 \times 3}$ , que representa a orientação do referencial do corpo  $j$  relativamente ao referencial inercial  $i$ , e um vetor de translação  ${}^i t_{O_i} \in R^{3 \times 1}$ , que expressa a posição da origem do novo referencial em relação ao inicial. O ponto  $P$  pode então ser expresso como:

$${}^i_jP = \left[ \begin{array}{ccc|c} {}^i_j\mathbf{R} & & & {}^i t_{O_i} \\ 0 & 0 & 0 & 1 \end{array} \right]. \quad (2.9)$$

A composição sequencial das matrizes 2.6, 2.7 e 2.8, numa ordem específica, permite representar a orientação do corpo com apenas três parâmetros, denominados ângulos de Euler. Estes correspondem a rotações sucessivas em torno de eixos de um referencial móvel, sendo por isso dependentes da ordem escolhida. Por exemplo, a sequência  $(\alpha, \beta, \gamma)^\top$  representa uma rotação segundo a convenção de Euler (Z-Y-X), com rotações aplicadas, respetivamente, em torno dos eixos  $z$ ,  $y$  e  $x$ .

Apesar de amplamente utilizadas, as matrizes de rotação e a sua utilização sob a forma de ângulos de Euler apresentam algumas limitações práticas. As matrizes de rotação possuem redundância de informação, representando três graus de

liberdade através de nove elementos. Além disso, a sua forma concatenada está sujeita a fenômenos como o *gimbal lock*, em que ocorre a perda de um grau de liberdade devido a singularidades numéricas associadas à utilização de funções trigonométricas.

Neste sentido, torna-se vantajoso recorrer a outras representações mais eficientes e numericamente estáveis. Uma das mais utilizadas é a representação através de quatérnios unitários. Estes elementos do espaço de quatro dimensões,  $H$ , permitem representar rotações tridimensionais sem ambiguidade, evitando singularidades e simplificando a interpolação de orientações. Segundo Sola (2017), os quatérnios,  $Q$  são definidos como:

$$Q = a + bi + cj + dk \in H, \quad (2.10)$$

onde  $\{a, b, c, d\} \in R$  e  $\{i, j, k\} \in I$ , de modo que:

$$i^2 = j^2 = k^2 = ijk = -1, \quad (2.11)$$

tal que,

$$ij = -ji = k, \quad jk = -kj = i, \quad ki = -ik = j. \quad (2.12)$$

Esta notação segue a definição introduzida por Hamilton (1844), sendo amplamente adotada em contextos de robótica, justificando assim a sua utilização ao longo da presente dissertação. No entanto, existem outras convenções relevantes, nomeadamente, a empregue no domínio aeroespacial pelo *Jet Propulsion Laboratory* (JPL) e pela *European Space Agency* (ESA), as quais diferem na ordem dos componentes e na forma como o produto de quatérnios é definido.

A partir da formulação de Hamilton, define-se então o quatérnio como:

$$q = \begin{bmatrix} q_w \\ q_v \end{bmatrix} = \begin{bmatrix} q_w \\ q_x \\ q_y \\ q_z \end{bmatrix}, \quad (2.13)$$

onde  $q$  representa um quatérnio composto por uma componente escalar,  $q_w$ , e uma componente vetorial,  $q_v = [q_x \ q_y \ q_z]^\top$ .

Segundo Sola (2017), um quatérnio unitário, isto é, com norma 1, representa uma rotação tridimensional ao codificar um ângulo  $\theta \in R$  e um eixo unitário  $u \in R^3$ , segundo a expressão:

$$q = \begin{bmatrix} q_w \\ q_v \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ u \sin\left(\frac{\theta}{2}\right) \end{bmatrix} \quad (2.14)$$

A rotação de um vetor  $x \in R^3$  é então realizada através do produto de quatérnios:

$$x' = q \otimes x \otimes q^* \quad (2.15)$$

onde  $x$  é representado como um quatérnio puro ( $[0, x]$ ), e  $q^*$  é o conjugado de  $q$ . Esta operação corresponde a uma rotação passiva do vetor no espaço.

Tanto as matrizes de rotação como os quatérnios expressam rotações tridimensionais pertencentes ao grupo especial ortogonal  $SO(3)$ . Este representa o conjunto de todas as transformações lineares em  $R^3$  que preservam o comprimento e orientação relativa de vetores.

Este grupo desempenha um papel central na robótica e navegação, pois descreve as possíveis rotações de corpos rígidos no espaço tridimensional. Um corpo é considerado rígido se as distâncias, ângulos e orientações relativas entre os seus pontos permanecerem invariantes durante o movimento.

## 2.4 Odometria

A palavra odometria, deriva de uma contração das palavras gregas *hodos*, que significa trajeto ou viagem e *metron*, que significa medida. No contexto da robótica e veículos autónomos, odometria é utilizada para se referir, no seu significado mais clássico, à utilização de dados provenientes dos atuadores de locomoção, como rodas ou lagartas, para estimar o movimento do veículo (Siciliano e Khatib 2016). O conceito base segundo Dudek e Jenkin (2010), é o desenvolvimento de um modelo matemático que descreva com base nas atuações dos atuadores (rodas, articulações, lagartas, etc.), como é afetado o movimento total do veículo.

Considere-se um modelo simplificado de um veículo de duas rodas que apenas se pode movimentar num plano bidimensional,  $(xy)$ , representado na Figura 2.5.

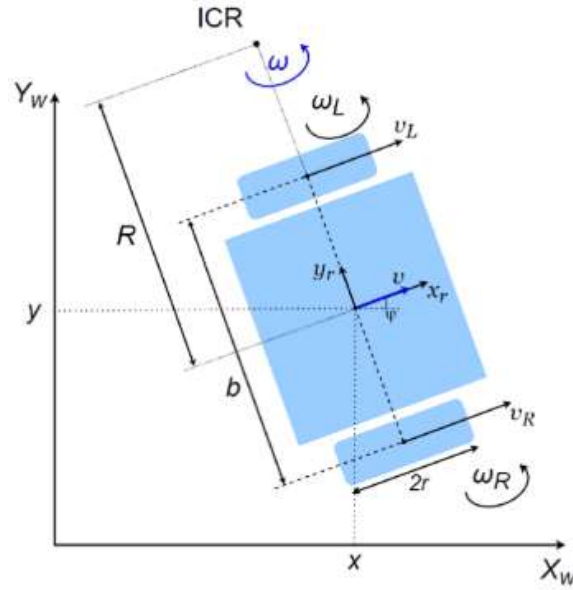


FIGURA 2.5: Modelo simplificado de veículo de duas rodas  
 Fonte : Adaptado de Filip et al. (2023)

Obtém-se então que,

$$\begin{cases} B\omega = \frac{v_R - v_L}{b} \\ Bv = \frac{v_R + v_L}{2}, \end{cases} \quad (2.16)$$

onde  $B\omega$  é a velocidade angular do veículo, no referencial do centro de massa do veículo, e  $Bv$  a velocidade linear deste no mesmo referencial, com  $v_R$  e  $v_L$  sendo a velocidade linear de cada uma das rodas e  $b$  a distância entre elas. A partir destas equações pode-se então, efetuar uma mudança de referencial destas velocidades para o referencial do mundo, recorrendo a uma matriz de rotação em torno do eixo  $z$ . Considerando que o veículo não é holonómico e não existem perturbações que o forcem, assume-se que não existe deslocamento lateral, então  $Bv_y = 0$ . Considerando  $\psi$  como o rumo, obtém-se,

$$\begin{bmatrix} wv_x \\ wv_y \\ w\omega \end{bmatrix} = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} Bv \\ 0 \\ B\omega \end{bmatrix}. \quad (2.17)$$

De onde resulta,

$$\begin{cases} {}^w v_x = Bv \cos(\psi) \\ {}^w v_y = Bv \sin(\psi) , \\ {}^w \omega = B\omega \end{cases} \quad (2.18)$$

em que  ${}^w v_x$  é a velocidade linear no referencial do mundo, ao longo do eixo  $x$ ,  ${}^w v_y$  a velocidade linear no eixo  $y$ , do mesmo referencial e  ${}^w \omega$  a velocidade angular, também no referencial do mundo. Por fim, integrando as diferentes velocidades conseguimos então obter as respectivas posições no referencial do mundo,

$$\begin{cases} p_x(t) = p_x(0) + \int_0^t {}^w v_x dt \\ p_y(t) = p_y(0) + \int_0^t {}^w v_y dt, \\ \psi(t) = \psi(0) + \int_0^t {}^w \omega dt \end{cases} \quad (2.19)$$

Assim, obtém-se um modelo simplificado que permite estimar a posição no referencial do mundo com base nas velocidades aplicadas a cada uma das rodas. No entanto, este modelo assume condições ideais e não contempla diversas fontes de incerteza, tais como erros na medição da velocidade efetivamente aplicada a cada roda, efeitos de atrito, irregularidades do terreno (exceto se a orientação 3D for considerada, como em 2.18), não linearidades do sistema real, entre outras.

Desta forma, surgem variantes da odometria que, com base em princípios semelhantes, estimam a posição do sistema no mundo usando diferentes tipos de sensores. Na presente dissertação são explorados os métodos de odometria, para além do clássico, compatíveis com os sensores integrados no sistema: odometria inercial, visual e LIDAR.

### 2.4.1 Odometria Inercial

A Odometria Inercial (OI) é uma técnica de localização e navegação, que utiliza dados provenientes da IMU para estimar a posição e a orientação de um objeto ao longo do tempo. Este processo baseia-se na integração sucessiva das acelerações e velocidades angulares medidas, partindo de condições iniciais conhecidas de posição, velocidade e atitude (Woodman 2007). Estes métodos são auto-contidos, isto é, após a aferição das condições iniciais, não são necessárias mais referências externas para determinar a informação desejada. Porém, estes métodos dependem da qualidade das IMU utilizadas. Estas apresentam erros estocásticos ao longo do seu tempo de operação, o que leva a que acumulação de incerteza nas medições, que

por sua vez conduz a erros de deriva na estimativa da posição velocidade e atitude (Narasimhappa et al. 2019).

A Figura 2.6 é uma representação simplificada de um sistema de odometria inercial (SOI) generalista.

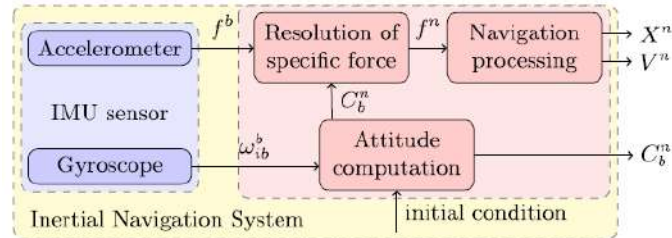


FIGURA 2.6: Diagrama simplificado de um SOI

Fonte : Mohamed et al. (2019)

Nesta figura é ilustrada a estrutura de um sistema que a partir da condição inicial conhecida, através das medições da IMU, se obtém a posição e velocidade atuais da plataforma.  $f^b$  e  $\omega_{ib}^b$  representam as acelerações lineares, e as velocidade angulares, nos 3 eixos, medidas pelos acelerômetros e giroscópios, respetivamente, no referencial do corpo.  $C_b^n$  é a matriz de rotação que representa a transformação de referencial do corpo para o mundo. Esta mesma matriz, é multiplicada por  $f^b$ , para se obter as acelerações representadas no referencial do mundo. Após esta operação, então a unidade de “Navigation Processing” deduz a posição e velocidade neste mesmo referencial,  $X^n$  e  $V^n$ , respetivamente (Mohamed et al. 2019).

## 2.4.2 Odometria Visual

O termo odometria visual, *Visual Odometry* (VO), foi introduzido por Nistér et al. (2004) para designar um método de estima da localização e navegação inteiramente baseado na análise de imagens, não sendo necessário conhecer previamente as condições iniciais do sistema. O nome advém da analogia com a odometria clássica. Tal como nesta se estima a pose do veículo pela integração das rotações das rodas, na VO a pose é inferida ao longo do tempo através da análise das variações sucessivas nas imagens captadas, resultantes do movimento relativo da câmara (Scaramuzza e Fraundorfer 2011).

Pode-se então classificar os métodos de VO, consoante os sensores empregues ou a abordagem utilizada. Relativamente às câmeras/sensores, podem ser *stereo*, monoculares, *stereo* ou monoculares omnidirecionais e RGB-D. Uma câmara RGB-D

é um sistema que, além de capturar imagens coloridas como abordado anteriormente, também regista informação relativa à profundidade para cada *pixel* (Sturm et al. 2012).

No que diz respeito à abordagem, distinguem-se três categorias principais. Os métodos baseados em *features* que envolvem a extração de pontos de interesse e a sua correspondência entre imagens consecutivas. Os métodos baseados em “aparência” que analisam variações diretas de intensidade dos *pixels* entre imagens. Por fim, os métodos híbridos que combinam ambos os paradigmas (Aqel et al. 2016). As abordagens baseadas em *features* recorrem à utilização dos detetores e descritores de imagem apresentados na Sec. 2.2.

Entre os métodos baseados em “aparência”, destaca-se o algoritmo de Lucas–Kanade, cuja formulação original foi apresentada em Lucas e Kanade (1981). Este método assume que a intensidade luminosa de um ponto de imagem se mantém constante ao longo de pequenos intervalos temporais e que o movimento do padrão de intensidade local é aproximadamente uniforme dentro de uma vizinhança espacial restrita. Com base nestas hipóteses, a estimativa do deslocamento de um ponto de interesse é obtida resolvendo um sistema de equações lineares derivado da expansão de Taylor da equação de intensidade,  $I(x+u, y+v, t+1) \approx I(x, y, t)$ . Esta expansão conduz à equação do fluxo ótico:

$$I_x u + I_y v + I_t = 0$$

em que  $I_x$ ,  $I_y$  e  $I_t$  representam, respetivamente, os gradientes espaciais e o gradiente temporal da intensidade da imagem. Para estimar o vetor de movimento, considera-se uma janela de  $n$  *pixels* e resolve-se o sistema de equações em torno dos pontos de interesse.

O método foi posteriormente refinado por Tomasi e Kanade (1991), que introduziram critérios para a seleção de pontos de interesse bem condicionados, resultando no algoritmo conhecido como KLT (*Kanade–Lucas–Tomasi*).

Apesar da sua eficiência computacional e precisão em cenários de movimento suave, o método KLT apresenta limitações em cenários de baixo contraste, variações abruptas de iluminação ou em presença de oclusões. Para mitigar esses efeitos, é comum integrar técnicas de rejeição de *outliers*, como o método RANSAC e empregar critérios adicionais de validação para garantir a robustez das correspondências.

O método *Random Sample Consensus* (RANSAC) (Fischler e Bolles 1981) permite identificar e excluir automaticamente correspondências inconsistentes entre

pontos de interesse em imagens consecutivas. A sua estratégia baseia-se na seleção aleatória de subconjuntos mínimos de dados para estimar uma transformação candidata compatível com um modelo geométrico definido. Esse modelo pode corresponder, por exemplo, a uma nuvem de pontos, uma matriz essencial ou um plano tridimensional. Em cada iteração, avalia-se quantos dos dados se ajustam ao modelo proposto dentro de um determinado limiar de erro, sendo classificados como *inliers*. As correspondências que não satisfazem esse critério são consideradas *outliers* e descartadas. O processo é repetido várias vezes, sendo retido o modelo com maior número de *inliers*, o qual pode ser posteriormente refinado com base nesses dados consistentes.

O RANSAC é amplamente utilizado tanto em abordagens baseadas em *features*, como em abordagens baseadas em “aparência”. Nos primeiros, é aplicado para validar correspondências entre descritores extraídos de pontos de interesse, permitindo estimar com robustez transformações geométricas. Já nas abordagens como o fluxo ótico KLT, o RANSAC é utilizado para filtrar vetores de movimento inconsistentes entre pontos de interesse rastreados, assegurando que apenas os deslocamentos compatíveis com um modelo global de movimento são considerados na estimação final.

No caso da VO monocular, particularmente relevante para a presente dissertação devido à arquitetura do sistema, tanto o movimento relativo como a reconstrução tridimensional do espaço devem ser inferidos a partir de dados bidimensionais obtidos por uma única câmera. Com cada nova imagem (*frame*), a escala relativa e a posição da câmera em relação aos *frames* anteriores é atualizada (Scaramuzza e Fraundorfer 2011). Por norma, métodos que envolvam câmeras monoculares, exigem pelo menos 3 *frames* diferentes para reconstruir o ambiente tridimensional (M. Li e Mourikis 2013). Como salientado por Mohamed et al. (2019), uma das limitações mais significativas consiste no facto do vetor de translação entre os dois primeiros *frames* ser apenas indicativo de uma direção, uma vez que não se dispõe *a priori* da escala. Por sua vez, outro dos desafios desta abordagem reside precisamente na determinação da escala absoluta, dado que a informação de profundidade não está disponível de forma direta. Devido a estes fatores, segundo Mourikis e Roumeliotis (2007), a forma mais comum de mitigar estas limitações é recorrer à fusão com sensores complementares, como IMU ou LIDAR, que permitem estimar a matriz de transformação inicial e atribuir um fator de escala ao sistema.

### 2.4.3 Odometria Lidar

A odometria LIDAR baseia-se na estimativa da posição relativa do veículo a partir de distâncias obtidas a conhecenças através de LIDAR. Estes métodos, recorrem normalmente a algoritmos *Iterative Closest Point* (ICP), para reconstruir o ambiente (Rusinkiewicz e Levoy 2001). Estes algoritmos comparam nuvens de pontos sequenciais, de forma a encontrar a matriz de rotação e translação que transforma a primeira na segunda, conseguindo assim recuperar o movimento relativo do sistema (Du et al. 2007).

## 2.5 Fusão Sensorial

A fusão sensorial é um processo que visa integrar a informação de diversos sensores para estimar, com maior exatidão, o estado de um sistema e o ambiente que o rodeia.

De acordo, com Mohamed et al. (2019), métodos de odometria que combinam informação de dois ou mais sensores, podem ser categorizados em duas vertentes distintas de fusão dos dados: abordagens baseadas em filtros e abordagens baseadas na otimização. Existe também a divisão no momento e fase de processamento em que os dados são fundidos, surgindo assim os métodos referidos em literatura diversa como *loosely-coupled* e *tightly-coupled* ou *early-fusion* e *late-fusion*.

Nos métodos *loosely-coupled* ou de *late-fusion*, a fusão ocorre numa fase posterior de processamento, após ser estimada a pose, independentemente, com a informação de cada sensor. A estima final do estado é então obtida através da combinação dessas estimativas parciais, utilizando métodos baseados em filtros ou técnicas de otimização (Konolige et al. 2011). A Figura 2.7a ilustra esta arquitetura, considerando como exemplo uma IMU e uma sequência de imagens adquiridas por uma câmara. Os dados de cada sensor são processados separadamente, produzindo estimativas da pose, que são subsequentemente integradas para gerar a estima final.

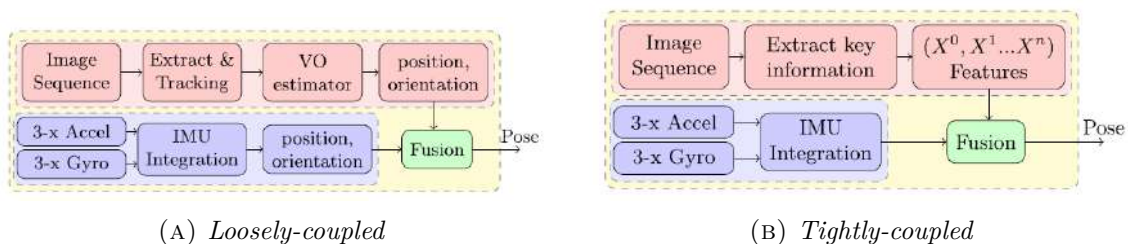


FIGURA 2.7: Exemplos de métodos de fusão sensorial

Fonte: Mohamed et al. (2019)

Nos métodos *tightly-coupled* ou de *early-fusion*, a estimativa da pose resulta da combinação dos dados de provenientes de cada sensor, antes da estimativa independente. Os dados “em bruto” são fundidos numa fase inicial. A Figura 2.7b apresenta um exemplo de diagrama de blocos, de uma estrutura generalista destes métodos.

Como mencionado anteriormente, existem diferentes abordagens para realizar a fusão de dados, nomeadamente as baseadas em filtros e as baseadas em otimização. As primeiras recorrem, geralmente, à utilização de Filtros de Kalman ou suas variantes, com o objetivo de propagar e atualizar a estimativa do estado com base em modelos probabilísticos de erro, modelos de observação, equações de movimento e medições. Por sua vez, as abordagens baseadas em otimização formulam o problema como uma minimização de erro entre as estimativas de estados anteriores e as observações disponíveis, permitindo uma estimação mais precisa do estado atual. Estas, são frequentemente utilizadas em contextos de *Simultaneous Localization and Mapping* (SLAM), com vista a melhorar simultaneamente a estima da trajetória e a construção do mapa.

Na presente secção, são abordados com mais detalhes os Filtros de Kalman e suas derivantes, tendo em conta a implementação desenvolvida no âmbito da dissertação.

### 2.5.1 Filtros de Kalman

O Filtro de Kalman (KF) é um dos mais conhecidos algoritmos para fusão sensorial e estima de estado de sistemas. Partindo do trabalho apresentado em Kalman (1960), podemos então compreender que o KF, é um algoritmo que permite estimar o estado de um qualquer sistema linear sujeito a ruído, quer na ação quer nas medições. Uma das grandes vantagens deste modelo é a sua recursividade, isto é, não ser necessário guardar todas os passados estados e atuações, apenas sendo necessários os estados e medições anteriores e as medições atuais. Para tal, o problema é dividido em duas grandes etapas, previsão de estado e atualização. Este modelo assume, porém, sistemas lineares e uma distribuição de ruído Gaussiana.

Assumindo  $x_k$  como o estado de um dado sistema, discreto, e  $x_{k-1}$  o seu estado anterior, conhecendo o modelo dinâmico deste, podemos escrever

$$x_k = F_k x_{k-1} + B_k u_{k-1} \quad (2.20)$$

em que, assumindo linearidade no sistema temos  $F_k$  com uma matriz de previsão, que relaciona ambos os estados pelo modelo dinâmico do sistema e  $B_k$  como uma matriz de controlo que relaciona as atuações externas  $u_k$ .

O KF representa o estado, em cada instante  $k$ , por uma distribuição com média  $x_k$  e covariância  $P_k$

$$P_k = F_k P_{k-1} F_k^T + Q_k, \quad (2.21)$$

com  $Q_k$ , como uma matriz que traduz incerteza associada ao ambiente.

Através da combinação das equações 2.20 e 2.21 temos a fase de predição. A fase de atualização tem em conta as medições realizadas e o erro associado a estas, para assim melhor calcular a nova estimativa do estado do veículo. Assim, surge,

$$z_k = H_k x_k + v_k, \quad (2.22)$$

com  $z_k$  como a observação atual,  $H_k$  uma matriz de observações e  $v_k$  como uma incerteza adicional associada às medidas.

A fase de atualização do estado é dada pelas respetivas equações:

$$\hat{x}'_k = \hat{x}_k + K'(z_k - H_k \hat{x}_k), \quad (2.23)$$

$$P'_k = P_k - K' H_k P_k, \quad (2.24)$$

$$K' = P_k H_k^T (H_k P_k H_k^T + R_k)^{-1}, \quad (2.25)$$

em que,  $\hat{x}_k$  é a estimativa *a priori* do estado, no instante  $k$  e  $\hat{x}'_k$ , a estimativa *a posteriori* do estado, no instante  $k$ .  $P_k$  e  $P'_k$  as matrizes de covariância do erro de estimativa *a posteriori* e *a priori*, respetivamente.  $K'$  é uma matriz que representa o ganho de Kalman, que determina um coeficiente na atualização da estimativa, que tem uma direta correlação com o grau de confiança que atribuímos ao modelo dinâmico ou aos sensores. Este ganho permite assim diminuir a incerteza associada à medição, isto é, se consideramos que os sensores no nosso veículo são bastante exatos, em relação ao modelo dinâmico, atribuímos uma maior relevância a estes para a estimativa do estado. A matriz de observação que relaciona o estado do sistema, com as medições, no instante  $k$  é dada por  $H_k$  e  $R_k$  representa a matriz de covariância do ruído de medição, no instante  $k$ .

Deste modo, o funcionamento do Filtro de Kalman pode resumir-se como um processo cíclico, alternando entre as etapas de previsão e atualização. A fase

de previsão utiliza o modelo dinâmico para estimar o estado seguinte e a respetiva incerteza, enquanto a fase de atualização incorpora novas observações, ajustando estas estimativas e reduzindo a incerteza associada ao estado estimado. Este ciclo iterativo assegura que a estimativa do estado se mantém sempre atualizada, refletindo a melhor combinação possível entre as informações disponíveis provenientes do modelo dinâmico e dos sensores.

### 2.5.2 Extended Kalman Filter

O Filtro de Kalman, descrito anteriormente, fornece uma estimativa ótima mediante condições de linearidade do sistema e uma distribuição de ruído gaussiana. Contudo, a maioria dos sistemas reais apresenta não-linearidades, o que inviabiliza a aplicação direta do KF clássico. Surge então uma variante designada *Extended Kalman Filter* (EKF), proposta em Julier e Uhlmann (1997), que adapta o KF ao tratamento de sistemas não-lineares através da linearização das funções envolvidas.

No EKF, as não-linearidades do veículo são aproximadas por meio de linearizações locais sucessivas. Segundo Wan e Van Der Merwe (2000), esta linearização é obtida através de uma expansão em série de Taylor de primeira ordem em torno do estado estimado, sendo que a qualidade da estimativa resultante depende diretamente da precisão desta aproximação.

Assim, o EKF mantém exatamente a mesma estrutura iterativa do KF clássico, alternando entre as fases de previsão e atualização. A principal diferença reside na obtenção das matrizes  $F_k$  e  $H_k$ , que são agora calculadas localmente como as Jacobianas das funções não-lineares de transição de estado  $f(x, u)$  e de observação  $z(x)$ , respetivamente:

$$F_k = \left. \frac{\partial f}{\partial x} \right|_{\hat{x}_{k-1}, u_{k-1}}, \quad (2.26)$$

$$H_k = \left. \frac{\partial z}{\partial x} \right|_{\hat{x}_k}. \quad (2.27)$$

Mantendo-se a restante estrutura matemática do KF clássico, as equações de previsão e atualização mantêm-se idênticas às apresentadas anteriormente. Desta forma, o EKF generaliza o Filtro de Kalman ao tratamento de sistemas não-lineares, através de aproximações sucessivas, preservando a mesma lógica iterativa e recursiva.

### 2.5.3 Error-State Extended Kalman Filter

Embora o EKF permita lidar com sistemas não lineares, a sua aplicação direta em problemas de navegação com orientação tridimensional e sensores inerciais revela-se desafiante. Estas dificuldades manifestam-se sob a forma de instabilidades numéricas, perda de consistência estatística e necessidade frequente de reajuste das matrizes de covariância, sobretudo quando os desvios sistemáticos dos sensores (*bias*) não são explicitamente incluídos no vetor de estado. Para contornar estas limitações, foi proposto em Roumeliotis et al. (1999) o *Error-State Extended Kalman Filter* (ESKF), uma reformulação do EKF que melhora a robustez e a consistência da estimação em sistemas não lineares.

De acordo com Sola (2017), o ESKF fundamenta-se na separação do estado verdadeiro do sistema em duas componentes: o estado nominal  $\hat{x}$  e o estado de erro  $\delta x$ . O estado nominal representa a melhor estimativa do sistema com base na integração direta das leituras da IMU, sem considerar explicitamente os ruídos nem perturbações do modelo. Por sua vez, o estado de erro capta pequenas perturbações acumuladas, modeladas por um sistema dinâmico linear com matrizes que variam no tempo e dependem do estado nominal. Paralelamente à estimação do estado nominal ocorre a estima do estado de erro.

A integração direta da IMU leva à acumulação de erro no estado nominal. Estas perturbações tornam-se observáveis apenas quando estão disponíveis medições externas, tipicamente mais esparsas. Nesses momentos, o estado de erro é corrigido, e o erro estimado  $\delta x$  é injetado no estado nominal, atualizando-o. Em seguida, o estado de erro é reinicializado a zero e a matriz de covariância associada é atualizada.

O estado nominal é então descrito pelas seguintes equações:

$$\dot{p} = v \tag{2.28a}$$

$$\dot{v} = R(a_m - a_b) + g \tag{2.28b}$$

$$\dot{q} = \frac{1}{2} q \otimes (\omega_m - \omega_b) \tag{2.28c}$$

$$\dot{a}_b = 0 \tag{2.28d}$$

$$\dot{\omega}_b = 0 \tag{2.28e}$$

$$\dot{g} = 0, \tag{2.28f}$$

onde a posição  $p \in R^3$  resulta da integração temporal da velocidade  $v$ . Esta, por sua vez, decorre da integração das medições de aceleração da IMU,  $a_m$ , corrigidas

pelo respetivo *bias*  $a_b$ . Como estas medições estão expressas no referencial do corpo, é necessário transformá-las para o referencial do mundo através da matriz de rotação  $R$ , obtida a partir do quaternião de orientação  $q$ . A esta aceleração transformada é somado o vetor da gravidade  $g$ , assumido constante. O autor ressalva, contudo, que em algumas aplicações é desejável estimar dinamicamente o vetor da gravidade, em vez de o considerar fixo.

A orientação do corpo é representada por um quaternião unitário  $q \in H$ , cuja evolução temporal é descrita pela equação (2.28c). Nesta, o operador  $\otimes$  representa a multiplicação de quaterniões, que permite atualizar a orientação a partir da velocidade angular medida  $\omega_m$ , corrigida pelo *bias*  $\omega_b$ .

Note-se que, embora no estado nominal os *bias*  $a_b$  e  $\omega_b$  sejam assumidos constantes, no estado de erro são tratados como variáveis.

As equações do estado de erro são definidas da seguinte forma:

$$\dot{\delta p} = \delta v \tag{2.29a}$$

$$\dot{\delta v} = -R[a_m - ab] \times \delta \theta - R\delta a_b + \delta g - Ra_n \tag{2.29b}$$

$$\dot{\delta \theta} = -[\omega_m - \omega_b] \times \delta \theta - \delta \omega_b - \omega_n \tag{2.29c}$$

$$\dot{\delta a_b} = a_w \tag{2.29d}$$

$$\dot{\delta \omega_b} = \omega_w \tag{2.29e}$$

$$\dot{\delta g} = 0, \tag{2.29f}$$

onde  $\delta p$ ,  $\delta v$  e  $\delta \theta$  representam os erros de posição, velocidade e orientação, respetivamente, em relação ao estado nominal.

A equação (2.29a) indica que o erro de posição propaga-se integrando o erro de velocidade. A equação (2.29b), descreve como o erro de velocidade é influenciado por erros na orientação (através do termo  $[a_m - ab] \times \delta \theta$ ), por desvios no viés do acelerómetro  $\delta a_b$ , e pelo viés na definição da gravidade  $\delta g$ , caso este seja incluído no vetor de estado. O termo  $a_n$  representa o ruído associado às medições do acelerómetro.

Na equação (2.29c), o erro na orientação é influenciado por desvios no viés do giroscópio  $\delta \omega_b$  e pelo ruído de medição  $\omega_n$ .  $a_w$  e  $\omega_w$  modelam respetivamente os vieses do acelerómetro e do giroscópio como processos de ruído. Finalmente, a equação (2.29f) assume que o vetor da gravidade é constante, sendo o seu erro modelado como estacionário.

Pode-se então representar o sistema, na ausência de medições externas adicionais, através dos seguintes vetores:

$$\mathbf{x} = \begin{bmatrix} p \\ v \\ q \\ a_b \\ \omega_b \\ g \end{bmatrix}, \quad \delta x = \begin{bmatrix} \delta p \\ \delta v \\ \delta \theta \\ \delta a_b \\ \delta \omega_b \\ \delta g \end{bmatrix}, \quad u_m = \begin{bmatrix} a_m \\ \omega_m \end{bmatrix}, \quad i = \begin{bmatrix} v_i \\ \theta_i \\ a_i \\ \omega_i \end{bmatrix}, \quad (2.30)$$

com  $x$  como o vetor de estado nominal,  $\delta x$  o vetor de estado de erro,  $u_m$  o vetor das medições e  $i$  o vetor das perturbações. Este último agrega os termos que modelam a incerteza nas medições das acelerações lineares, velocidades angulares e nos respetivos vieses, representando as suas covariâncias.

Obtém-se então, que o estado de erro é agora dado por:

$$\delta x \leftarrow f(x, \delta x, u_m, i) = F_x(x, u_m) \cdot \delta x + F_i \cdot i,$$

onde  $\delta x$  torna-se função do produto da matriz Jacobiana do estado erro ( $F_x$ ) com o próprio vetor de erro anterior, somado à distribuição gaussiana das covariâncias modeladas pela matriz Jacobiana do vetor de perturbação  $F_i$  multiplicada pelo próprio vetor  $i$ .

A equação de predição do estado do erro do ESKF é então dada por,

$$P = F_x P F_x^\top + F_i Q_i F_i^\top, \quad (2.31)$$

onde,  $\delta x \sim \mathcal{N}(\hat{\delta x}, P)$ , sendo  $Q_i$  a matriz de covariância das perturbações. As Jacobianas  $F_x$ ,  $F_i$  e a matriz  $Q_i$  são então definidas como:

$$F_x = \left. \frac{\partial f}{\partial \delta x} \right|_{x, u_m} = \begin{bmatrix} I & I\Delta t & 0 & 0 & 0 & 0 \\ 0 & I & -R[a_m - a_b]_\times \Delta t & -R\Delta t & 0 & I\Delta t \\ 0 & 0 & R^\top[(\omega_m - \omega_b)\Delta t]_\times & 0 & -I\Delta t & 0 \\ 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & 0 & I \end{bmatrix} \quad (2.32)$$

$$F_i = \left. \frac{\partial f}{\partial i} \right|_{x, u_m} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad Q_i = \begin{bmatrix} V_i & 0 & 0 & 0 \\ 0 & \Theta_i & 0 & 0 \\ 0 & 0 & A_i & 0 \\ 0 & 0 & 0 & \Omega_i \end{bmatrix}. \quad (2.33)$$

Seguida da fase de previsão do estado de erro, surge então a fase de correção do mesmo, através das medições realizadas por outros sensores. Esta fase envolve três etapas principais:

- medição e correção do estado de erro,
- injeção do novo estado de erro no estado nominal,
- *reset* do estado de erro.

Define-se então a função de observação  $h$ , correspondente ao sensor com uma determinada covariância  $V$ . Desta forma, implementa-se a matriz Jacobiana de observações,  $H$ , em relação a estado de erro, cuja formulação é a seguinte,

$$H \equiv \left. \frac{\partial h}{\partial \delta x} \right|_x. \quad (2.34)$$

Com isso calcula-se o ganho de Kalman:

$$K = PH^\top (HPH^\top + V)^{-1}, \quad (2.35)$$

e atualiza-se a matriz de covariância do estado de erro,  $P$ :

$$P \leftarrow (I - KH)P. \quad (2.36)$$

Após a atualização do estado erro, dada pela formulação acima, segue-se então a injeção deste no estado nominal, da seguinte forma,

$$p \leftarrow p + \delta\hat{p} \quad (2.37a)$$

$$v \leftarrow v + \delta\hat{v} \quad (2.37b)$$

$$q \leftarrow q \otimes q\{\delta\hat{\theta}\} \quad (2.37c)$$

$$a_b \leftarrow a_b + \delta\hat{a}_b \quad (2.37d)$$

$$\omega_b \leftarrow \omega_b + \delta\hat{\omega}_b \quad (2.37e)$$

$$g \leftarrow g + \delta\hat{g}. \quad (2.37f)$$

Por último, realiza-se o reestabelecimento do estado de erro, o que implica a atualização da respetiva matriz de covariância, de acordo com:

$$P \leftarrow GPG^\top, \quad (2.38)$$

onde  $G$  é a matriz Jacobiana da função  $g()$ , que representa a operação de reparametrização do estado de erro após a correção do estado nominal. Esta etapa assegura que a covariância continue consistente com a nova definição do erro. No caso mais simples, em que  $g$  é linear ou próximo da identidade, assume-se  $G = I_{18}$ . De facto, em muitas implementações práticas esta atualização é frequentemente omitida, o que equivale a considerar implicitamente  $G = I$ .

Conclui-se, assim, o enquadramento teórico essencial à compreensão dos princípios que regem a perceção e estimação de estado em sistemas robóticos, com especial enfoque nos modelos sensoriais, métodos de odometria e algoritmos de fusão sensorial. Estabelecida esta base conceptual, avança-se para a análise crítica das principais abordagens descritas na literatura, com o objetivo de identificar soluções consolidadas, metodologias emergentes e tendências relevantes que sustentam e fundamentam as opções adotadas na implementação desenvolvida.

# Capítulo 3

## Estado da Arte

À luz dos recentes conflitos na Europa e no Médio Oriente, tornou-se evidente a crescente utilização de meios autónomos no panorama operacional. Veículos não tripulados de baixo custo, rápida empregabilidade e arquitetura modular passaram a assumir um papel central nas operações militares.

Contudo, a maioria dos sistemas aéreos e de superfície depende fortemente de sinais GNSS para estimar a sua posição com precisão. Conscientes desta vulnerabilidade, forças opositoras recorrem frequentemente a técnicas como o *jamming* ou o *spoofing*, comprometendo assim a eficácia e fiabilidade operacional destes sistemas. O *jamming* consiste na emissão deliberada de sinais de elevada potência na mesma gama de frequências dos sinais GNSS, provocando a saturação do recetor e a consequente perda de informação. Já o *spoofing* baseia-se na transmissão de sinais falsificados, modelados para imitar os legítimos, conduzindo o recetor a calcular posições incorretas.

Os efeitos destas técnicas, aliados às limitações inerentes à propagação do sinal GNSS, tornam-se particularmente evidentes em cenários urbanos e de guerrilha, onde as operações decorrem em zonas densamente edificadas ou em espaços interiores, naturalmente privados de cobertura. Embora estas fragilidades sejam conhecidas há mais de três décadas, a atual intensificação do uso de sistemas autónomos reforça a urgência no desenvolvimento de soluções alternativas e robustas para localização e navegação.

Neste enquadramento, o presente capítulo aborda o estado da arte relativo aos métodos de odometria e às estratégias de fusão sensorial considerados relevantes para esta dissertação.

### 3.1 Odometria LIDAR

O método proposto por J. Zhang, Singh et al. (2014), denominado LOAM (Lidar Odometry and Mapping), implementa um sistema de SLAM baseado em LIDARs 3D, que explora características geométricas do ambiente, nomeadamente arestas e superfícies planas, extraídas a partir de nuvens de pontos fornecidas pelo sensor. A identificação destas características baseia-se na análise da variação da distância entre um ponto  $i$  e os seus vizinhos  $j \in S$  dentro da mesma nuvem, traduzida no cálculo de um coeficiente de suavidade  $c$ , definido como:

$$c = \frac{1}{|S| \cdot \|X_{(k,i)}^L\|} \sum_{\substack{j \in S \\ j \neq i}} \|X_{(k,i)}^L - X_{(k,j)}^L\|, \quad (3.1)$$

em que  $S$  representa o conjunto de pontos vizinhos do ponto  $i$  na nuvem considerada e  $X_{(k,i)}^L$  corresponde ao vetor posição do ponto  $i$  no referencial do sensor  $L$ , no instante  $k$ .

Esta métrica quantifica a irregularidade geométrica local em torno de cada ponto, permitindo distinguir pontos conspícuos (valores elevados de  $c$ ) de superfícies planas (valores reduzidos de  $c$ ). A normalização pelo número de vizinhos  $|S|$  e pela distância ao sensor  $\|X_{(k,i)}^L\|$  assegura robustez relativamente à densidade e profundidade do ponto. A segmentação em arestas e planos permite definir correspondências mais estruturadas, como ponto-linha e ponto-plano, reduzindo a ambiguidade típica das correspondências ponto-a-ponto e aumentando significativamente a robustez da estimativa de movimento.

O LOAM divide o processo de SLAM em duas componentes que operam em paralelo e a diferentes frequências. A componente de odometria, executada a uma frequência elevada (10 Hz), estima a deslocação relativa entre varrimentos consecutivos, resolvendo uma otimização não linear com recurso ao método de Levenberg–Marquardt, que ajusta iterativamente os parâmetros do movimento de modo a minimizar o erro global entre as correspondências geométricas. Em paralelo, a componente de mapeamento corre a uma frequência mais baixa (1 Hz) e realiza um alinhamento mais preciso da nuvem de pontos no referencial global, atualizando incrementalmente o mapa 3D do ambiente. A integração opcional de dados provenientes de uma IMU contribui para mitigar os efeitos de movimentos rápidos ou ambientes pouco estruturados, melhorando a robustez da estimativa de pose.

No trabalho desenvolvido por Shan e Englott (2018), é proposto um método

de odometria LIDAR, otimizado para veículos terrestres com recursos computacionais limitados, baseado no LOAM, mencionado anteriormente. O sistema, denominado LEGO-LOAM, incorpora um processo de segmentação da nuvem de pontos de modo a melhorar a eficiência. Inicialmente, procede-se à extração dos pontos correspondentes ao solo através de uma avaliação coluna a coluna, que não assume que os pontos pertencem ao mesmo plano. Esses pontos são classificados como pertencentes ao solo e excluídos da segmentação, o que reduz o ruído e aumenta a robustez na extração de *features*.

Os pontos remanescentes são agrupados em segmentos, a partir dos quais se extraem *features* distintas com base num índice de rugosidade local de forma semelhante ao método LOAM. As *features* planas provenientes do solo são utilizadas para estimar variações em  $z$  e os ângulos de *roll* e *pitch*; enquanto os restantes são utilizados para estimar movimento ao longo de  $x$ ,  $y$  e os ângulos de *yaw*.

Esta abordagem permite reduzir o tempo de processamento em aproximadamente 35 % a 48 % em relação ao LOAM original, mantendo a precisão da estimativa de pose.

## 3.2 Odometria LIDAR-Inercial

O método apresentado em Meyers et al. (2020), propõe um método de fusão sensorial *tightly-coupled* entre dados de LIDAR e IMU, implementado através de um *factor graph*. Neste grafo, cada tipo de sensor contribui com restrições específicas, *factors*, que em conjunto com os estados prévios do sistema (*nodes*), através de métodos de otimização, geralmente não lineares, permitem melhorar a estimativa da pose (Dellaert, Kaess et al. 2017). O LIO-SAM recorre à pré-integração dos dados inerciais, seguindo a abordagem de Forster et al. (2016), para estimar o movimento relativo entre instantes e remover a distorção dos dados LIDAR. Esta pré-integração gera fatores que são otimizados em conjunto com os fatores de odometria LIDAR, permitindo corrigir os vieses da IMU e alinhar com maior exatidão as nuvens de pontos. Além disso, o sistema adota um mecanismo eficiente baseado em *keyframes* e uma janela deslizante para preservar a performance em tempo real. O ponto forte do LIO-SAM reside na sua modularidade e capacidade de integrar sensores adicionais (como o GNSS) no grafo, mostrando resultados competitivos com métodos anteriores e elevada robustez a ambientes diversos.

O sistema FAST-LIO (Xu e F. Zhang 2021) implementa uma solução de

odometria LIDAR-inercial *tightly-coupled*, baseada num Filtro de Kalman Estendido Iterativo (IEKF). Esta abordagem recorre à fusão direta dos pontos extraídos do LIDAR com as medições da IMU, evitando uma fase intermédia de registo de *scans*. O IEKF é uma extensão do EKF clássico em que, a cada iteração da fase de atualização, a função de observação é novamente linearizada em torno da estimativa atual do estado. Esta abordagem é particularmente vantajosa em cenários altamente não lineares, como UAVs sujeitos a rápidas mudanças de orientação e aceleração, ou em ambientes com reduzida estrutura.

Durante o intervalo de tempo entre medições LIDAR, o sistema executa a propagação do estado com base nas medições da IMU, utilizando um modelo de movimento contínuo e a respetiva propagação da incerteza através das equações comuns do EKF. Contudo, como os pontos LIDAR são adquiridos de forma sequencial ao longo do tempo, é necessário compensar a distorção de movimento induzida durante o *scan*. Para tal, é realizada uma retro-propagação individual de cada ponto até ao instante final do *scan*, utilizando o estado estimado. Os pontos compensados são então projetados para o referencial global e utilizados na construção de resíduos geométricos, definidos como a distância ao plano ou aresta correspondente no mapa previamente construído. A inovação é então incorporada através de uma forma otimizada do ganho de Kalman, cujo cálculo foi reformulado para depender apenas da dimensão do estado (tipicamente 18), evitando a inversão de matrizes de grande dimensão associadas ao número elevado de pontos LIDAR, devido ao facto de ser um método *tightly-coupled*. Derivado deste trabalho surgiu também o trabalho de Xu, Cai et al. (2022), FAST-LIO2.

### 3.3 Odometria Visual-Inercial

A odometria Visual-Inercial surge como uma técnica que combina métodos de odometria Visual e Inercial, de modo a colmatar as deficiências de cada um. Segundo Mohamed et al. (2019), métodos de localização baseados em odometria visual são significativamente afetados por condições ambientais, tais como, luz, sombras e imagens desfocadas. Por sua vez, os métodos inerciais, baseados na informação fornecida pela IMU, têm a vantagem de não ser afetados pelo ambiente, mas por norma o erro é cumulativo, isto é, a incerteza tende a aumentar com o tempo. Assim, integrando os dois métodos, conseguimos um sistema mais robusto e preciso. Surge assim a *visual inertial-odometry*(VIO).

A revisão da literatura na presente secção é centrada na análise de métodos

que utilizem uma câmera monocular, considerando a sua relevância neste contexto. Esta prende-se com o facto de uma câmera monocular apresentar um reduzido custo, quando comparada com sistemas *stereo* e RGB-D, o que no contexto militar é desejável uma vez que estes algoritmos podem ser aplicados a vários veículos dispensáveis, tornando assim o custo de produção e implementação destes métodos relevante. Para esta escolha é também relevante o facto de que a plataforma utilizada, o Leo Rover, possuir apenas uma câmera.

O método apresentado por Qin et al. (2018), designado VINS-Mono, combina de forma *tightly-coupled* a informação proveniente de uma câmera monocular e de uma IMU para estimar o movimento do sistema. Para rastrear o movimento relativo da câmera entre imagens consecutivas, o sistema utiliza o método de Lucas-Kanade, também conhecido como fluxo ótico KLT (Lucas e Kanade 1981), classificado como *appearance-based*, abordado na secção 2.4.2. A estimação do estado é realizada através de uma otimização não linear, por método de *factor-graph*, sobre uma janela deslizante de 20 a 30 poses, onde são consideradas conjuntamente as medidas visuais e inerciais. A informação da IMU é pré-integrada entre *frames*, gerando medições que são otimizadas juntamente com os fatores visuais, permitindo assim estimar com maior precisão a trajetória e obter a escala relativa. Ao utilizar uma janela deslizante, VINS-Mono realiza múltiplas otimizações locais em vez de acumular o histórico completo, garantindo menor custo computacional e mantendo precisão.

O sistema proposto por Bloesch et al. (2015) apresenta uma abordagem de odometria visual-inercial baseada num Filtro de Kalman Estendido (EKF) que utiliza diretamente os erros de intensidade em excertos (*patches*) de imagem como termo de inovação na fase de atualização. A estimativa do estado é realizada segundo uma perspectiva centrada no sistema, em que cada ponto visual observado é representado por um vetor de direção (no referencial da câmera,  $\mu_i$ ) e um parâmetro de distância. Esta distância é inferida através de uma formulação de distância inversa, na qual a profundidade é dada por  $d_i = 1/\rho_i$ , sendo  $\rho_i$  estimado pelo filtro. No momento da deteção, este valor é inicializado com uma estimativa fixa e uma variância elevada, permitindo a sua posterior correção ao longo do tempo. À medida que o sistema se desloca e observa o mesmo ponto a partir de diferentes posições, o filtro compara a projeção esperada do ponto na imagem com a sua posição real observada. A discrepância entre ambas gera um termo de inovação, que é incorporado no EKF para atualizar a estimativa da distância. A combinação dos parâmetros  $\mu_i$  e  $\rho_i$  permite reconstruir a posição tridimensional do ponto no referencial da câmera por multiplicação escalar, sendo esta posteriormente transformada para o referencial

global com base na pose estimada. Esta abordagem reduz o número de variáveis associadas a cada marco evitando reparametrizações. As *features* visuais são detetadas com recurso ao algoritmo FAST, o qual identifica cantos na imagem com base em variações locais de intensidade. Os pontos detetados são posteriormente avaliados segundo um critério baseado na métrica de Shi–Tomasi. Para garantir uma distribuição espacial uniforme das *features* ao longo da imagem, é aplicada uma técnica (*bucketing*), que limita o número de pontos por região. O número total de *features* ativas no filtro é mantido constante, sendo os pontos mais antigos descartados com base em métricas internas de qualidade e tempo de seguimento.

### 3.4 Odometria Visual-Inercial-LIDAR

O trabalho apresentado por Zuo et al. (2019) propõe o algoritmo LIC-Fusion, uma abordagem de odometria *tightly-coupled* que integra diretamente dados de uma câmara monocular, um LIDAR 3D e uma IMU recorrendo a um filtro MSCKF (*Multi-State Constraint Kalman Filter*). Este, introduzido por Mourikis e Roumeliotis (2007), é uma variante do EKF que além do estado atual derivado das medições da IMU, mantém temporariamente no vetor de estado as estimativas da posição e orientação da câmara em instantes anteriores. Com isso, é possível utilizar as observações visuais de um mesmo ponto feitas em diferentes momentos para impor restrições entre as poses da câmara, sem ter de estimar explicitamente a posição 3D desses pontos. O MSCKF projeta essas restrições de forma eficiente, reduzindo a complexidade computacional para um valor linear no número de *features*, o que o torna adequado para aplicações em tempo real, mesmo quando as observações são intermitentes ou ruidosas.

O sistema realiza uma estimativa contínua de alta frequência com base na IMU, fundindo essas previsões com observações extraídas do LIDAR e da câmara, quando estas ocorrem. O processamento das nuvens de pontos LIDAR baseia-se na extração de *features* geométricas, de forma semelhante aos métodos previamente apresentados. Para acelerar a correspondência entre estas *features* em *scans* consecutivos, é utilizada uma estrutura de dados chamada árvore KD (*k-d tree*), que organiza os pontos num espaço hierárquico e permite buscas eficientes de vizinhos mais próximos. Relativamente à componente visual, são extraídas *features* utilizando o detetor FAST, que são posteriormente rastreadas entre *frames* consecutivos através do algoritmo de fluxo ótico KLT.

O trabalho proposto por Zheng, Q. Zhu et al. (2022) introduz o algoritmo

FAST-LIVO, um sistema de odometria LIDAR-Inercial-Visual (LIVO) com fusão sensorial *tightly-coupled*, baseado numa variante do *Error-State Extended Kalman Filter* (ESKF). A arquitetura integra dois subsistemas principais: um módulo de odometria LIDAR-Inercial (LIO), que regista diretamente a nuvem de pontos LIDAR num mapa tridimensional; e um módulo de odometria Visual-Inercial, que reutiliza esses mesmos pontos, associando-lhes informação visual. A correspondência entre os pontos da nuvem do LIDAR e os *pixels* da imagem é realizada através de uma projeção geométrica direta, com base na calibração extrínseca entre os sensores. Cada ponto 3D captado pelo LIDAR é inicialmente expresso no seu referencial próprio e transformado para o referencial global utilizando a pose estimada do sistema. Para projetar esse ponto no plano da imagem, o sistema converte-o do referencial global para o referencial da câmara, aplicando as transformações extrínsecas calibradas entre o LIDAR, a IMU e a câmara, bem como a pose global estimada no instante da imagem. Por fim, aplica-se o modelo de projeção da câmara para determinar as coordenadas do *pixel* correspondente na imagem. Esta correspondência permite associar ao ponto LIDAR pequenos blocos da imagem (*patches*) observados em *frames* anteriores, sendo armazenados juntamente com a pose da câmara no momento da captura. Quando uma nova imagem é adquirida, os pontos do mapa são novamente projetados para o plano da imagem atual e os respetivos *patches* associados são recuperados. O erro entre os valores de intensidade observados e os armazenados é então minimizado, permitindo estimar com precisão a nova pose do sistema. Para reforçar a robustez visual, o sistema integra ainda um mecanismo eficiente de rejeição de *outliers*, baseado na deteção de oclusões e descontinuidades de profundidade. As estimativas de pose são atualizadas iterativamente através do ESKF, onde são fundidas as medições inerciais com os resíduos de alinhamento geométrico (LIDAR) e da imagem. Derivado deste trabalho surge também o FAST-LIVO2 (Zheng, Xu et al. 2024).

## 3.5 Desafios Identificados

A revisão da literatura permitiu identificar um conjunto de desafios transversais às diferentes abordagens analisadas, que continuam a condicionar a sua aplicação em sistemas e cenários operacionais reais.

Em primeiro lugar, destaca-se a limitação imposta pela execução em tempo real em plataformas com recursos computacionais restritos. Apesar dos avanços alcançados por métodos como o LOAM e o ORB-SLAM (Campos et al. 2021), bem como pelas suas variantes otimizadas, as capacidades necessárias para que estes

algoritmos operem de forma fiável em tempo real ainda não são compatíveis com sistemas de baixo custo.

Outro desafio relevante diz respeito à integração prática destas soluções em sistemas com configurações e equipamentos distintos. A disponibilização de implementações de referência em código aberto, como o VINS-Mono ou o LIO-SAM, tem facilitado a validação em *datasets* padrão. Contudo, a sua adaptação a plataformas com diferentes sensores e arquiteturas permanece exigente, uma vez que requer calibrações rigorosas e ajustes específicos ao *hardware* disponível. Esta complexidade adicional pode, em certos casos, tornar a adaptação de soluções pré-existentes mais difícil do que o desenvolvimento de implementações concebidas de raiz para a configuração considerada.

A integração de múltiplos sensores constitui igualmente um aspeto determinante, uma vez que nenhum sensor isolado garante desempenho robusto em todas as condições operacionais. Embora as arquiteturas multi-sensoriais apresentem elevado potencial, a inclusão de novos sensores aumenta significativamente a complexidade do sistema. Esta realidade evidencia a importância de desenvolver soluções flexíveis e resilientes, justificando a metodologia apresentada no capítulo seguinte.

Conclui-se, assim, a revisão da literatura, na qual foram identificadas as abordagens mais consensuais, discutidos os seus princípios de funcionamento, analisadas as respetivas implementações e identificadas dificuldades transversais às demais. Esta síntese crítica permitiu evidenciar soluções tecnicamente consolidadas, reconhecer limitações recorrentes e destacar métodos emergentes, constituindo o enquadramento para a metodologia descrita no Capítulo 4.

# Capítulo 4

## Metodologia

O presente capítulo tem como objetivo caracterizar a plataforma robótica utilizada, detalhar a integração dos sensores empregues e apresentar de forma sistemática a metodologia adotada.

### 4.1 Leo Rover

O sistema utilizado para a condução da recolha da informação e validação do algoritmos foi o Leo Rover. Este UGV, desenhado para pesquisa e ensino foi adquirido pelo Centro de Investigação Naval (CINAV), como instrumento de apoio à investigação e desenvolvimento por parte da Marinha Portuguesa. O sistema, adquirido à empresa *fictionlab* está ilustrado na Figura 4.1.



FIGURA 4.1: Leo Rover  
Fonte: Fictionlab (2025)

### 4.1.1 Características Físicas e Arquitetura de Hardware

O Leo Rover é um UGV compacto, de quatro rodas com tração integral, arquitetura modular e controlo de direção por velocidade diferencial. Está preparado para operar em ambientes interiores e exteriores. A sua suspensão é passiva oscilante, baseada em braços basculantes solidários a cada par de rodas do mesmo lado, permitindo a adaptação do sistema às irregularidades do terreno. O corpo central do rover é constituído por uma estrutura em alumínio que suporta: uma caixa estanque, a qual aloja os principais componentes eletrónicos; a bateria e pontos de fixação para a instalação de sensores ou carga adicional. Tem dimensões de  $424 \times 445 \times 303$ mm, com um peso total de 7kg. Atinge velocidades lineares máximas de aproximadamente 0.4 m/s e velocidades angulares máximas de 1 rad/s. O sistema é capaz de transportar até 5kg de carga útil e transpor obstáculos com alturas até 70mm.

Cada uma das quatro rodas é equipada com motores DC, controlados por sinais *Pulse Width Modulation* (PWM), e *encoders*. Os pneus são em borracha, preenchidos com espuma, com diâmetro externo de aproximadamente 130mm e largura de 70mm. A alimentação do sistema é fornecida por uma bateria com capacidade de 5800mAh, permitindo um tempo de operação máximo estimado de até 4 horas.

A arquitetura de hardware é baseada em dois elementos principais: um microcomputador Raspberry Pi 5 e o LeoCore, um micro controlador desenvolvido especificamente para esta plataforma. O LeoCore agrega os controlos de baixo nível, incluindo os *drivers* dos motores, leitura dos encoders, aquisição dos dados da IMU, controlo de LEDs e controlo do fornecimento de energia ao Raspberry Pi. O Raspberry Pi, por sua vez, concentra todas as funções de alto nível do sistema, tais como, a execução do sistema operativo, do *middleware* ROS (*Robot Operating System*) e fornecer a interface para comunicação com sensores externos e com o utilizador.

O sistema inclui uma câmara Arducam de 12.3 MP com lente M12 de grande angular ( $158^\circ$  FOV horizontal), conectada ao Raspberry Pi. A conectividade do rover é assegurada por Wi-Fi dual-band (2.4 e 5 GHz) com uma antena que garante alcance de comunicação até 100m. Adicionalmente, o rover dispõe de uma porta USB tipo A, na parte superior da estrutura desenhada para a integração de sensores através da comunicação com o Raspberry Pi.

### 4.1.2 Arquitetura de Software

O sistema operativo principal do Leo Rover é uma distribuição personalizada baseada em *Ubuntu 20.04*, com suporte nativo ao ROS. A estrutura de software é baseada em uma arquitetura de nós ROS que comunicam entre si através de tópicos, facilitando a aquisição, processamento, controlo e visualização dos dados.

A interface de utilizador do Leo Rover é acessível através de um navegador web, permitindo o controlo remoto e a monitorização em tempo real do sistema. Na sua configuração base, esta interface disponibiliza a transmissão em direto da câmara, a leitura do nível de tensão da bateria, o controlo de movimento por teclado ou através de um *joystick* virtual, bem como botões para reinicializar e desligar o rover.

A Figura 4.2 apresenta a arquitetura dos *nodes* ou nós ROS, que compõem a funcionalidade padrão do sistema sem considerar a integração de sensores adicionais ou nós auxiliares. São também representados os tópicos utilizados e a forma como estes interagem com o *hardware*.

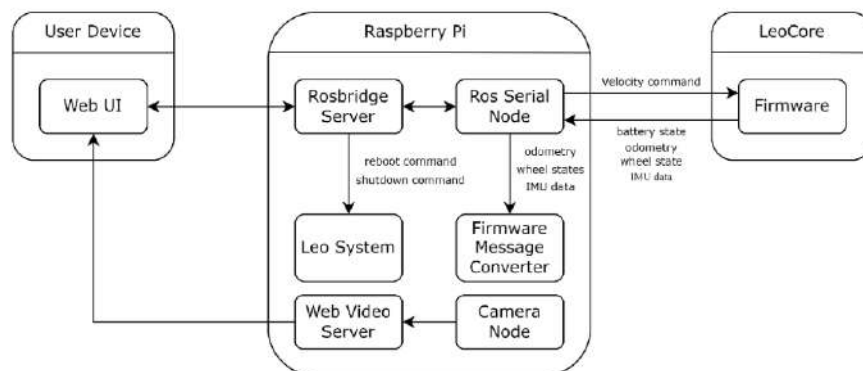


FIGURA 4.2: Estrutura ROS do Leo Rover

Fonte: Fictionlab (2025)

Todo o ecossistema do Leo Rover é *open-source*, com acesso público ao código-fonte do *firmware*, *software* ROS, esquemas eletrónicos e modelos CAD, sob licenças permissivas, facilitando a adaptação e extensão do sistema em contextos académicos.

## 4.2 Integração de Sensores no Sistema

A presente secção visa descrever a integração dos sensores no Leo Rover. O LIDAR foi incorporado por constituir uma das principais fontes de informação para o algoritmo de estimação de estado do veículo. O sensor GNSS e o sistema de *beacons Indoor Positioning System*(IPS), foram utilizados como referência de *ground-truth*, devido à sua elevada precisão, permitindo a posterior validação quantitativa das medições obtidas com o método desenvolvido. Todos os sensores foram ligados a um *hub* USB, conectada ao rover.

A presente secção descreve a integração dos sensores no Leo Rover. O LIDAR foi incorporado por constituir uma das principais fontes de informação para o algoritmo de estimação de estado do veículo. O sensor GNSS, para além de fornecer medições absolutas de posição que podem ser integradas no filtro, foi também utilizado como referência de *ground-truth*, em conjunto com o sistema de *beacons Indoor Positioning System* (IPS), devido à sua elevada precisão, permitindo a validação quantitativa das estimativas obtidas com o método desenvolvido. Todos os sensores foram ligados a um *hub* USB, conectado ao rover.

### 4.2.1 Integração do LIDAR

O LIDAR utilizado é um RPLIDAR A3, da SLAMTEC (Figura 4.3). Trata-se de um sensor LIDAR 2D, com alcance máximo de 25 m e mínimo de 0,2 m, adequado para operação em ambientes interiores e exteriores. Funciona através de um sistema de varrimento rotativo tendo assim um *Field of View* (FOV) de 360° no plano horizontal. Apresenta uma taxa de amostragem até 16 kHz, com frequência de varrimento ajustável entre 5 e 15 Hz. No contexto da dissertação, o sensor foi utilizado no modo de operação *standard*, com frequência de rotação de 10 Hz, taxa de amostragem de 16 kHz e resolução angular de 0,225°.



FIGURA 4.3: RPLIDAR-A3

Fonte: SLAMTEC (2025)

Para a fixação do sensor no *rover*, foi utilizado um suporte impresso em 3D, disponibilizado pela *fictionlab*<sup>1</sup>. O LIDAR foi montado na parte frontal do veículo, conforme destacado na Figura 4.4. Dado que o sensor foi rigidamente fixado à estrutura do veículo através do suporte fornecido, considera-se que o referencial do LIDAR está alinhado com o referencial do rover. De acordo com a convenção adotada, descrita no Anexo II, e considerando a forma como o sensor foi instalado, o eixo  $x$  do LIDAR corresponde ao eixo  $-y$  do rover, enquanto o eixo  $y$  do LIDAR corresponde ao eixo  $x$  do rover. Após esta correção de referencial, o sistema coincide com a convenção geral ROS.

A integração do sensor no ambiente ROS foi efetuada com base nas instruções fornecidas na documentação oficial<sup>2</sup>. Após a configuração, passaram a ser publicadas num tópico mensagens que incluem o *timestamp* da leitura, os ângulos inicial e final do varrimento (em radianos) e um vetor de distâncias, em que cada elemento representa a medição correspondente a um ângulo específico.

---

<sup>1</sup><https://docs.fictionlab.pl/leo-rover/integrations/addon-adapters>

<sup>2</sup><https://docs.fictionlab.pl/leo-rover/integrations/lidars/slamtec-rplidar-a2>



FIGURA 4.4: RPLIDAR-A3 Integrado

### 4.2.2 Integração do recetor GNSS

Como mencionado anteriormente, foi integrado um recetor de sinal GNSS com o objetivo integrar medições absolutas de posição e validar as trajetórias percorridas pelo rover em ambientes de teste exteriores. Nestes cenários, o sensor revelou desempenho adequado para a função de referência, permitindo comparar as estimativas obtidas com o método desenvolvido. No entanto, em ambientes interiores verificou-se a reduzida fiabilidade do recetor. Em medições de teste ocorridas no Departamento de Ciências e Tecnologias da Escola Naval, foi observada qualitativamente uma elevada variabilidade das posições reportadas, com desvios que podiam atingir dezenas de metros, tornando inviável a sua utilização como fonte primária de localização. O recetor utilizado foi um módulo UBlox EVK-M8N, ilustrado na Figura 4.5, que suporta múltiplos sistemas globais de navegação por satélite, incluindo GPS/SBAS, GLONASS, Galileo e BeiDou.



FIGURA 4.5: Módulo UBlox EVK-M8N

Fonte: UBlox EVK-M8N GPS Module (2025)

A integração no ambiente ROS foi realizada com base na documentação oficial<sup>3</sup>. Após a devida configuração, passou a estar disponível um tópico, no qual são publicadas mensagens, contendo informação como o *timestamp*, latitude, longitude, altitude e a respetiva matriz de covariância associada à medição.

As medições de latitude, longitude e altitude seguem a convenção do sistema geodésico WGS84 (*World Geodetic System 1984*), que define um elipsoide de referência global (Defense Mapping Agency 1987).

### 4.2.3 Integração do *Indoor Positioning System*

O sistema de posicionamento *indoor* da Marvelmind é um IPS (Indoor Positioning System) projetado para fornecer dados de localização em 2D ou 3D com elevada precisão, tipicamente da ordem dos  $\pm 2\text{cm}$ , de acordo com a respetiva *datasheet*. Funciona com base na medição do tempo de voo de sinais ultrassónicos emitidos e recebidos por *beacons*. O conjunto utilizado, ilustrado na Figura 4.6, é composto por cinco *beacons* com capacidade de transmissão e receção e um modem com suporte para até 250 elementos e 250 submapas.



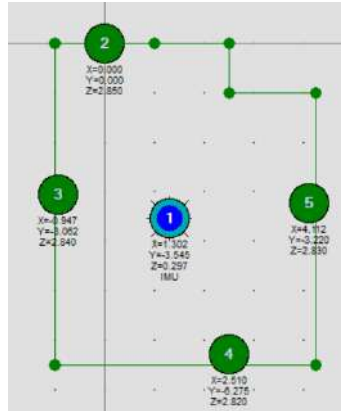
FIGURA 4.6: Marvelmind Starter Set Super-MP-3D

Fonte: Super-MP3D (2025)

De modo a obter coordenadas tridimensionais, é necessário instalar pelo menos quatro unidades em posições fixas, que funcionam como referências estacionárias. Estas permitem determinar, por trilateração, a posição da unidade móvel, designada *hedgehog*, acoplada ao objeto a localizar. A configuração do sistema é realizada através do *Dashboard* da Marvelmind, ilustrado na Figura 4.7, onde se executa a calibração inicial, se definem os referenciais de coordenadas e se introduz a altura de cada unidade fixa em relação ao solo, sendo as restantes posições relativas calculadas automaticamente pelo sistema.

---

<sup>3</sup><https://docs.fictionlab.pl/leo-rover/integrations/positioning-systems/ublox-evk-m8n>

FIGURA 4.7: Representação do espaço na aplicação *Dashboard*

Após esta calibração, o sistema publica autonomamente as estimativas de posição no ambiente ROS, sem necessidade de manter o *Dashboard* ativo. A integração foi realizada com base na documentação disponibilizada pela fictionlab<sup>4</sup>. As medições são publicadas num tópico, sob a forma de mensagens contendo as coordenadas  $x$ ,  $y$  e  $z$  no referencial previamente definido. Os *beacons* fixos foram instalados num laboratório do Departamento de Ciências e Tecnologias da Escola Naval. Todos os testes realizados em ambiente interior foram realizados nesse mesmo laboratório, com a configuração evidenciada na Figura 4.7.

A Figura 4.8 apresenta o Leo Rover na sua configuração final, com todos os sensores integrados. Estão destacados a vermelho o LIDAR, a verde o recetor GNSS e a azul o *hedgehog*.

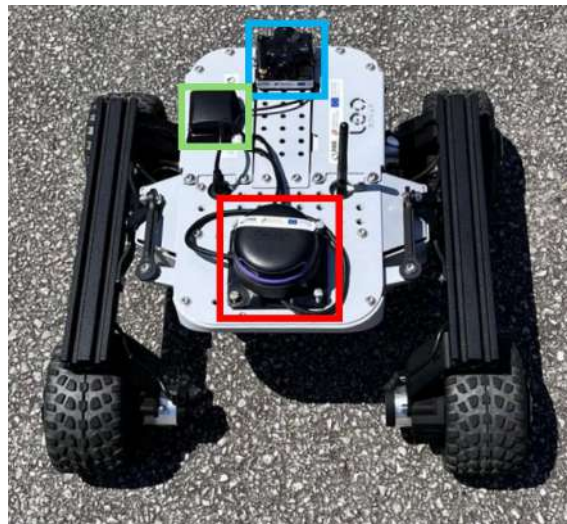


FIGURA 4.8: Leo Rover após integração dos sensores

<sup>4</sup><https://docs.fictionlab.pl/leo-rover/integrations/positioning-systems/marvelmind-rtlsn>

## 4.3 Abordagem

A presente secção descreve a abordagem adotada para a estimação de estado do sistema, fundamentada nos princípios teóricos apresentados no Capítulo 2 e nas soluções analisadas na revisão do estado da arte (Capítulo 3). Entre as várias arquiteturas possíveis, optou-se por uma abordagem de *late fusion*, ou *loosely-coupled*, que privilegia a modularidade e a integração flexível de sensores heterogêneos. Esta opção não se restringe ao objetivo imediato de assegurar a navegação do Leo Rover, mas visa também explorar metodologias generalizáveis a diversas plataformas, incluindo veículos não terrestres, como USVs. Neste enquadramento, o paradigma de *late fusion* revela-se particularmente adequado, uma vez que permite incorporar e substituir sensores, assegurando a adaptabilidade do sistema a distintos contextos operacionais. Esta motivação encontra-se alinhada com iniciativas internacionais como o projeto SET-309 da NATO, que procura desenvolver arquiteturas PNT (*Positioning, Navigation and Timing*) modulares e configuráveis, capazes de responder às exigências de cenários operacionais com disrupção de sinal GNSS.

Esta opção de arquitetura mostra-se particularmente adequada a plataformas terrestres não tripuladas (UGVs) de baixo custo, concebidas com carácter descartável e de fácil substituição. Neste enquadramento, a utilização de uma arquitetura de *late fusion* que dispensa a construção explícita de mapas do ambiente, como é requerido em abordagens clássicas de SLAM, privilegiando antes uma solução leve e eficiente, com menores exigências computacionais, mas ainda assim capaz de fornecer estimativas de estado com a precisão necessária.

A arquitetura proposta assenta num *Error-State Extended Kalman Filter*, na qual as medidas da IMU são integradas diretamente na fase de predição do filtro, constituindo a base da propagação temporal. Os restantes sensores são tratados como fontes auxiliares de correção, podendo ser adicionados ou substituídos de forma flexível, consoante os requisitos da missão e as características do ambiente.

A arquitetura do sistema é ilustrada na Figura 4.9, evidenciando a estrutura modular adotada e a lógica de funcionamento da abordagem *loosely-coupled*.

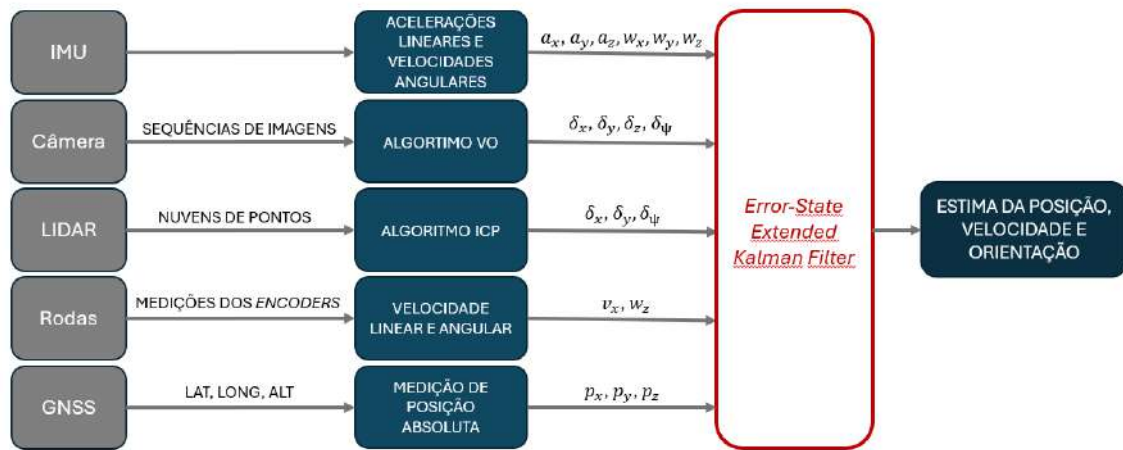


FIGURA 4.9: Esquema Representativo da Abordagem Utilizada

Com o objetivo de uniformizar a representação das medições e da pose estimada, foi adotado um referencial solidário ao corpo do veículo, ilustrado na Figura 4.10. Este sistema de coordenadas encontra-se centrado no centro de massa do UGV e segue a convenção definida na proposta REP 103 do *Robot Operating System* (ROS). De acordo com esta especificação, o eixo  $x$  aponta no sentido frontal do veículo, o eixo  $y$  no sentido lateral esquerdo e o eixo  $z$  no sentido vertical ascendente. As rotações são consideradas no sentido anti-horário em torno dos eixos  $x$  (*roll*,  $\phi$ ),  $y$  (*pitch*,  $\theta$ ) e  $z$  (*yaw*,  $\psi$ ). Todas as medições que alimentam o filtro de estado de erro são expressas neste referencial.

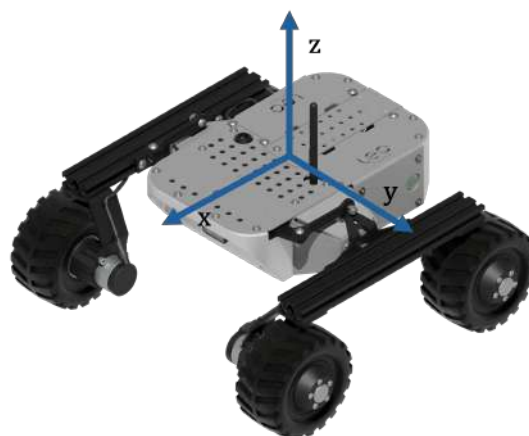


FIGURA 4.10: Referencial Leo Rover

O sistema é alimentado por medições provenientes de quatro sensores distintos: a IMU, a câmera do Leo Rover, o LIDAR integrado e os *encoders* das rodas

do veículo.

A IMU constitui o núcleo da propagação do estado, fornecendo diretamente ao filtro as medições de aceleração linear  $(a_x, a_y, a_z)$  e velocidade angular  $(w_x, w_y, w_z)$ , publicadas a uma frequência de aproximadamente 100 Hz.

A câmera fornece sequências de imagens que são processadas por um algoritmo de odometria visual (VO), responsável por estimar os incrementos de translação e rotação  $(\delta_x, \delta_y, \delta_z, \delta_\psi)$  entre pares de imagens consecutivos. De forma semelhante, o LiDAR gera nuvens de pontos que alimentam um algoritmo baseado em *Iterative Closest Point* (ICP), a partir do qual se obtêm incrementos espaciais  $(\delta_x, \delta_y, \delta_\psi)$ , no plano horizontal do veículo e do LIDAR, entre varrimentos sucessivos.

Em ambos os casos, considera-se apenas o incremento de rotação em torno do eixo vertical,  $\delta_\psi$ . Esta simplificação decorre das restrições cinemáticas do veículo, que se desloca predominantemente num plano e não realiza rotações significativas em torno dos eixos  $x$  ou  $y$ . Importa ainda referir que, por se tratar de um sensor LIDAR 2D, a estimativa de rotações noutros eixos se torna intrinsecamente limitada e mais complexa. Adicionalmente, verificou-se experimentalmente que a odometria visual fornece medições consideravelmente ruidosas nas componentes de rotação em torno dos eixos  $x$  e  $y$ , sendo, por isso, descartadas neste contexto.

Os incrementos são calculados relativamente a um referencial inicial assumido como  $(0, 0, 0)$ , sendo a rotação sempre interpretada em relação à orientação de inicial.

As medições dos *encoders* das rodas permitem calcular a velocidade linear  $v_x$  e a velocidade angular  $w_z$ , que são também incorporadas no ESKF como observações adicionais. Estes parâmetros são extraídos diretamente de um tópico ROS, o qual fornece a velocidade resultante com base na leitura individual de cada roda.

Por fim, as medições provenientes do recetor GNSS podem assumir um carácter permanente, intermitente ou omissivo na introdução de informação no filtro. Em ambientes com degradação de sinal GNSS, estas medições podem ser desconsideradas ou utilizadas de forma seletiva, dependendo do contexto operacional e da disponibilidade de sinal. As observações de latitude, longitude e altitude são fornecidas em relação ao elipsoide WGS 84, conforme referido anteriormente. Para que possam ser incorporadas no filtro como medições absolutas de posição, é necessário proceder à conversão para o referencial ROS. Nos cenários experimentais considerados nesta dissertação, esta conversão de referenciais foi realizada em pós-processamento.

Nas secções seguintes, procede-se à caracterização da IMU e detalham-se os algoritmos de estimação dos incrementos de pose com base na câmara e no LIDAR.

### 4.3.1 Caracterização da IMU

Com o objetivo de caracterizar os acelerómetros e os giroscópios utilizados, em particular os *bias* associados a cada medição, realizou-se um ensaio com o rover estacionário, registando as leituras dos sensores sem perturbações. Esta caracterização é relevante, uma vez que as medições da IMU contribuem diretamente para a propagação do estado do sistema. A Figura 4.11 apresenta os valores medidos em função do tempo.

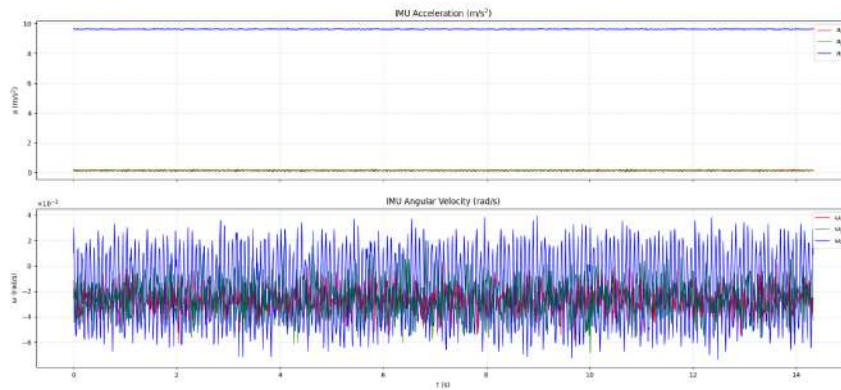


FIGURA 4.11: Medições com o rover estacionário

Relativamente aos acelerómetros, as médias das leituras foram de:  $0.131532 \text{ m/s}^2$  (eixo  $x$ ),  $0.129142 \text{ m/s}^2$  (eixo  $y$ ) e  $-0.186558 \text{ m/s}^2$  (eixo  $z$ ) com desvios padrão de  $0.041879$ ,  $0.031858$  e  $0.018181 \text{ m/s}^2$ , respetivamente. Para remover a componente gravítica aplicou-se a correção correspondente ao vetor da gravidade  $g = (0, 0, -9.80665) \text{ m/s}^2$ .

A estimativa precisa dos *bias* dos acelerómetros é mais exigente do que a do giroscópio, devido à presença constante da gravidade e à elevada sensibilidade a variações mínimas de orientação, incluindo inclinações residuais da IMU e pequenos desníveis do piso. Nestas condições, a aceleração gravítica projeta-se nos três eixos em função da atitude, o que dificulta dissociar o *bias* verdadeiro dos efeitos de inclinação. Uma calibração rigorosa requer condições controladas, como uma superfície perfeitamente nivelada, ou medições em múltiplas orientações conhecidas, de modo a isolar o contributo da gravidade e obter uma estimativa mais fiável do *bias*.

Relativamente aos giroscópios, os resultados foram substancialmente melhores, evidenciando maior exatidão. Os *bias* médios foram  $-0.002715$  rad/s no eixo  $x$ ,  $-0.002408$  rad/s no eixo  $y$  e  $-0.001761$  rad/s no eixo  $z$  com desvios padrão de  $0.000962$ ,  $0.001309$  e  $0.002810$  rad/s, respetivamente.

Estes valores foram usados para inicializar os respetivos termos no estado do filtro, facilitando a convergência inicial. Contudo, verificou-se que, entre os diferentes cenários de teste, apresentavam variações ligeiras, o que reforça a necessidade de incluir estes parâmetros no vetor de estado do sistema.

#### 4.3.2 Algoritmo VO

Para a implementação do algoritmo de odometria visual, foi adotada uma abordagem fundamentada na revisão do estado da arte, contemplando inicialmente três métodos distintos: o fluxo ótico de Kanade-Lucas-Tomasi (KLT), a combinação do detetor FAST com descritores ORB, e o *pipeline* completo do ORB, que atua simultaneamente como detetor e descritor. A escolha destes algoritmos baseou-se na sua ampla aceitação na comunidade científica e na sua eficácia comprovada em sistemas com restrições computacionais. No entanto, no decorrer do trabalho, foi utilizado exclusivamente o método baseado no fluxo ótico KLT, por ser o único a produzir resultados consistentes e fiáveis no contexto dos dados disponíveis.

O método tem a seguinte estrutura de processamento:

- Definição dos parâmetros intrínsecos da câmara;
- Deteção e descrição de *features*;
- Correspondência e seguimento de *features* entre imagens consecutivas;
- Estimativa da matriz essencial;
- Extração da rotação e translação entre *frames*;
- Estimativa da escala relativa;
- Transformação para o referencial do Leo Rover;
- Estimativa de  $(\delta_x, \delta_y, \delta_z, \delta_\psi)$

A abordagem foi implementada através de um *script Python*, executado em pós-processamento, com base em *rosbags* contendo imagens comprimidas captadas pela câmara do rover.

Os parâmetros intrínsecos da câmera, nomeadamente a matriz de calibração  $K$  e o vetor de distorção  $D$ , foram extraídos diretamente das mensagens publicadas num tópico ROS, de acordo com uma determinada tipologia de mensagem.

A matriz intrínseca  $K \in R^{3 \times 3}$  descreve a projeção de pontos tridimensionais, expressos no referencial da câmera, para coordenadas bidimensionais no plano da imagem, sendo geralmente expressa da seguinte forma:

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (4.1)$$

onde  $f_x, f_y$  representam as distâncias focais em *pixels* nos eixos horizontal e vertical, e  $c_x, c_y$  as coordenadas do centro ótico da imagem.

O vetor de distorção  $D \in R^5$ , utilizado para corrigir as deformações introduzidas pela lente, é definido por:

$$D = [k_1 \quad k_2 \quad p_1 \quad p_2 \quad k_3] \quad (4.2)$$

em que  $k_1, k_2, k_3$  são os coeficientes de distorção radial, e  $p_1, p_2$  os coeficientes de distorção tangencial. Estes parâmetros foram utilizados no processo de retificação das imagens, realizado com uma função da biblioteca OpenCV. Esta função conduz as operações necessárias para corrigir a distorção ótica da lente. As imagens foram extraídas do *rosbag*, convertidas para tons de cinzento e retificadas segundo o procedimento descrito. Posteriormente, foi aplicado o filtro *CLAHE* com o objetivo de melhorar o contraste.

Foram então ajustados os parâmetros dos processos de deteção, descrição e seguimento de *features*, de forma a garantir uma elevada robustez e estabilidade nas correspondências entre *frames*. A escolha dos parâmetros foi efetuada tendo em conta as características específicas de cada método. A partir das correspondências de *features* em imagens consecutivas, é então estimada a matriz essencial,  $E \in R^{3 \times 3}$ , implementada através de ferramentas da biblioteca OpenCV. Esta recorre também ao algoritmo RANSAC para garantir robustez face a correspondências erradas (*outliers*). O RANSAC permite identificar e rejeitar *features* que não respeitem a geometria epipolar, ajustando a matriz essencial com base num subconjunto consistente de correspondências válidas. A matriz  $E$  encapsula assim a relação geométrica entre dois pontos de vista sucessivos. A partir de  $E$ , é então extraída a

rotação  $R \in SO(3)$  e a translação  $t \in R^3$  entre os *frames*, através de outra função da mesma biblioteca.

Importa salientar que a translação e a rotação estimadas a partir da matriz essencial se encontram expressas numa escala relativa e arbitrária, sem correspondência direta com distâncias físicas no mundo real. Esta ambiguidade de escala, implica que as transformações obtidas estejam apenas corretas até um fator escalar, o qual depende da geometria do cenário observado e da configuração da câmera. Para superar esta limitação, torna-se necessário estimar um fator de escala com base em informação adicional.

Adaptando as abordagens propostas por Gräter et al. (2015), Ke e Kanade (2003), Song et al. (2015) e D. Zhou et al. (2019), optou-se por estimar o fator de escala assumindo que um subconjunto das *features* identificadas corresponde ao plano do solo. Esta aproximação baseia-se no princípio de que, sendo o veículo um UGV está por natureza defeito no solo e o FOV da câmera o contempla na maioria das situações. A identificação desse subconjunto permite ajustar geometricamente um plano 3D aos pontos e a partir da sua orientação e posição relativas, inferir uma medida de escala com base numa referência física conhecida.

Como etapa preliminar, foi definida uma região de interesse (ROI) na imagem, dentro da qual se assume que os pontos visíveis pertencem ao plano do solo. Esta região foi selecionada com base em critérios empíricos e está assinalada a vermelho na Figura 4.12.

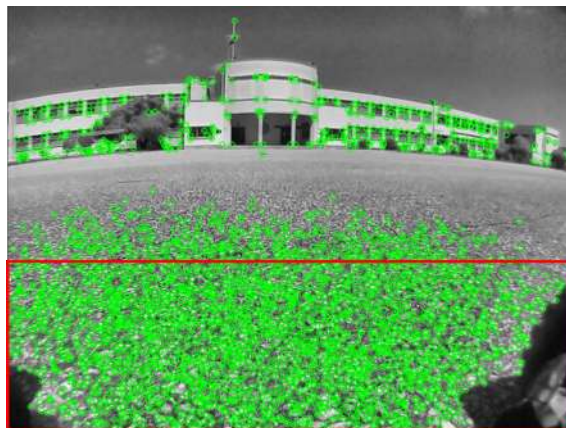


FIGURA 4.12: Região de interesse (ROI)

A ROI abrange aproximadamente os 40% inferiores da imagem, zona que além de captar uma maior densidade de pontos próximos da câmera, coincide frequentemente com a presença visível da roda dianteira do veículo. Uma vez que esta

roda está fisicamente em contacto com o solo, considera-se válida a assunção de que os pontos observados nessa região se distribuem sobre o plano do chão. Esta aproximação pode, naturalmente, ser afetada em cenários com inclinações acentuadas ou obstáculos elevados; no entanto, revelou-se robusta na generalidade dos contextos operacionais analisados. É importante ressaltar que os detetores raramente adquiriam *features* nesta mesma roda, considerando-se assim uma fonte de ruído desprezível. Caso tal se revelasse necessário, poderia ainda ser aplicada uma máscara à imagem de forma a remover a área correspondente à roda.

Com os pontos 3D selecionados nessa região, é realizado um ajuste a um plano recorrendo ao algoritmo RANSAC, que permite eliminar pontos discrepantes e encontrar a orientação média da superfície observada. O plano é modelado pela equação geral:

$$ax + by + cz + d = 0 \quad (4.3)$$

em que  $(a, b, c)$  definem o vetor normal unitário ao plano, e  $d \in \mathbb{R}$  representa o termo independente da equação. A estimação inicia-se com a reconstrução tridimensional de um conjunto de pontos correspondentes  $q_1 \leftrightarrow q_2$ , obtidos por triangulação linear, com recurso a uma biblioteca OpenCV, com base nas matrizes de projeção  $P_1$  e  $P_2$ :

$$P_1 = K \begin{bmatrix} I & 0 \end{bmatrix}, \quad P_2 = K \begin{bmatrix} R & t \end{bmatrix} \quad (4.4)$$

onde  $K \in \mathbb{R}^{3 \times 3}$  representa a matriz intrínseca da câmara, e  $R, t$  definem a transformação relativa entre poses sucessivas.

Para cada iteração do RANSAC, são selecionados aleatoriamente três pontos 3D,  $X_1, X_2, X_3$ , a partir dos quais se calcula o vetor normal ao plano:

$$n = \frac{(X_2 - X_1) \times (X_3 - X_1)}{\|(X_2 - X_1) \times (X_3 - X_1)\|} \quad (4.5)$$

e o termo independente  $d$  é obtido por substituição de um dos pontos na equação do plano:

$$d = -n \cdot X_1 \quad (4.6)$$

O conjunto de *inliers* corresponde aos pontos cuja distância ao plano é inferior a um limiar  $\varepsilon$ :

$$|n \cdot X_i + d| < \varepsilon \quad (4.7)$$

Sobre os pontos classificados como *inliers*, calcula-se o seu centróide:

$$\bar{X} = \frac{1}{N} \sum_{i=1}^N X_i \quad (4.8)$$

e ajusta-se um plano final por decomposição em valores singulares (SVD). A matriz de covariância dos pontos centralizados é decomposta, e o vetor normal é estimado como o vetor singular associado ao menor valor. A distância entre a câmera e o plano é finalmente estimada como:

$$d_{\text{plano}} = |n \cdot \bar{X} + d| \quad (4.9)$$

e a escala relativa entre o movimento estimado e o deslocamento real é corrigida por:

$$s = \frac{h}{d_{\text{plano}}} \quad (4.10)$$

sendo  $h$  a altura conhecida da câmera em relação ao solo.

Este fator é finalmente aplicado ao vetor de translação estimado entre *frames*, convertendo assim a trajetória estimada para o referencial físico do mundo.

A Figura 4.13 ilustra, de forma simplificada, os conceitos abordados na estimação da escala no algoritmo de odometria visual. O vetor a laranja representa a normal ao plano ajustado às *features* detetadas na imagem e projetadas no espaço tridimensional (círculos a azul), enquanto o vetor a preto corresponde à distância estimada  $d$ , medida entre a origem do referencial da câmera e o plano.

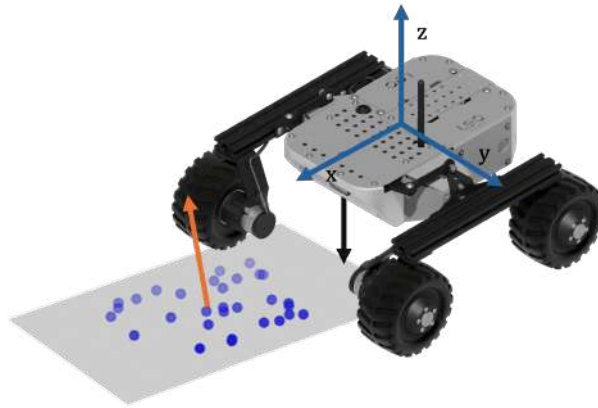


FIGURA 4.13: Representação geométrica da normal ao plano do solo e da distância  $d$  no referencial da câmera

A matriz de transformação final entre pares de imagens é então construída e convertida para o referencial do corpo do veículo, em conformidade com a convenção REP 103 do *Robot Operating System* (ROS), garantindo a coerência com as medições dos restantes sensores. Os incrementos de translação ( $\Delta x$ ,  $\Delta y$ ,  $\Delta z$ ) e rotação ( $\Delta\psi$ ) são extraídos diretamente da matriz de transformação estimada e armazenados com o respetivo *timestamp*, para posterior integração.

### 4.3.3 Algoritmo LIDAR

O algoritmo desenvolvido para o módulo de odometria LIDAR baseia-se no método *ICP* (*Iterative Closest Point*), seguindo os princípios introduzidos por J. Zhang, Singh et al. (2014), que recorre à extração de *features* geométricas a partir de nuvens de pontos bidimensionais para efetuar a estimativa da pose do sensor.

O processo inicia-se com a leitura sequencial das mensagens publicadas no tópico do sensor, extraídas de um ficheiro *rosvbag*. Cada *scan* fornece medições de distâncias  $r_i$  associadas a ângulos  $\theta_i$ , que são convertidas para coordenadas cartesianas no referencial do sensor (ilustrado no Anexo II) através da transformação:

$$x_i = r_i \cos(\theta_i), \quad y_i = r_i \sin(\theta_i) \quad (4.11)$$

Estas coordenadas representam a distribuição dos obstáculos detetados em torno do veículo, permitindo construir uma nuvem de pontos para cada instante de tempo. Estas medições não fornecem diretamente a posição do rover; a sua

trajetória é inferida pela comparação entre nuvens de pontos consecutivas, estimando o movimento relativo do sensor entre varrimentos sucessivos.

Para reduzir redundância e aumentar a eficiência computacional, é aplicado um processo de *downsampling* por grelha *voxel*, no qual o espaço 2D é discretizado em células de lado  $v$ , sendo os pontos contidos em cada célula substituídos pelo seu centróide. Esta operação preserva a estrutura da nuvem de pontos, ao mesmo tempo que reduz o número de pontos a processar (Rusu e Cousins 2011).

Feita esta aproximação, segue-se a extração de *features* com base na curvatura angular local  $\kappa_i$ , estimada para cada ponto  $p_i$  segundo:

$$\kappa_i = \left\| \sum_{j=i-w}^{i+w} p_j - (2w + 1) \cdot p_i \right\| \quad (4.12)$$

em que  $w$  representa o número de vizinhos considerados em cada lado de uma janela angular centrada no ponto. Esta métrica, inspirada na proposta original do algoritmo LOAM (3.1), constitui uma forma simplificada de avaliar a irregularidade geométrica local em 2D. Os pontos com valores elevados de  $\kappa_i$  são classificados como arestas ou *edges*, enquanto os de valor reduzido são considerados planos ou *flats*, conforme o critério detalhado no Capítulo 3.

Para cada um desses pontos de interesse, estima-se a normal local com o objetivo de caracterizar a orientação da superfície subjacente. Esta operação é realizada com base na Análise de Componentes Principais (PCA), aplicada à vizinhança euclidiana de cada ponto. A normal resultante corresponde ao vetor ortogonal ao primeiro vetor próprio da matriz de covariância dos vizinhos, conforme descrito por Hoppe et al. (1992). A inclusão destas normais permite que o alinhamento entre varrimentos tenha em conta não apenas a distância entre pontos, mas também a sua orientação geométrica local, aumentando a robustez da estimativa.

A estimação do movimento relativo entre varrimentos é realizada com um algoritmo ICP adaptado, que estabelece correspondências geométricas entre os pontos das nuvens sucessivas. A segmentação prévia em arestas e superfícies planas permite aplicar critérios diferenciados: ponto-para-linha nas *flats* e ponto-para-ponto nas *edges*.

Cada correspondência é ponderada através de uma função de custo robusta do tipo *Huber loss*, que atenua a influência de *outliers* e medições ruidosas (Huber 1992). Esta função, com parâmetro  $\delta$ , atribui pesos  $w$  aos resíduos  $r$  segundo:

$$w = \begin{cases} 1, & \text{se } \|r\| \leq \delta \\ \frac{\delta}{\|r\|}, & \text{caso contrário} \end{cases} \quad (4.13)$$

Com base nas correspondências ponderadas, é estimada a transformação relativa entre dois varrimentos consecutivos, representada por um vetor incremental  $\delta = (\delta_x, \delta_y, \delta_\theta)$ , correspondente à translação e rotação no plano, expressa no referencial do sensor LIDAR. Estes incrementos são posteriormente convertidos para o referencial do Leo Rover de modo a alimentar o ESKF.

## 4.4 Error-State Extended Kalman Filter

A presente secção descreve a implementação concreta do filtro *Error-State Extended Kalman Filter* (ESKF) adotado neste trabalho, com base na formulação teórica apresentada na Secção 2.5.3.

Tal como proposto por Sola (2017), o estado verdadeiro do sistema é decomposto em duas componentes: o estado nominal  $x$ , que descreve a trajetória estimada com base na integração das medições inerciais, e o estado de erro  $\delta x$ , que representa desvios de pequena magnitude em torno do estado nominal, não observados, correspondentes a ruído nos sensores e erros de modelação. A evolução destes estados é regida por duas fases distintas: a fase de predição, onde o estado nominal é propagado segundo o modelo cinemático do sistema e o estado de erro evolui segundo uma dinâmica linearizada; e a fase de correção, onde o vetor de erro é atualizado com base em observações externas e, posteriormente, injetado no estado nominal.

O vetor de estado nominal considerado nesta implementação é definido por:

$$x = [p \quad v \quad q \quad a_b \quad \omega_b]^\top \quad (4.14)$$

em que  $p \in \mathbb{R}^3$  representa a posição do sistema no referencial global,  $v \in \mathbb{R}^3$  a velocidade linear,  $q \in \text{SO}(3)$  o quaternião que codifica a orientação,  $a_b \in \mathbb{R}^3$  o *bias* do acelerómetro e  $\omega_b \in \mathbb{R}^3$  o *bias* do giroscópio. Seguindo a abordagem de Sola (2017), optou-se por não incluir explicitamente o vetor da gravidade no estado, assumindo-o constante e conhecido.

O estado de erro associado é dado por:

$$\delta x = [\delta p \quad \delta v \quad \delta \theta \quad \delta a_b \quad \delta \omega_b]^\top \quad (4.15)$$

onde  $\delta p, \delta v \in \mathbb{R}^3$  correspondem aos erros de posição e velocidade,  $\delta \theta \in \mathbb{R}^3$  ao erro de orientação, e  $\delta a_b, \delta \omega_b \in \mathbb{R}^3$  aos erros associados aos *bias* da IMU.

A IMU constitui a base da propagação do estado, fornecendo medições contínuas de acelerações lineares  $a_m$  e velocidades angulares  $\omega_m$ . Estas medições são utilizadas para atualizar posição, velocidade e orientação segundo as equações de evolução do estado nominal, não sendo portanto tratadas como observações externas de correção no ESKF.

As equações de atualização do vetor de estado nominal são:

$$p \leftarrow p + v \Delta t + \frac{1}{2} (R(a_m - a_b) + g) \Delta t^2 \quad (4.16a)$$

$$v \leftarrow v + (R(a_m - a_b) + g) \Delta t \quad (4.16b)$$

$$q \leftarrow q \otimes q((\omega_m - \omega_b) \Delta t) \quad (4.16c)$$

$$a_b \leftarrow a_b \quad (4.16d)$$

$$\omega_b \leftarrow \omega_b, \quad (4.16e)$$

onde  $a_m$  e  $\omega_m$  são as leituras da IMU, corrigidas pelos respetivos *bias*  $a_b$  e  $\omega_b$ . O termo  $R$  é a matriz de rotação obtida a partir do quaternião  $q$ , que permite transformar as acelerações do corpo para o referencial inercial, ao qual se adiciona a gravidade  $g$  para retirar a sua contribuição nas medições. A posição  $p$  e a velocidade  $v$  são então atualizadas com base nestas grandezas. A orientação é propagada pelo quaternião  $q$ , multiplicado pelo incremento associado à velocidade angular. Finalmente, os *bias* do acelerómetro e do giroscópio são considerados constantes ao longo do tempo, tal como expresso em (4.16d) e (4.16e).

De forma análoga, o estado de erro evolui segundo:

$$\delta p \leftarrow \delta p + \delta v \Delta t \quad (4.17a)$$

$$\delta v \leftarrow \delta v + \left( -R[a_m - a_b]_{\times} \delta \theta - R \delta a_b + \delta g \right) \Delta t + v_i \quad (4.17b)$$

$$\delta \theta \leftarrow R^{\top} \{ (\omega_m - \omega_b) \Delta t \} \delta \theta - \delta \omega_b \Delta t + \theta_i \quad (4.17c)$$

$$\delta a_b \leftarrow \delta a_b + a_i \quad (4.17d)$$

$$\delta \omega_b \leftarrow \delta \omega_b + \omega_i \quad (4.17e)$$

neste caso,  $\delta p, \delta v$  e  $\delta \theta$  representam os erros de posição, velocidade e orientação. Os termos  $\delta a_b$  e  $\delta \omega_b$  modelam os erros nos *bias* da IMU.

A fase de correção do filtro, seguindo a formulação teórica apresentada na Secção 2.5.3, baseia-se nas observações providenciadas pelos sensores integrados, nomeadamente as rodas, a odometria visual (VO), a odometria LIDAR e GNSS. Cada um destes sensores é modelado através de três componentes fundamentais:

- uma função de observação  $h(x)$ , que mapeia o vetor de estado nominal para o espaço das observações;
- uma matriz Jacobiana  $H = \frac{\partial h}{\partial \delta x}$ , que relaciona variações no estado de erro com a variação esperada nas observações.
- e uma matriz de covariância  $R$ , que modela a incerteza associada à respetiva medição;

As seguintes secções detalham a modelação de cada um dos sensores empregues.

#### 4.4.1 Rodas

Conforme definido anteriormente, as medições provenientes das rodas consistem na velocidade linear  $v_x$  e na velocidade angular  $\omega_z$ . Para que estas observações pudessem ser integradas no filtro, foram necessárias duas adaptações fundamentais.

Em primeiro lugar, assumiu-se explicitamente o veículo não-holonómico, o que implica que o movimento se dá apenas na direção longitudinal do corpo. Esta restrição é implementada definindo a velocidade do corpo no referencial do rover como:

$$\mathbf{v}_B = \begin{bmatrix} v_x \\ 0 \\ 0 \end{bmatrix} \quad (4.18)$$

Esta restrição contribui também para a estabilidade numérica do método, ao impor artificialmente que não existe movimento lateral nem vertical. Assim, mesmo na presença de medições incorretas nessas componentes, o filtro tende a corrigi-las de forma consistente, compensando erros provenientes da IMU ou de outros sensores. Esta abordagem é justificada pelo facto de, na maioria do tempo, o rover se deslocar exclusivamente no plano longitudinal.

A segunda adaptação resulta do facto de a velocidade angular não estar diretamente representada no vetor de estado. Consequentemente, não é possível comparar de forma direta a estimativa do filtro com a medição de velocidade angular fornecida pelas rodas. Para ultrapassar esta limitação, define-se o quaternião  $\tilde{q}$

correspondente à rotação incremental, obtido a partir do produto da velocidade angular medida pelo intervalo de tempo:

$$\tilde{q} = \omega_z \cdot \Delta t \quad (4.19)$$

O erro associado à orientação é, assim, dado pela diferença entre a orientação estimada pelo filtro e o quaterniã  $\tilde{q}$ . Desta forma, a observação da orientação é construída com base na rotação incremental estimada pelas medições de velocidade angular das rodas.

Assim, o vetor de medições da odometria pode ser representado como:

$$z_{\text{wheels}} \approx \begin{bmatrix} v_B \\ q \end{bmatrix} \quad (4.20)$$

onde  $v_B$  representa a velocidade no corpo e  $q$  o quaterniã de orientação resultante da integração da velocidade angular.

A matriz Jacobiana da função de observação das rodas, linearizada em torno do estado de erro, é dada por:

$$H_{\text{wheels}} = \begin{bmatrix} 0_{3 \times 3} & R^\top & \frac{\partial(R^\top v)}{\partial \delta \theta} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{4 \times 3} & 0_{4 \times 3} & \frac{1}{2} J_q & 0_{4 \times 3} & 0_{4 \times 3} \end{bmatrix} \quad (4.21)$$

Nesta expressão:

- O termo  $R^\top$  representa a matriz de rotação transposta, que converte a velocidade linear estimada no referencial inercial para o referencial do corpo. Esta rotação é necessária porque as medições da odometria referem-se à velocidade no corpo, enquanto o vetor de estado contém a velocidade expressa no mundo.
- A derivada  $\frac{\partial(R^\top v)}{\partial \delta \theta}$  corresponde ao desenvolvimento da rotação composta a partir do quaterniã, expressando a variação da velocidade observada no corpo em função de pequenos erros angulares. Este termo resulta da diferenciação da rotação em relação ao vetor de erro de orientação.
- O termo  $\frac{1}{2} J_q$  resulta da linearização do produto entre o quaterniã nominal  $q$  e um incremento infinitesimal de rotação representado por  $\delta q$ . Para incrementos

suficientemente pequenos, o quaternião incremental pode ser aproximado por:

$$\delta q \approx \left[ 1, \frac{1}{2} \delta \theta^\top \right]^\top$$

e, por conseguinte, a atualização do quaternião é dada por:

$$q \cdot \delta q \approx q + \frac{1}{2} J_q \delta \theta$$

onde  $J_q \in \mathbb{R}^{4 \times 3}$  é a jacobiana do quaternião em relação ao vetor de rotação infinitesimal definida, de acordo com Sola (2017), por:

$$J_q = \begin{bmatrix} -q_x & -q_y & -q_z \\ q_w & -q_z & q_y \\ q_z & q_w & -q_x \\ -q_y & q_x & q_w \end{bmatrix} \quad (4.22)$$

#### 4.4.2 LIDAR e VO

Com o intuito de garantir uma estrutura coerente e modular, que permita a fácil integração de diferentes sensores no sistema, adotou-se uma definição semelhante à das medições das rodas. Tanto o algoritmo de odometria LIDAR como o de odometria visual produzem vetores de incremento de pose entre instantes consecutivos. Considerando que cada transformação ocorre num intervalo de tempo  $\Delta t$ , a divisão desses incrementos pela duração do intervalo permite estimar as velocidades linear e angular correspondentes. Os deslocamentos nos eixos  $x, y, z$  são, assim, tratados como velocidades lineares, enquanto as variações de orientação, em particular a rotação em torno do eixo  $z$ , são interpretadas como velocidades angulares.

$$z_{\text{LIDAR}} = \begin{bmatrix} d_x \\ d_y \\ d_\psi \end{bmatrix} \cdot \Delta t \approx \begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix} \approx \begin{bmatrix} v_b \\ q \end{bmatrix} \quad z_{\text{VO}} = \begin{bmatrix} d_x \\ d_y \\ d_z \\ d_\psi \end{bmatrix} \cdot \Delta t \approx \begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_z \end{bmatrix} \approx \begin{bmatrix} v_b \\ q \end{bmatrix} \quad (4.23)$$

Em ambas as expressões, as velocidades lineares  $v_b$  estão expressas no referencial do corpo, e  $\omega_z$  corresponde à componente da velocidade angular em torno do eixo vertical. Desta forma, a matriz Jacobiana da função de observação associada

às medições provenientes do LIDAR e da odometria visual adota a mesma forma que a utilizada no modelo da odometria de rodas (Equação 4.21).

As matrizes de covariância associadas ao ruído de observação foram definidas de forma adaptativa, em função da qualidade das estimativas fornecidas por cada sensor. Em vez de se adotarem valores fixos, optou-se por uma modelação empírica, uma vez que a determinação exata das covariâncias através de métodos estatísticos ou analíticos é, na prática, complexa e fortemente dependente do contexto operacional.

Tanto no caso da odometria LIDAR como na odometria visual, foi implementado um modelo adaptativo em que a variância do ruído é ajustada dinamicamente em função da qualidade das *features* detetadas no processo de estimação. Em ambos os casos, a relação funcional entre a qualidade da medição e o ruído assumido foi definida empiricamente, com base em testes e observações experimentais.

No caso específico da odometria LIDAR, a precisão na estimação dos incrementos relativos de pose está diretamente relacionada com o número de características geométricas identificadas em cada varrimento, nomeadamente *edges* e *flats*. De forma geral, quanto maior o número de pontos bem estruturados, maior a fiabilidade da transformação estimada entre dois instantes consecutivos. Com base nesta observação, foi adotado o seguinte modelo para o cálculo da variância associada às medições:

$$N_{\text{total}} = N_{\text{edges}} + N_{\text{flats}} \quad (4.24)$$

$$\sigma = \frac{1}{\frac{1}{2}N_{\text{total}}^2 + 1} \quad (4.25)$$

A matriz de covariância  $R \in \mathbb{R}^{7 \times 7}$ , utilizada no modelo de observação, é então construída como uma matriz diagonal com variância uniforme:

$$R = \sigma^2 \cdot I_7 \quad (4.26)$$

onde  $I_7$  representa a matriz identidade de dimensão  $7 \times 7$ .

De forma análoga, no caso da odometria visual, a variância do ruído foi também parametrizada em função da qualidade das *features* utilizadas na estimação da pose relativa entre *frames*. Para esse efeito, consideraram-se dois fatores: o

número de correspondências utilizadas na estimação da matriz de transformação, e o número de *inliers* associados à estimação da escala relativa com base no plano do solo. Estes dois indicadores refletem diretamente a robustez e fidelidade da transformação estimada. Um maior número de *features* válidas na correspondência entre imagens tende a resultar em estimativas mais estáveis, enquanto um número elevado de pontos consistentes com o plano do solo sugere uma escala mais fidedigna.

Com base nestes critérios, definiu-se empiricamente o seguinte modelo para o cálculo da variância:

$$N_{\text{total}} = N_{\text{matches\_pose}} + N_{\text{inliers\_plane}} \quad (4.27)$$

$$\sigma = \frac{1}{2N_{\text{total}} + 1} \quad (4.28)$$

Tal como no caso anterior, a matriz de covariância  $R \in \mathbb{R}^{7 \times 7}$  foi modelada como uma matriz diagonal com variância uniforme:

$$R = \sigma^2 \cdot I_7 \quad (4.29)$$

Este tipo de modelação adaptativa revelou-se particularmente vantajoso tanto para a odometria visual como para a odometria LIDAR, permitindo ao filtro ajustar automaticamente o grau de confiança atribuído a cada correção em função da qualidade da informação disponível.

### 4.4.3 GNSS

As medições GNSS, quando integradas no sistema, são tratadas como observações absolutas de posição no referencial local  $\{ENU\}$  (East–North–Up). Este referencial é definido em torno de uma referência inicial correspondente à primeira medição válida  $(\varphi_0, \lambda_0, h_0)$ , em que  $\varphi$  representa a latitude,  $\lambda$  a longitude e  $h$  a altitude. A cada instante, a posição geodésica  $(\varphi, \lambda, h)$  fornecida pelo recetor é convertida para coordenadas locais  $(p_x^{ENU}, p_y^{ENU}, p_z^{ENU})$  através da transformação implementada pela função *geodetic2enu* da biblioteca *Python*.

O vetor de medições GNSS pode, assim, ser expresso como:

$$z_{\text{GNSS}} = \begin{bmatrix} p_x^{\text{ENU}} \\ p_y^{\text{ENU}} \\ p_z^{\text{ENU}} \end{bmatrix}, \quad (4.30)$$

sendo esta grandeza diretamente comparável com a posição estimada no mesmo referencial.

A matriz Jacobiana da função de observação, linearizada em torno do estado nominal, incide apenas sobre as componentes de posição:

$$H_{\text{GNSS}} = \begin{bmatrix} I_3 & 0_{3 \times 3} & 0_{3 \times 3} & \cdots \end{bmatrix}. \quad (4.31)$$

A matriz de covariância associada ao ruído de medição é definida como:

$$R = \text{diag}(\sigma_x^2, \sigma_y^2, \sigma_z^2), \quad (4.32)$$

podendo ser construída a partir das variâncias fornecidas pela informação pelo receptor GNSS ou, em alternativa, modelada empiricamente em função do contexto operacional.

Concluída a definição dos modelos de observação, bem como das respetivas matrizes de covariância para cada sensor, procede-se à fase de correção do filtro. Esta fase tem como objetivo ajustar o estado estimado com base nas medições externas, compensando os desvios acumulados durante a propagação.

A correção do estado de erro decorre em três etapas principais: a atualização do vetor de erro com base na inovação entre a medição e a observação prevista, a injeção desse erro no estado nominal e o reinício do vetor de erro e atualização da respetiva covariância.

A correção inicia-se com o cálculo do ganho de Kalman, que pondera a confiança relativa entre a estimativa atual e a nova medição:

$$K = PH^\top (HPH^\top + R)^{-1} \quad (4.33)$$

Com o ganho de Kalman calculado, procede-se à atualização do vetor de erro com base na inovação, definida pela diferença entre a medição real  $z$  e a observação prevista  $h(x)$ :

$$\delta \hat{x} \leftarrow K \cdot (z - h(x)) \quad (4.34)$$

Este vetor de erro estimado é então injetado no estado nominal, corrigindo diretamente os seus componentes:

$$p \leftarrow p + \delta \hat{p} \quad (4.35a)$$

$$v \leftarrow v + \delta \hat{v} \quad (4.35b)$$

$$q \leftarrow q \otimes q \{ \delta \hat{\theta} \} \quad (4.35c)$$

$$a_b \leftarrow a_b + \delta \hat{a}_b \quad (4.35d)$$

$$\omega_b \leftarrow \omega_b + \delta \hat{\omega}_b \quad (4.35e)$$

Por fim, a matriz de covariância do erro é atualizada para refletir a nova incerteza associada ao sistema após a correção:

$$P \leftarrow (I - KH)P, \quad (4.36)$$

e o vetor de erro é reinicializado a zero. Em implementações mais completas, esta etapa pode ser acompanhada por uma reparametrização do erro e da sua covariância através de uma matriz Jacobiana adicional  $G$ , que assegura a consistência estatística após a operação de injeção. No presente trabalho, e na maioria das aplicações, tal como referido por Sola (2017), assume-se  $G = I$ , dispensando tal atualização explícita.

Esta sequência completa o ciclo de correção do ESKF.

Na secção seguinte, apresentam-se os testes realizados e os resultados obtidos, com uma análise detalhada e crítica da implementação desenvolvida.

# Capítulo 5

## Análise de Resultados

Este capítulo tem como objetivo apresentar e analisar criticamente os resultados experimentais obtidos no âmbito da presente dissertação. Para tal, são introduzidos os diferentes cenários de teste considerados, bem como os dados recolhidos e as métricas utilizadas para avaliação do desempenho.

A Secção 5.1 descreve as métricas adotadas para quantificar a precisão e consistência das estimativas de trajetória. Seguidamente, a Secção 5.2 apresenta em detalhe os diversos cenários experimentais, concebidos com o intuito de avaliar, de forma sistemática, as vantagens e limitações dos métodos propostos. Por fim, a Secção 5.3 discute aspetos complementares e transversais aos vários ensaios, relevantes para a correta interpretação e contextualização dos resultados.

### 5.1 Métricas Utilizadas

De modo a avaliar quantitativamente a precisão das estimativas, foi utilizada a ferramenta EVO (Grupp 2017). Esta ferramenta permite comparar trajetórias estimadas com trajetórias de referência (*ground truth*), recorrendo a métricas estatísticas.

As três métricas principais utilizadas para avaliar o desempenho dos métodos de estimação de estado foram as seguintes:

- **Erro Absoluto de Posição (APE — Absolute Pose Error)**: quantifica a discrepância entre cada ponto da trajetória estimada e o respetivo ponto da trajetória de referência. Para cada instante de tempo  $i$ , o APE é definido como a norma euclidiana da diferença entre a posição real  $p_i$  e a posição estimada  $\hat{p}_i$ :

$$\text{APE}_i = \|p_i - \hat{p}_i\| \quad (5.1)$$

- **Root Mean Square Error (RMSE) do APE:** fornece uma medida agregada da precisão global da trajetória estimada, sendo particularmente útil por penalizar com maior severidade os desvios mais significativos. Ao elevar os erros ao quadrado antes de calcular a média, o RMSE amplifica o impacto de erros pontuais elevados, permitindo detetar episódios localizados de maior imprecisão. Esta métrica é, por isso, valiosa para complementar a análise do APE médio, fornecendo uma perspectiva mais sensível à variabilidade do erro ao longo da trajetória:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N \|p_i - \hat{p}_i\|^2} \quad (5.2)$$

onde  $N$  representa o número total de amostras temporais consideradas.

- **Erro Relativo de Posição (RPE — Relative Pose Error):** avalia o erro na estimativa dos incrementos relativos entre pares de poses separados por um intervalo fixo de tempo. Esta métrica é particularmente sensível à consistência local do movimento e permite detetar deriva acumulada e erros temporais. Ao contrário do APE, o RPE é independente do alinhamento global da trajetória. A sua forma é dada por:

$$\text{RPE}_i = \|(p_{i+\Delta} - p_i) - (\hat{p}_{i+\Delta} - \hat{p}_i)\| \quad (5.3)$$

Ao longo da análise realizada, sempre que são referidas as métricas APE e RPE, estas correspondem ao valor médio calculado considerando todos os pontos da trajetória estimada.

As trajetórias de referência utilizadas na avaliação dos métodos de estimação foram obtidas por duas abordagens distintas, consoante o ambiente de ensaio.

Em contexto interior, foi utilizado o sistema de localização Marvelmind, que fornece a posição do módulo móvel (*hedgehog*) com base na configuração dos *beacons* ilustrada na Figura 4.7. Como o centro do referencial deste sistema está definido no *beacon* fixo número 2, o alinhamento com o referencial do rover foi realizado subtraindo a medição inicial a todas as subseqüentes, assumindo essa posição como origem. A correspondência entre eixos foi estabelecida com base na orientação inicial do rover, que era sempre colocado alinhado com o *beacon* 2. Assim, o eixo  $y$  dos *beacons* corresponde ao eixo  $x$  do rover, e o eixo  $x$  dos *beacons* corresponde ao eixo  $y$  do rover.

Nos ensaios em ambiente exterior, a trajetória de referência foi obtida a partir das medições fornecidas pelo sistema GNSS previamente descrito, com uma frequência de aquisição de aproximadamente 4 Hz. Dada a elevada exatidão das posições registradas, a trajetória foi construída a partir da sua projeção sequencial. Para garantir a coerência entre o referencial do GNSS e o referencial do corpo do rover, foi aplicado um processo de alinhamento em pós-processamento. Este consistiu na seleção de um conjunto de pontos consecutivos no início do trajeto, onde o movimento é linear e a influência da deriva é desprezável. Através do ajuste de uma reta a esses pontos, foi identificado o vetor de direção dominante, que foi posteriormente alinhado com o eixo longitudinal  $x$  do rover. Este procedimento foi adotado sistematicamente em todos os ensaios realizados no exterior.

## 5.2 Descrição dos Cenários Experimentais

Com o intuito de validar a adaptabilidade e robustez do sistema desenvolvido em condições operacionais distintas, foram realizados cinco ensaios experimentais. Cada trajeto foi selecionado com o propósito de enfatizar características particulares do ambiente e, conseqüentemente, colocar à prova o método desenvolvido:

1. Trajeto retilíneo em ambiente interior estruturado;
2. Trajeto com padrão tipo “corta-relvas” em ambiente interior;
3. Trajeto retilíneo curto em ambiente exterior aberto, sem estruturas envolventes;
4. Trajeto retangular longo em ambiente exterior aberto, para avaliar a estabilidade do sistema ao longo do tempo;
5. Trajeto retilíneo em ambiente exterior com vegetação densa e zonas de sombra;

### 5.2.1 Trajeto 1

O primeiro ensaio foi realizado num ambiente interior, no laboratório de Eletrônica do Departamento de Ciências e Tecnologias da Escola Naval, conforme ilustrado na Figura 5.1. Este cenário representa um contexto típico onde a recepção de sinal GNSS é inexistente ou severamente degradada. Devido à regularidade do pavimento e à estabilidade estrutural do ambiente, era expectável um bom desempenho por parte dos métodos de odometria baseados em LIDAR e rodas, tornando este

um cenário de validação adequado para ambientes *indoor*. Na imagem, a localização de dois dos *beacons* fixos encontram-se destacadas a amarelo.



FIGURA 5.1: Cenário experimental do Trajeto 1

A trajetória percorrida teve aproximadamente 4 m de extensão, sendo maioritariamente retilínea, com uma ligeira correção a meio do percurso motivada por uma tendência de deriva lateral para a esquerda observada no movimento do rover.

Numa primeira fase, foi realizada a estimação da pose do veículo com cada um dos algoritmos de forma independente. A Figura 5.2 apresenta as trajetórias resultantes: o gráfico à esquerda representa a evolução no plano  $XY$ , enquanto o da direita mostra o plano  $XZ$ . A trajetória de referência, obtida a partir das medições dos *beacons*, é representada a vermelho. As trajetórias estimadas pela odometria visual e pela odometria LIDAR surgem a azul e verde, respetivamente, resultando da soma incremental dos incrementos de pose obtidos pelos algoritmos. A trajetória a amarelo corresponde à integração direta das velocidades linear e angular medidas pelas rodas.

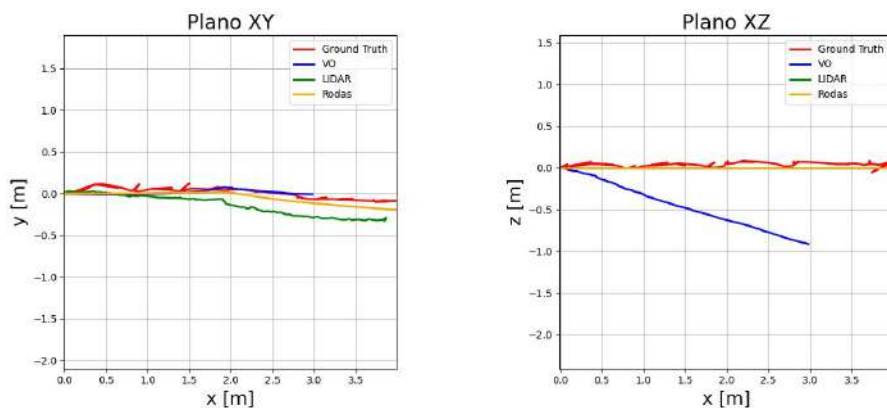


FIGURA 5.2: Trajeto 1 — Estimativas por métodos isolados

Embora a trajetória de referência apresente ruído próprio das medições dos *beacons*, esta constitui a melhor referência externa disponível para comparação.

A análise das trajetórias estimadas revela que a odometria nas rodas apresenta um desempenho bastante robusto, com boa aproximação à trajetória de referência. De forma semelhante, a estimativa gerada a partir dos incrementos de pose do LIDAR demonstra uma precisão aceitável, beneficiando da riqueza estrutural do ambiente e da boa definição das superfícies refletoras. Por outro lado, a odometria visual registou o pior desempenho entre os métodos isolados, fortemente afetada pelas condições de iluminação, que comprometeram a detecção e o seguimento estável de *features* visuais ao longo do percurso.

A Tabela 5.1 apresenta os resultados quantitativos obtidos para os métodos isolados. A odometria nas rodas (ODO) apresenta o menor erro absoluto (APE = 0.254 m) e o menor erro quadrático médio (RMSE = 0.271 m), refletindo um desempenho robusto. O método baseado em LIDAR apresenta um APE ligeiramente superior (0.304 m), mas um RMSE idêntico (0.304 m), o que demonstra boa consistência da estimativa. Já a odometria visual registra os maiores valores de erro, tanto em termos de APE (0.385 m) como de RMSE (0.440 m), com maior variância, evidenciando maior instabilidade ao longo do percurso, observável especialmente pelo erro acumulado em  $z$ .

TABELA 5.1: Trajeto 1 — Avaliação dos Métodos Isolados

Método	APE[m]	RMSE	RPE[m]	APE[std]	RPE[std]
ODO	<b>0.254</b>	<b>0.284</b>	<b>0.049</b>	<b>0.164</b>	0.046
LIDAR	0.304	0.342	0.050	0.167	<b>0.044</b>
VO	0.385	0.439	0.050	0.188	<b>0.044</b>

No que diz respeito ao erro relativo (RPE), todos os métodos apresentam desempenhos muito próximos, com RMSE entre 0.049 m e 0.050 m, o que indica que as estimativas de movimento entre poses consecutivas são localmente consistentes, mesmo quando o erro absoluto diverge ao longo do tempo.

Conclui-se, assim, que apesar da VO poder apresentar valores absolutos competitivos após alinhamento global, os métodos baseados nas medições das rodas e LIDAR oferecem trajetórias mais regulares, estáveis e alinhadas com a geometria real do ambiente.

Na fase seguinte, avaliou-se o desempenho da fusão sensorial com recurso ao filtro ESKF desenvolvido. A Figura 5.3 ilustra o resultado obtido com o filtro alimentado apenas pela IMU. Observa-se uma divergência rápida face à trajetória real, consequência da acumulação dos erros inerentes à integração contínua das medições da IMU sem qualquer mecanismo de correção.

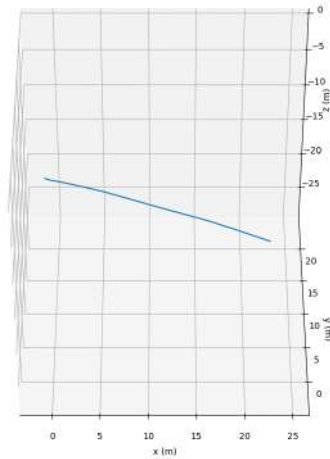


FIGURA 5.3: Trajeto 1 — Estimativa apenas com a IMU

Este resultado reforça a necessidade de incorporar medições adicionais durante a fase de correção do filtro, de modo a mitigar a deriva e assegurar uma estimativa fiável da pose. No Anexo III são apresentadas as trajetórias resultantes da fusão entre a IMU com cada um dos sensores (LIDAR, VO e rodas).

TABELA 5.2: Trajeto 1 — Avaliação dos Métodos Compostos com ESKF

Método	APE[m]	RMSE	RPE[m]	APE[std]	RPE[std]
IMU + VO	0.438	0.523	0.065	0.188	0.058
IMU + ODO	0.285	0.323	<b>0.048</b>	0.161	0.044
IMU + LIDAR	0.300	0.346	<b>0.048</b>	0.164	<b>0.043</b>
IMU + ODO + LIDAR	<b>0.278</b>	<b>0.317</b>	<b>0.048</b>	<b>0.160</b>	0.044
TOTAL	0.280	0.320	<b>0.048</b>	<b>0.160</b>	0.044

A Tabela 5.2 apresenta os resultados da avaliação da estimativa da posição com recurso ao filtro ESKF. O melhor desempenho foi obtido com a combinação IMU+ODO+LIDAR, que registou o menor erro absoluto de posição (APE =

0.278 m) e o menor erro quadrático médio (RMSE = 0.317 m), mantendo uma consistência local elevada (RPE = 0.048 m). Este resultado evidencia a complementaridade das medições: a estabilidade das rodas em ambientes planos e a robustez geométrica do LIDAR contribuíram para uma estima mais precisa e fiável.

As fusões IMU+ODO (APE = 0.285 m, RMSE = 0.323 m) e IMU+LIDAR (APE = 0.300 m, RMSE = 0.346 m) também apresentaram desempenhos satisfatórios, com valores de erro relativo idênticos (RPE = 0.048 m), refletindo trajetórias regulares e localmente coerentes.

Em contraste, a fusão entre IMU e VO apresentou o pior desempenho entre os métodos compostos, com APE = 0.438 m, RMSE = 0.523 m, e um erro relativo mais elevado (RPE = 0.065 m). Este resultado deve-se à fraca iluminação e consequente escassez de *features* visuais, que comprometeram a qualidade das medições e degradaram a estimativa da pose.

Por fim, a combinação de todos os sensores (TOTAL) não superou a fusão IMU+ODO+LIDAR, registrando um ligeiro aumento do erro absoluto (APE = 0.280 m) e do erro quadrático médio (RMSE = 0.320 m). Esta degradação é atribuída ao ruído introduzido pelas medições visuais, cuja incerteza elevada, embora ponderada no filtro, continua a afetar negativamente a estimativa final.

A Figura 5.4 apresenta a comparação entre a melhor estimativa isolada, a melhor trajetória obtida através do filtro e a trajetória de referência. Embora do ponto de vista numérico a integração das velocidades das rodas apresente o menor valor de APE e RMSE, observa-se que a trajetória estimada pelo filtro, representada a azul, acompanha de forma mais coerente a trajetória de referência.

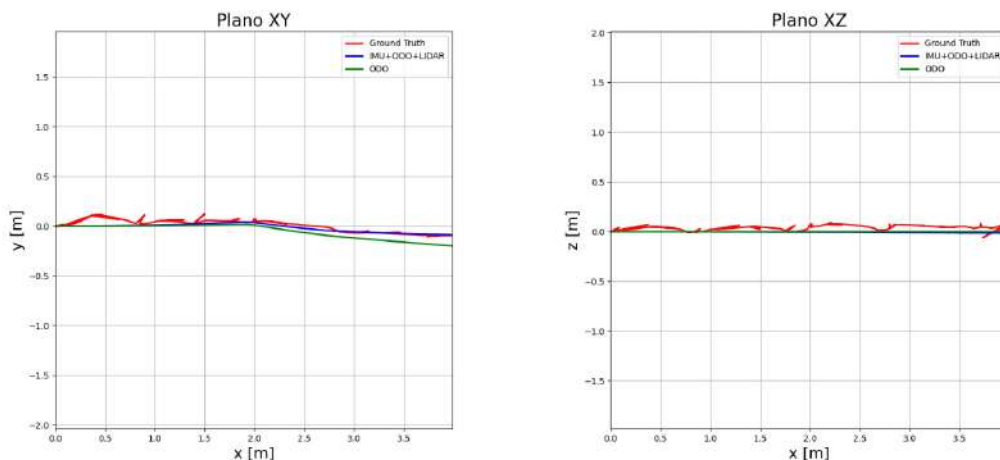


FIGURA 5.4: Trajeto 1 — Comparação entre as melhores estimativas

## 5.2.2 Trajeto 2

O segundo trajeto consistiu num padrão tipo “corta-relvas” de curta extensão, realizado em ambiente interior. Caracterizou-se por sucessivas mudanças de direção, com rotações frequentes e trajetos curtos entre cada inversão. Este tipo de configuração foi projetado para testar a robustez dos algoritmos em transições rápidas e avaliar a estabilidade do sistema em ambientes confinados e estruturados.

Inicialmente foi então realizada a estimação da trajetória pelos métodos isolados. No Anexo III, na Figura III.2 estão ilustradas as estimativas de posição para cada um dos métodos independentemente.

A Tabela 5.3 apresenta os resultados obtidos para os métodos isolados. Verifica-se que a odometria baseada em rodas apresenta o menor erro absoluto (APE = 0.741 m) e o menor erro quadrático médio (RMSE = 0.741 m), com um desvio padrão reduzido (std = 0.341 m), refletindo um desempenho consistente ao longo da trajetória. O método LIDAR regista um erro ligeiramente superior (APE = 0.813 m, RMSE = 0.780 m), embora com desvio padrão semelhante (std = 0.357 m), beneficiando da estrutura do ambiente.

Em contraste, a odometria visual apresenta erros significativamente mais elevados, com um APE de 9.150 m e um RMSE de 10.801 m, acompanhados por um desvio padrão de 3.505 m, evidenciando instabilidade ao longo do percurso. Este desempenho inferior deve-se à sensibilidade da VO às condições visuais do ambiente, nomeadamente à iluminação reduzida e à escassez de *features* detetadas.

No que diz respeito ao erro relativo (RPE), os métodos baseados em rodas e LIDAR mantêm consistência local razoável, com valores médios próximos de 0.025 m. No entanto, a VO apresenta um erro relativo elevado (RPE = 0.555 m), refletindo a acumulação de erro ao longo do tempo

TABELA 5.3: Trajeto 2 — Avaliação dos Métodos Isolados

Método	APE[m]	RMSE	RPE[m]	APE[std]	RPE[std]
ODO	<b>0.741</b>	<b>0.741</b>	<b>0.025</b>	<b>0.341</b>	<b>0.043</b>
LIDAR	0.813	0.780	0.026	0.357	<b>0.043</b>
VO	9.150	10.801	0.555	3.505	0.551

Conclui-se que, embora a VO consiga por vezes estimativas razoáveis a curto prazo, o método apresenta erros cumulativos que conduzem a estimativas

muito enviesadas para trajetos mais longos e com alterações de orientação. Em contrapartida, os métodos baseados em LIDAR e rodas demonstram maior robustez e consistência, sendo mais adequados para ambientes interiores.

Após a análise dos métodos isolados, avaliou-se a fusão dos dados da IMU com cada sensor adicional, considerando diferentes configurações do filtro ESKF. No caso da integração com o LIDAR, verificou-se que o desempenho melhora quando se utilizam apenas os incrementos de posição em  $x$  (convertidos em velocidade  $v_x$ ) e de orientação  $\psi$  (interpretados como velocidade angular  $w$ ) como medições. Esta configuração reduz o impacto de ruído presente noutras componentes, nomeadamente na deriva lateral, aumentando a consistência da estimativa.

A Figura 5.5 compara as trajetórias resultantes de duas variantes da fusão IMU+LIDAR: uma com todas as componentes de velocidade ( $v_x$ ,  $v_y$  e  $w$ ), designada por “LIDAR-TOTAL”, e outra apenas com  $v_x$  e  $w$ , identificada como “LIDAR-ADAPTADO”. Os valores de APE obtidos confirmam a vantagem desta última: o método “LIDAR-ADAPTADO” apresentou um erro absoluto médio de 0.655 m, inferior ao valor de 0.769 m registado pela configuração “LIDAR-TOTAL”. Perante estes resultados, adotou-se a configuração “LIDAR-ADAPTADO” na integração das medições do sensor no filtro ESKF.

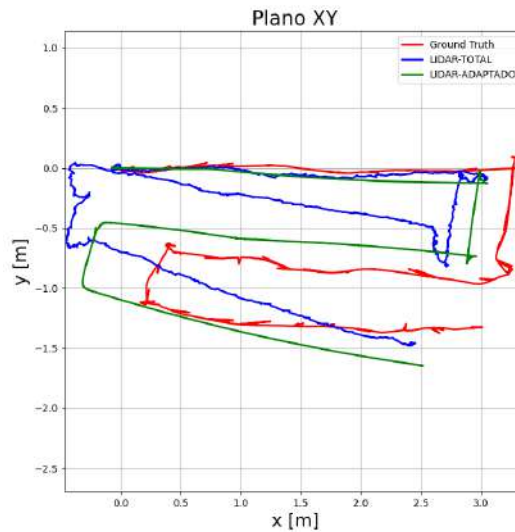


FIGURA 5.5: Trajeto 2 — Comparação entre variantes da fusão IMU+LIDAR

De forma a avaliar o impacto da integração de diferentes sensores no desempenho do filtro, procedeu-se à comparação das várias combinações consideradas. A

Figura 5.6 apresenta as estimativas resultantes da aplicação do filtro ESKF com as diferentes configurações de sensores no Trajeto 2.

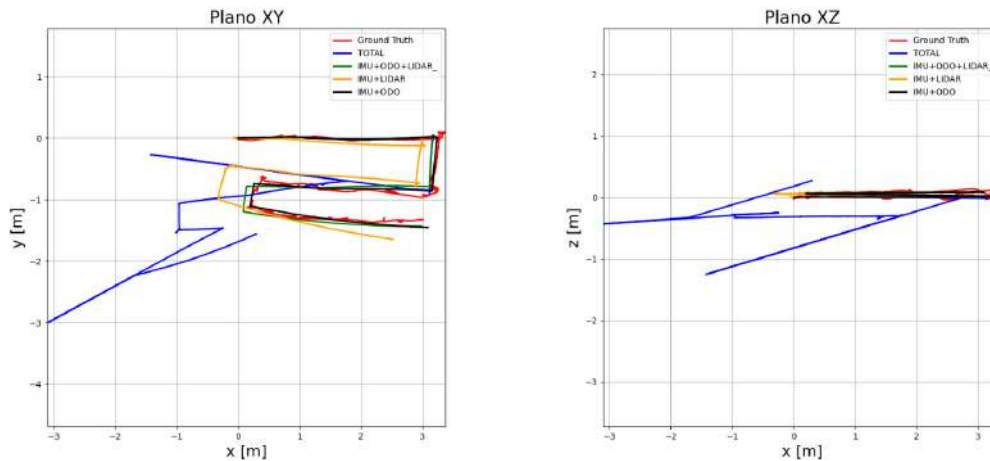


FIGURA 5.6: Trajeto 2 — Estimativas com ESKF

Entre as combinações avaliadas, a integração IMU+ODO demonstrou o melhor desempenho, apresentando uma trajetória estimada estável e bem alinhada com o percurso de referência. A fusão com o LIDAR revelou resultados ligeiramente inferiores, atribuídos à menor precisão na estimativa do ângulo de orientação  $\psi$ , uma limitação identificada dos algoritmos de odometria LIDAR. Por outro lado, a combinação IMU+VO evidenciou um desvio significativo, consequência da baixa qualidade das medições visuais neste trajeto, fortemente afetadas por condições de iluminação e escassez de *features* fiáveis. Este resultado expõe uma limitação típica da abordagem *loosely-coupled*, onde medições ruidosas de um dos sensores afetam fortemente a estimativa global, uma vez que a estimativa é feita continuamente com um erro cumulativo não existindo mecanismos de correção intermédios para os algoritmos isolados. Esta lacuna na estimativa por parte da VO, já observada na trajetória isolada (Figura III.2), é agravada pela combinação com a IMU, como ilustrado na Figura III.3.

Embora fosse teoricamente possível mitigar este efeito ajustando as matrizes de covariância associadas ao ruído das observações, uma penalização excessiva resultaria na desvalorização artificial das medições da odometria visual. Assim, esta análise permite demonstrar de forma explícita o impacto negativo que medições imprecisas podem ter numa arquitetura de fusão sensorial baseada em ESKF.

A Tabela 5.4 apresenta os resultados da avaliação da trajetória estimada com recurso ao filtro ESKF . A combinação IMU+ODO registou o melhor desempenho global, com o menor erro absoluto médio ( $APE = 0.650$  m) e o menor erro quadrático médio ( $RMSE = 0.741$  m), refletindo uma trajetória estável e próxima da realidade. A fusão IMU+LIDAR obteve resultados praticamente equivalentes ( $APE = 0.655$  m,  $RMSE = 0.746$  m), demonstrando também elevada robustez.

A fusão das três fontes (IMU+ODO+LIDAR) não trouxe ganhos significativos face às combinações parciais, apresentando métricas quase idênticas. Tal sugere que, neste cenário, os contributos dos sensores LIDAR e rodas são já complementares de forma eficaz mesmo quando considerados separadamente.

Por outro lado, a combinação IMU+VO revelou um desempenho bastante inferior, com um APE de 180.241 m e um RMSE de 224.522 m, refletindo as limitações abordadas anteriormente.

TABELA 5.4: Trajeto 2 — Avaliação dos Métodos Compostos com ESKF

Método	APE[m]	RMSE	RPE[m]	APE[std]	RPE[std]
IMU + VO	180.241	224.522	0.953	133.879	17.796
IMU + ODO	<b>0.650</b>	<b>0.741</b>	<b>0.025</b>	<b>0.341</b>	<b>0.043</b>
IMU + LIDAR	0.655	0.746	0.026	0.357	<b>0.043</b>
IMU + ODO + LIDAR	0.656	0.742	0.026	0.347	<b>0.043</b>
TOTAL	1.123	1.178	0.028	0.356	0.072

Por fim, a fusão de todas as fontes sensoriais (TOTAL) não resultou numa melhoria do desempenho global, registando-se inclusivamente um ligeiro agravamento do erro absoluto médio ( $APE = 1,123$  m). Este aumento é atribuível ao ruído introduzido pelas medições da odometria visual. Ainda assim, destaca-se a capacidade do filtro em atenuar de forma significativa os erros associados a este sensor.

### 5.2.3 Trajeto 3

O terceiro ensaio foi conduzido na Parada da Escola Naval, conforme ilustrado na Figura 5.7. Este ambiente caracteriza-se por ser exterior, aberto e pouco estruturado, apresentando uma escassez significativa de elementos. A seleção deste

cenário teve como principal objetivo avaliar a robustez dos métodos em condições de campo aberto, frequentemente observadas em contextos operacionais exteriores.

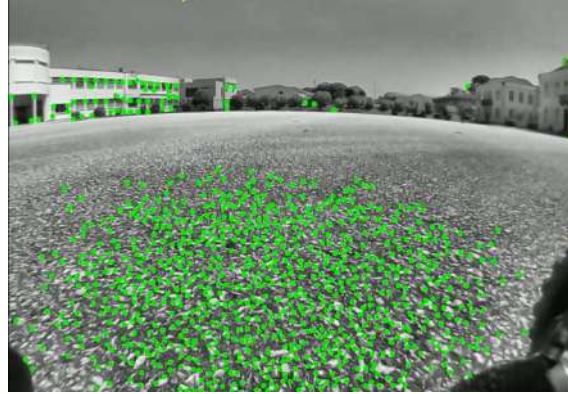
O trajeto executado pelo rover foi aproximadamente retilíneo, com uma extensão de cerca de 5 metros, sem correções de direção ao longo do percurso. A trajetória de referência foi obtida com base nas medições fornecidas pelo sistema GNSS através do processo previamente descrito.



FIGURA 5.7: Cenário experimental do Trajeto 3

Inicialmente, foram avaliados os métodos de estimação de forma isolada. A Figura III.4 apresenta as trajetórias estimadas utilizando apenas os sensores individualmente. Observa-se que a reduzida estrutura do ambiente, nomeadamente a ausência de obstáculos próximos do rover (a distâncias inferiores ao alcance do LIDAR, cerca de 25 metros), comprometeu o desempenho da odometria LIDAR. Nestas condições, o sistema produziu medições essencialmente associadas ao ruído, o que resultou numa estimativa enviesada da trajetória.

Por outro lado, a Figura III.5 mostra os resultados obtidos através da odometria de rodas e da odometria visual. Verifica-se um desempenho satisfatório de ambas as abordagens, em linha com os resultados obtidos nos cenários anteriores. No caso da odometria visual, este bom desempenho deve-se à elevada deteção de *features*, favorecida por uma iluminação natural e pela presença de textura no solo. Estas condições permitiram não só uma boa estimativa da translação entre *frames*, como também uma estimativa fiável da escala relativa. A Figura 5.8 apresenta um exemplo de *frame* utilizado durante o processamento do algoritmo.

FIGURA 5.8: *Frame* ilustrativo da detecção de *features*

A Tabela 5.5 apresenta os resultados obtidos com a aplicação individual dos métodos de estimação ao Trajeto 3. Como mencionado anteriormente, o método baseado em LIDAR registou o pior desempenho global, com um erro absoluto médio elevado ( $APE = 12.664$  m) e um erro quadrático médio ainda mais significativo ( $RMSE = 14.216$  m). Em contraste, a odometria de rodas apresentou um resultado notavelmente superior, com o menor erro absoluto médio ( $APE = 0.090$  m) e o menor erro relativo de posição ( $RPE = 0.018$  m), mantendo também os menores desvios padrão entre os métodos testados. Estes resultados confirmam a estabilidade e fiabilidade da odometria de rodas em ambientes planos e bem definidos, tal como verificado nos ensaios anteriores.

A odometria visual apresentou igualmente um desempenho robusto, com  $APE = 0.336$  m e  $RPE = 0.039$  m, beneficiando da boa iluminação e da presença de textura no solo.

TABELA 5.5: Trajeto 3 — Avaliação dos Métodos Isolados

Método	APE[m]	RMSE	RPE[m]	APE[std]	RPE[std]
LIDAR	12.664	14.216	1.200	6.459	2.877
ODO	<b>0.090</b>	<b>0.098</b>	<b>0.018</b>	<b>0.038</b>	<b>0.011</b>
VO	0.336	0.432	0.039	0.272	0.030

Seguida da avaliação dos métodos isoladamente, foi então avaliado o desempenho do filtro ESKF, à semelhança dos ensaios anteriores variando as diferentes combinações de sensores.

A Tabela 5.6 apresenta os resultados da avaliação da trajetória estimada com recurso ao filtro ESKF no Trajeto 3. A combinação IMU+ODO foi a que

apresentou o melhor desempenho global, com o menor erro absoluto médio (APE = 0.073 m) e o menor erro quadrático médio (RMSE = 0.079 m), refletindo uma trajetória bastante próxima da realidade, com elevada consistência local.

TABELA 5.6: Trajeto 3 — Avaliação dos Métodos Compostos com ESKF

Método	APE[m]	RMSE	RPE[m]	APE[std]	RPE[std]
IMU + VO	0.301	0.437	0.082	0.317	0.250
IMU + ODO	<b>0.073</b>	<b>0.079</b>	<b>0.019</b>	<b>0.029</b>	<b>0.011</b>
IMU + LIDAR	44.615	47.444	5.669	16.139	10.997
IMU + ODO + VO	0.096	0.104	0.020	0.039	0.010
TOTAL	0.839	0.873	0.050	0.239	0.159

A combinação IMU+VO apresentou um desempenho inferior, com um erro absoluto médio de 0.301 m e um RMSE de 0.437 m. Este resultado deve-se, em grande parte, a uma incongruência inicial nos alinhamentos temporais entre a informação da IMU e da VO, durante a qual a IMU propaga o estado praticamente sem correções. Esta ausência temporária de medições induz uma acumulação de erro que origina uma pequena deriva, posteriormente corrigida com a entrada das estimativas fornecidas pela odometria visual. Já a combinação IMU+ODO+VO revelou um desempenho significativamente mais robusto, com um APE de 0.096 m e um RMSE de 0.104 m. Embora estes valores sejam ligeiramente superiores aos obtidos com a fusão IMU+ODO isoladamente, a presença da odometria de rodas contribui para estabilizar a trajetória nos instantes iniciais, garantindo uma transição mais suave e consistente na integração das medições da VO.

A combinação IMU+LIDAR apresentou os piores resultados, com um erro médio muito elevado (APE = 44.615 m, RMSE = 47.444 m), comprometendo totalmente a utilidade da estimativa; tal como observado nos métodos isolados, este fraco desempenho resulta da falta de estrutura no ambiente e da ausência de obstáculos próximos. Por fim, a combinação de todos os sensores não proporcionou melhorias adicionais: embora o desempenho seja aceitável (APE = 0.839 m), o erro absoluto aumentou de forma significativa face às melhores combinações parciais, indicando que a presença de medições degradadas do LIDAR contribuiu negativamente para a estimativa final.

A Figura 5.9 apresenta as trajetórias estimadas para as diferentes combinações de sensores avaliadas. A combinação IMU+LIDAR foi omitida da visualização

para evitar comprometer a legibilidade do gráfico, uma vez que os erros associados a esta configuração são significativamente superiores em escala face às restantes.

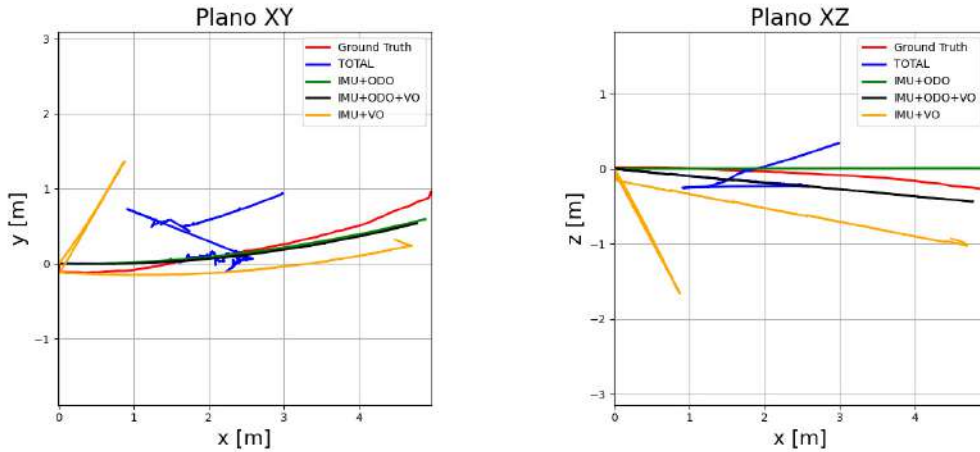


FIGURA 5.9: Trajeto 3 — Estimativas ESKF

#### 5.2.4 Trajeto 4

O quarto trajeto consistiu num percurso de geometria retangular, com aproximadamente 18 metros de comprimento por 10 metros de largura, realizado na Parada da Escola Naval. Este cenário foi concebido com o intuito de avaliar a estabilidade temporal do sistema, isto é, a sua capacidade de manter uma estimativa de pose coerente ao longo de percursos mais extensos. Em situações com maior duração de navegação inercial, os erros acumulados tornam-se mais expressivos, permitindo identificar com maior clareza eventuais deficiências na modelação do sistema ou na qualidade das correções fornecidas pelos sensores.

Neste contexto, foram inicialmente avaliados os métodos isolados. À semelhança do verificado no Trajeto 3, a odometria baseada em LIDAR apresentou o pior desempenho, devido à escassa presença de obstáculos no ambiente envolvente. A odometria visual, por sua vez, revelou também um erro elevado quando aplicada sem correções intermédias. No caso da odometria de rodas, observa-se um viés consistente que se explica, por diferenças mínimas no diâmetro efetivo das rodas: em percursos retilíneos, os encoders do lado esquerdo tendem a contabilizar uma distância ligeiramente superior à do lado direito, o que faz a estimativa integrada “virar” para a direita e distorce a trajetória. Com base numa análise empírica, foi introduzido um *bias* constante na medição da velocidade angular, compensando esse desvio sistemático. Esta correção foi aplicada à medição, uma vez que o modelo de

estado e o filtro não incluíam explicitamente essa componente. A Figura 5.10 apresenta a comparação entre as estimativas da odometria de rodas com e sem o *bias*, em relação à trajetória de referência.

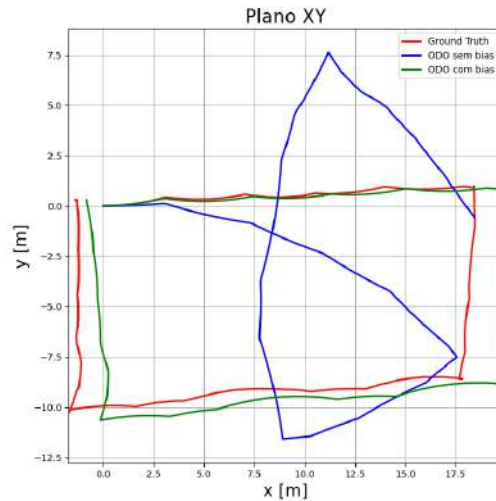


FIGURA 5.10: Comparação das diferentes estimativas obtidas pela odometria de rodas

A inclusão do *bias* revelou-se essencial para garantir estimativas fiáveis. Sem essa correção, o erro absoluto médio atingiu  $APE = 5.357\text{ m}$  e o  $RMSE = 6.296\text{ m}$ , refletindo uma acumulação significativa de erro ao longo do percurso. Com a modelação do *bias*, os valores reduziram-se para  $APE = 1.245\text{ m}$  e  $RMSE = 1.368\text{ m}$ , demonstrando a eficácia da correção em trajetos de maior extensão. Assim, a inclusão do *bias* foi mantida para os restantes testes em cenários longos.

Foram então avaliados individualmente os três métodos de odometria: visual, LIDAR e baseada em rodas, de forma a caracterizar o seu desempenho isolado no cenário do Trajeto 4.

A Figura 5.11 apresenta as estimativas de posição obtidas pelos métodos de odometria visual, LIDAR e rodas, aplicados isoladamente.

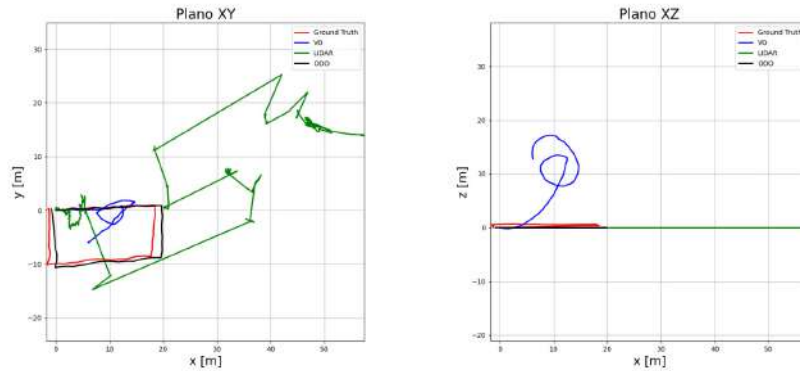


FIGURA 5.11: Estimativas com métodos isolados no Trajeto 4

A Tabela 5.7 sintetiza os resultados obtidos, com base na comparação com a trajetória de referência.

TABELA 5.7: Trajeto 4 — Avaliação dos Métodos Isolados

Método	APE [m]	RMSE	RPE [m]	APE [std]	RPE [std]
VO	13.034	14.628	0.103	6.641	0.061
LIDAR	28.721	33.032	0.353	16.316	1.658
ODO	<b>1.245</b>	<b>1.368</b>	<b>0.097</b>	<b>0.567</b>	<b>0.072</b>

A odometria baseada nas rodas (ODO) apresentou o melhor desempenho global, com um erro absoluto médio ( $APE = 1.245$  m) e erro quadrático médio ( $RMSE = 1.368$  m) significativamente inferiores aos restantes métodos.

A odometria visual (VO) registou um desempenho fraco, com  $APE = 13.034$  m e  $RMSE = 14.628$  m, valores que refletem acumulação de erro ao longo do tempo. Apesar disso, o erro relativo médio ( $RPE = 0.103$  m) manteve-se baixo, evidenciando correções locais razoavelmente fiáveis. No entanto, é evidente a acumulação de deriva ao longo do eixo  $z$ .

Por sua vez, o método baseado em LIDAR revelou o pior desempenho, com valores de erro muito elevados,  $APE = 28.721$  m e  $RMSE = 33.032$  m. Estes resultados refletem as limitações do LIDAR previamente abordadas, em ambientes de campo aberto ou sem estrutura geométrica.

Em seguida, procedeu-se à fusão dos diferentes sensores. Para efeitos de comparação, foi também avaliada uma variante com integração do sinal GNSS no filtro. Embora o foco da dissertação seja a navegação em ambientes com disrupção

GNSS, a sua utilização pontual, por exemplo em fases iniciais, é plausível. As medições GNSS foram convertidas para o referencial do rover com o mesmo processo de alinhamento usado para as trajetórias de referência. Verificou-se que a covariância fornecida pelo sensor era significativamente superior à observada na prática. Por isso, foi atribuído um valor fixo de ruído, associado a exatidão de cerca de 10 cm, mais coerente com a informação fornecida pelo sistema.

A Tabela 5.8 resume o desempenho das diferentes combinações de sensores avaliadas com o filtro ESKF, num cenário com acesso a medições GNSS.

TABELA 5.8: Trajeto com GNSS — Avaliação dos Métodos Compostos com ESKF

Método	APE[m]	RMSE	RPE[m]	APE[std]	RPE [std]
IMU + GNSS	<b>0.177</b>	<b>0.111</b>	<b>0.096</b>	<b>0.142</b>	<b>0.070</b>
IMU + ODO	1.598	1.750	0.097	0.713	0.072
IMU + VO	2.533	2.831	0.113	1.264	0.119
IMU + ODO + VO	2.973	3.374	0.097	1.595	0.070
TOTAL	3.213	3.663	0.104	1.758	0.134
IMU + LIDAR	2967.917	3788.306	41.389	2354.300	279.409

A fusão entre a IMU e o GNSS apresentou os melhores resultados, com um erro absoluto médio de apenas  $APE = 0.177$  m, um  $RMSE = 0.111$  m e erro relativo médio observado ( $RPE = 0.096$  m), refletindo uma trajetória altamente consistente e precisa. Estes resultados enquadram-se no esperado, dado que a trajetória de referência utilizada é também proveniente do GNSS.

Em contraponto, a integração entre a IMU e o LIDAR revelou-se gravemente comprometida neste cenário, com um erro absoluto médio superior a 2967 m e um  $RMSE = 3788$  m. Estes valores refletem a presença de dados severamente corrompidos provenientes do sensor LIDAR exacerbados pela integração descontrolada característica da IMU.

No caso da combinação IMU+ODO, a integração com a IMU resultou num  $APE = 1.598$  m e  $RPE = 0.097$  m, descrevendo uma trajetória suavizada mas sujeita a deriva crescente, agravada pela ausência de correções absolutas. Curiosamente, a fusão com a IMU não melhorou significativamente o desempenho face à odometria isolada.

## 5.2. Descrição dos Cenários Experimentais

A integração da IMU com a odometria visual beneficiou a estimativa, reduzindo consideravelmente a acumulação de erro ao longo do eixo  $z$ , típica da VO isolada. Apesar de apresentar um APE = 2.533 m e um RMSE = 2.831 m, esta combinação revelou um erro relativo médio moderado (RPE = 0.113 m), indicando correções locais eficazes. A comparação com a odometria visual isolada que registava APE = 13.034 m e RMSE = 14.628 m, evidencia a importância da integração da IMU na contenção da deriva acumulada.

A fusão IMU+ODO+VO registou um APE = 2.973 m e um RMSE = 3.374 m, com um erro relativo médio de RPE = 0.097 m, igual ao da combinação IMU+ODO. Esta equivalência no RPE sugere que as correções locais se mantêm eficazes, ainda que a contribuição da VO tenha introduzido perturbações na consistência global da estimativa, refletidas no aumento do APE.

A configuração com todos os sensores integrados, excepto o GNSS (TOTAL) apresentou um desempenho ligeiramente inferior à combinação IMU+ODO+VO, com APE = 3.213 m e RMSE = 3.663 m. Este resultado decorre da inclusão do LIDAR, mantém alguma influência negativa no sistema composto.

A Figura 5.12 apresenta as trajetórias estimadas pelas diferentes fusões abordadas, excluindo a IMU+LIDAR devido à discrepância acentuada de escala.

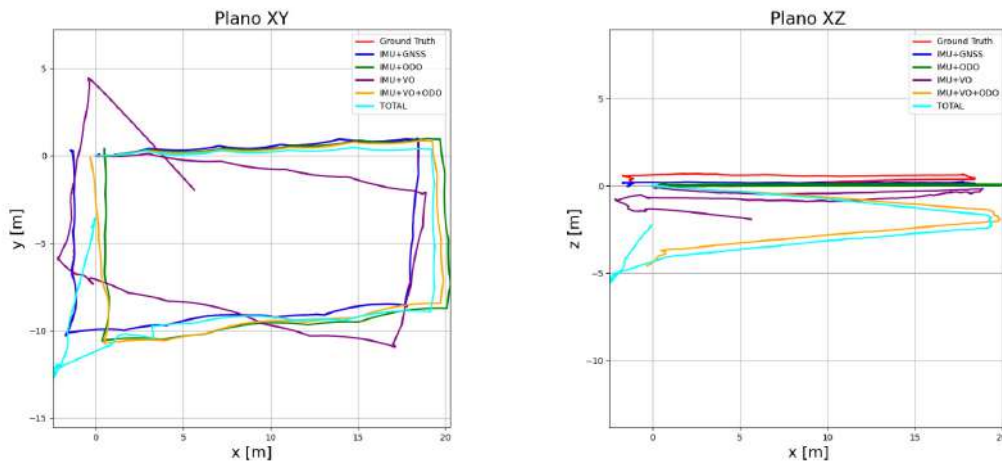


FIGURA 5.12: Trajeto 5 — Estimativas ESKF

Por fim, avaliou-se o desempenho do filtro em condições de interrupção parcial do sinal GNSS. Para tal, foram simulados três cenários distintos, em que a informação do GNSS foi disponibilizada exclusivamente durante um intervalo de 10 segundos: no início da trajetória, a meio do percurso, e nos 10 segundos finais. Esta

análise foi conduzida para as combinações, IMU+ODO, IMU+ODO+VO e a fusão completa de todos os sensores (TOTAL).

No caso da combinação IMU+ODO (Figura 5.13), observa-se que a introdução do GNSS no início do trajeto contribui para melhorar a estimativa, aproximando-a da trajetória de referência. No entanto, a inserção do sinal GNSS a meio do percurso introduz um erro significativo, desestabilizando a trajetória. Quando o sinal GNSS é introduzido apenas nos instantes finais, o impacto na estimativa é negligenciável, porém não positivo.

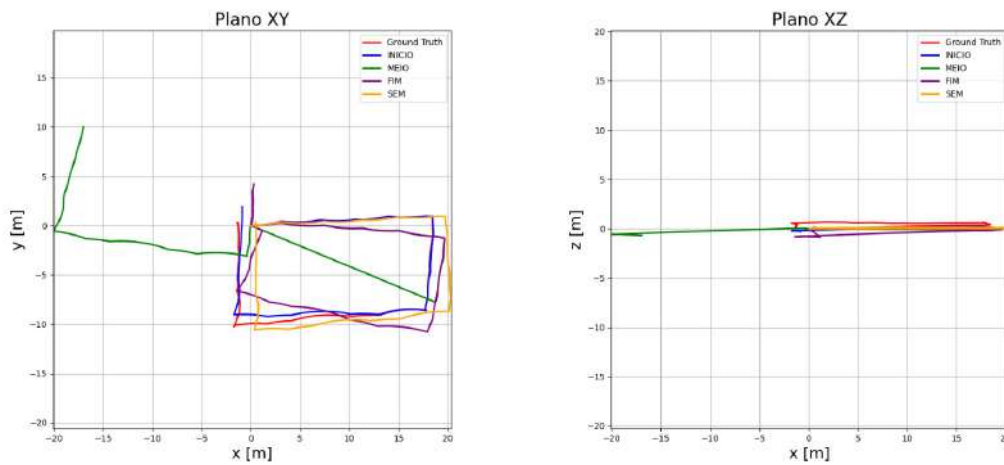


FIGURA 5.13: Comparação de diferentes estimativas — IMU+ODO

Na configuração IMU+ODO+VO (Figura 5.14), verifica-se que a introdução do GNSS, independentemente do momento, tende a distorcer a trajetória estimada. Ainda assim, observa-se um comportamento semelhante ao da combinação IMU+ODO no caso da inserção do GNSS a meio da trajetória, com uma degradação súbita da estimativa.

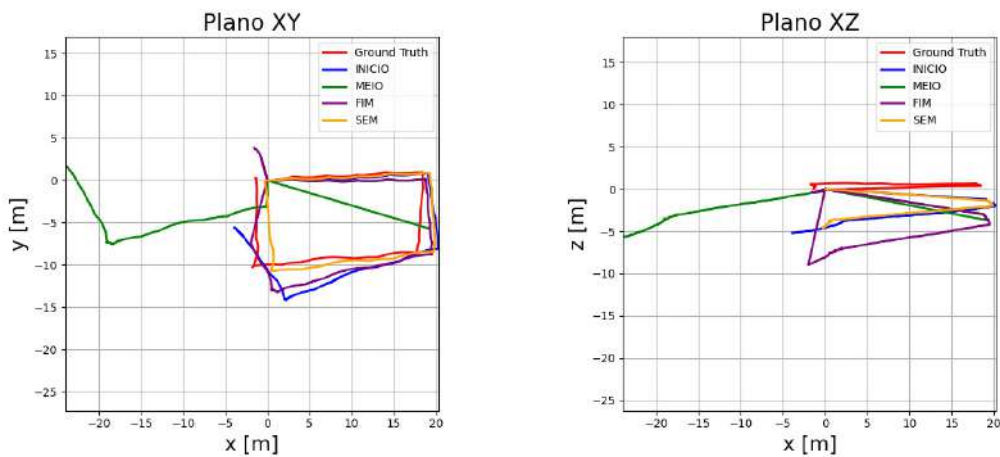


FIGURA 5.14: Comparação de diferentes estimativas — IMU+ODO+VO

Na configuração com fusão completa de todos os sensores (TOTAL), verifica-se que a introdução do GNSS em qualquer das janelas temporais testadas não conduz a melhorias na estimativa da trajetória, conforme ilustrado na Figura 5.15. O erro observado aquando da reintrodução do sinal a meio do percurso revela-se semelhante ao registado nas restantes combinações sensoriais.

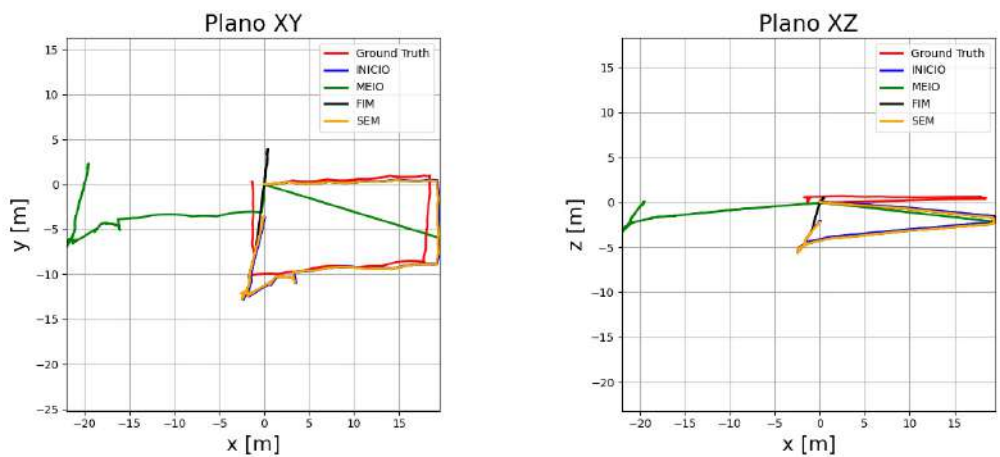


FIGURA 5.15: Comparação de diferentes estimativas — TOTAL

A origem comum do erro nos três cenários aparenta estar associada a um momento próximo dos 150 segundos de percurso, onde o filtro começa a estimar ângulos de *pitch* e *roll* significativos. Quando o sinal GNSS é introduzido fora deste intervalo, nomeadamente nos primeiros ou nos últimos 20 segundos, o filtro não consegue compensar adequadamente essa deriva, resultando nos desvios apresentados.

Esta limitação evidencia a necessidade de melhorias adicionais no modelo de fusão sensorial, especialmente em contextos dinâmicos com disponibilidade intermitente de sensores absolutos como o GNSS.

### 5.2.5 Trajeto 5

O quinto ensaio foi realizado num campo contíguo à Escola Naval, próximo da zona ilustrada na Figura 5.16. Trata-se de um cenário exterior com vegetação dispersa, zonas de sombra e folhagem em movimento, fatores que introduzem variabilidade visual e dificultam a extração de características estáveis tanto no LIDAR como na imagem. O terreno irregular permitiu ainda avaliar a robustez da odometria baseada nas rodas.



FIGURA 5.16: Zona do Trajeto 5

À semelhança dos trajetos anteriores, os métodos de odometria foram avaliados individualmente. A estimativa baseada no LIDAR revelou-se imprecisa, não devido à ausência de obstáculos, mas à variabilidade provocada pelo movimento da vegetação com o vento, que introduz ruído nas leituras e afeta negativamente a consistência dos *scans*.

As estimativas de posição obtidas pelos métodos isolados encontram-se representadas na Figura III.6. Devido à distorção de escala presente na trajetória estimada pelo LIDAR, esta foi omitida na Figura III.7, para melhor visualização dos restantes métodos.

A análise da Figura III.7 evidencia um desempenho surpreendentemente robusto da odometria visual, apesar das condições desafiantes do ambiente. A presença de sombras e objetos em movimento não comprometeu significativamente a

estimação. A Figura 5.17 ilustra um *frame* típico, onde se observa a ausência de *features* detetadas nas zonas de sombra.

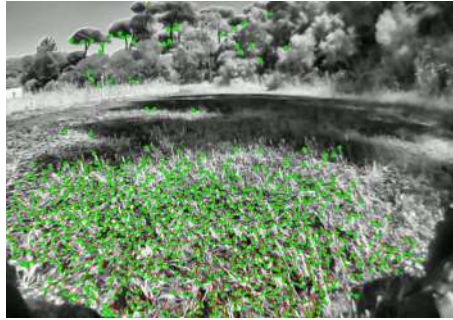


FIGURA 5.17: *Frame* ilustrativo da detecção de *features*, no Trajeto 5

A Tabela 5.9 apresenta os resultados obtidos com a aplicação individual dos métodos de estimação ao Trajeto 5. À semelhança do verificado nos restantes cenários exteriores, a odometria baseada em LIDAR apresentou o pior desempenho, com um erro absoluto médio de  $APE = 12.903\text{ m}$  e um erro quadrático médio de  $RMSE = 18.396\text{ m}$ . Estes valores refletem a instabilidade provocada pela presença de folhagem em movimento e pela escassez de superfícies estáticas confiáveis. A odometria visual obteve um desempenho robusto, com  $APE = 0.309\text{ m}$  e  $RPE = 0.037\text{ m}$ , sendo ligeiramente superada pela odometria de rodas em termos de erro médio. A odometria de rodas apresentou os melhores resultados neste cenário, com  $APE = 0.215\text{ m}$  e  $RPE = 0.021\text{ m}$ , demonstrando que apesar de terreno menos regular, continua a ser um sensor bastante robusto.

TABELA 5.9: Trajeto 5 — Avaliação dos Métodos Isolados

Método	APE [m]	RMSE	RPE [m]	APE [std]	RPE [std]
LIDAR	12.903	18.396	1.304	13.112	5.187
VO	0.309	0.417	0.037	0.059	0.081
ODO	<b>0.215</b>	<b>0.211</b>	<b>0.021</b>	<b>0.015</b>	<b>0.017</b>

Posteriormente à avaliação dos métodos isoladamente foi também avaliada a fusão dos múltiplos sensores. A Tabela 5.10 apresenta os resultados da avaliação da trajetória estimada com recurso ao filtro ESKF no Trajeto 5.

Tal como verificado nos testes anteriores, a fusão IMU+LIDAR apresentou o pior desempenho global, com um erro absoluto médio de  $APE = 15.136\text{ m}$  e

um RMSE de 20.385 m, evidenciando a sua sensibilidade à presença de vegetação dinâmica e à ausência de estrutura no ambiente. Também o erro relativo foi significativamente elevado (RPE = 1.451 m), inviabilizando a sua utilização prática neste cenário.

As fusões IMU+VO+ODO e IMU+ODO revelaram o desempenho mais robusto, com erros absolutos médios de APE = 0.404 m e 0.416 m, respetivamente. Ambas as configurações apresentaram os menores erros relativos (RPE = 0.023 m), bem como valores de RMSE bastante reduzidos e desvios padrão baixos, indicando trajetórias consistentes, com elevada precisão global e local. A combinação IMU+VO revelou um erro absoluto médio mais elevado (APE = 0.898 m), apesar de um desempenho local razoável (RPE = 0.034 m). A ausência de correções provenientes da odometria de rodas resultou numa deriva acumulada ao longo da trajetória, que não foi compensada apenas com as estimativas visuais. A configuração IMU+ODO+LIDAR apresentou um erro ligeiramente superior (APE = 0.477 m), sem ganhos evidentes face às fusões sem LIDAR. Esta combinação mostrou um erro relativo superior (RPE = 0.029 m). Demonstrado que a inclusão do LIDAR não trouxe benefícios relevantes neste contexto.

Finalmente, a fusão de todos os sensores resultou num erro APE = 0.429 m e RMSE = 0.462 m, valores próximos das melhores configurações parciais. Embora o LIDAR tenha impacto negativo, a presença simultânea de medições robustas da VO e da odometria permitiu ao filtro manter uma estimativa global estável.

TABELA 5.10: Trajeto 5 — Avaliação dos Métodos Compostos com ESKF

Método	APE[m]	RMSE	RPE[m]	APE[std]	RPE[std]
IMU + VO	0.898	1.108	0.034	0.650	0.050
IMU + ODO	0.416	0.472	<b>0.023</b>	<b>0.202</b>	<b>0.016</b>
IMU + LIDAR	15.136	20.385	1.451	13.654	5.299
IMU + ODO + VO	<b>0.404</b>	<b>0.436</b>	<b>0.023</b>	0.244	<b>0.016</b>
IMU + ODO + LIDAR	0.477	0.541	0.029	0.257	0.024
TOTAL	0.429	0.462	0.025	0.216	0.018

A Figura 5.18 apresenta as estimativas de trajetória obtidas pelas diferentes combinações de sensores. A análise visual permite ainda identificar a presença de um erro sistemático associado à odometria visual, comum a vários trajetos, que se manifesta na tendência para estimar em demasia deslocamento ao longo do eixo  $z$ .

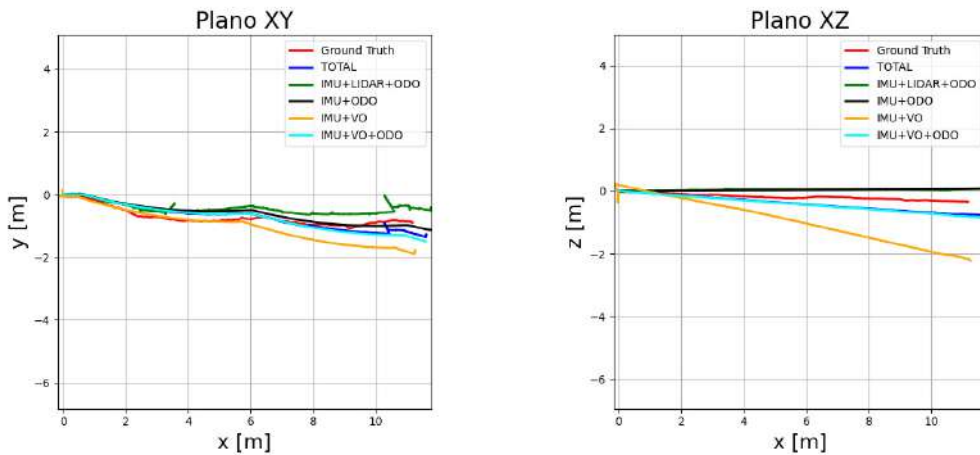


FIGURA 5.18: Trajeto 5 — Estimativas ESKF

### 5.3 Considerações Adicionais

A análise sistemática realizada ao longo dos diferentes cenários experimentais permitiu não apenas aferir o desempenho individual e combinado dos sensores considerados, mas também identificar padrões transversais e limitações recorrentes, cuja compreensão é essencial para a concepção e desenvolvimento de sistemas de navegação robustos, capazes de operar em ambientes com interrupção de sinal GNSS.

No que respeita aos sensores individuais, a odometria baseada nas rodas revelou, em geral, o desempenho mais consistente e robusto. No entanto, foi evidenciada a sua vulnerabilidade à deriva sistemática e à acumulação de erro ao longo de trajetos extensos. Para mitigar esta limitação, foi aplicada uma compensação empírica ao *bias* da velocidade angular, a qual demonstrou ganhos significativos em trajetos longos. Contudo, para uma abordagem mais sistemática e rigorosa, recomenda-se a futura integração do *bias* como variável explícita no vetor de estado do filtro ESKF.

A odometria baseada em LIDAR apresentou um desempenho altamente dependente da estrutura do ambiente. Em cenários interiores com superfícies estáticas bem definidas, o algoritmo produziu estimativas precisas e estáveis. No entanto, em ambientes abertos, pouco estruturados ou com elementos dinâmicos, a qualidade das correspondências degradou-se substancialmente, comprometendo a fiabilidade da estimativa. Neste contexto, a integração de um sensor LIDAR tridimensional, otimizado para operação em ambientes exteriores, constitui uma evolução promissora ao aumentar significativamente a quantidade e diversidade de informação espacial disponível. Tal sensor permitirá melhorar a estimativa tanto da orientação

como da posição do veículo, mesmo em cenários de mais reduzida estrutura.

A odometria visual apresentou um desempenho mais variável, fortemente influenciado pelas condições de iluminação. Embora sujeita a acumulação de erro ao longo do tempo, particularmente evidente no eixo vertical, a odometria visual demonstrou coerência local satisfatória, como evidenciado pelos baixos valores de erro relativo em vários trajetos. Tal característica torna-a especialmente útil enquanto fonte de correções incrementais, sendo porém muito suscetível a acumulação de erros de deriva ao longo de um trajeto.

A avaliação dos métodos compostos demonstrou inequivocamente os benefícios da fusão sensorial através do filtro ESKF. A integração de medições inerciais com observações externas, sejam elas visuais, geométricas ou baseadas em rodas, permitiu mitigar a deriva típica da navegação inercial pura, estabilizando a trajetória estimada e melhorando a coerência global da pose. De forma geral, as configurações de fusão apresentaram desempenhos superiores face às estimativas isoladas, validando a abordagem adotada. Não obstante, a eficácia da fusão mostrou-se fortemente dependente da qualidade e fiabilidade das medições envolvidas. A presença de sensores com ruído significativo ou medições inconsistentes, como observado no caso do LIDAR em campo aberto ou da VO em condições de iluminação adversa, introduziu perturbações relevantes na trajetória estimada, mesmo após ponderação pelas respectivas covariâncias. Este comportamento evidencia uma das fragilidades inerentes à abordagem *loosely-coupled*, onde a ausência de mecanismos de correção intermédia pode levar a deriva descontrolada, inviabilizando a estimativa final.

Importa ainda salientar que, em diversos trajetos, as melhores estimativas foram obtidas com configurações sensoriais parciais (e.g., IMU+ODO+LIDAR), sendo que a inclusão de todos os sensores disponíveis (configuração TOTAL) nem sempre se traduziu em melhorias adicionais. Esta observação realça a importância de uma afinação mais cuidada do filtro, bem como de uma seleção criteriosa dos sensores a integrar, devendo esta ser adaptada às características específicas de cada ambiente. De forma geral, observou-se que a combinação IMU+ODO+LIDAR é mais eficaz em ambientes interiores, enquanto a configuração IMU+ODO+VO tende a produzir melhores resultados em cenários exteriores, beneficiando das condições de iluminação natural proporcionadas pela luz solar.

A introdução de correções GNSS, mesmo que de forma episódica, demonstrou potencial para estabilizar trajetórias e limitar a propagação de erro, mas também revelou limitações relevantes na integração destes dados com o restante sistema. Em particular, a ocorrência de picos de erro aquando da reintrodução de medições

GNSS indica que o modelo de fusão atual carece de mecanismos robustos de transição e adaptação quando confrontado com dados intermitentes. Assim, é evidente a necessidade de um módulo mais avançado de gestão da disponibilidade sensorial, que permita integrar o GNSS de forma eficiente e segura, mitigando os efeitos adversos de alinhamentos espaciais ou temporais imperfeitos.

Na secção seguinte são apresentadas as principais conclusões desta dissertação, bem como propostas para trabalhos futuros.



# Capítulo 6

## Conclusões

No decurso da presente dissertação, foi concebida e implementada uma abordagem de localização e navegação para veículos terrestres não tripulados (UGVs) em cenários caracterizados por interrupção ou degradação severa do sinal GNSS. A investigação foi conduzida com recurso ao Leo Rover, uma plataforma robótica versátil equipada com diversos sensores, sobre a qual se integrou e avaliou a arquitetura proposta. O foco centrou-se na fusão de múltiplas fontes sensoriais, com especial ênfase na integração de unidades inerciais (IMU), medições provenientes dos *encoders* das rodas, dados de um sensor LIDAR e imagens captadas pela câmara do rover, utilizando *Error-State Kalman Filter* (ESKF).

Foi inicialmente realizado um enquadramento teórico abrangente, onde se abordaram os principais conceitos relacionados com os sensores utilizados, fundamentos de odometria, representações de posição e orientação e estratégias de fusão sensorial. Esta análise foi complementada com uma revisão da literatura, de modo a contextualizar o estado da arte e as abordagens atuais à problemática em estudo.

A arquitetura adotada, seguiu uma abordagem *loosely-coupled* ou de *late fusion*, na qual a estimativa do estado é inicialmente obtida de forma independente pelas medições de cada sensor, sendo posteriormente fundida no filtro ESKF. No caso da odometria de rodas, foram integradas as velocidades linear e angular provenientes dos *encoders*. Para a odometria LIDAR, implementou-se um algoritmo ICP, que, com base na correspondência entre *scans* consecutivos, permitiu estimar incrementos de translação nos eixos  $x$  e  $y$ , e de rotação em torno de  $z$  ( $\psi$ ). A odometria visual baseou-se na deteção e seguimento de *features* entre imagens consecutivas através de um algoritmo baseado no fluxo ótico KLT, permitindo estimar o movimento relativo do veículo. Para resolver o problema da escala, foi desenvolvida uma metodologia baseada na identificação de uma região do solo na imagem, à qual se ajustou um plano. Comparando a distância perpendicular do centro ótico da câmara a este

plano com a altura real do rover ao solo, foi possível estimar uma escala relativa para converter o deslocamento na imagem em deslocamento físico.

Foram realizados cinco ensaios distintos, desenhados para testar de forma sistemática os limites e fragilidades de cada abordagem isoladamente, assim como avaliar os benefícios da fusão de múltiplas fontes sensoriais.

Os resultados experimentais demonstraram que a utilização do ESKF permite atenuar eficazmente a deriva acumulada associada ao uso isolado dos sensores, assegurando uma estimação contínua, estável e coerente da pose do veículo em ambientes sem sinal GNSS. Verificou-se que a precisão da trajetória estimada depende fortemente da configuração de sensores utilizada, destacando-se as combinações IMU+ODO+LIDAR em ambientes interiores e IMU+ODO+VO em cenários exteriores como as mais eficazes.

Adicionalmente, foi evidenciado que a fiabilidade das estimativas depende diretamente da qualidade das observações. Medições ruidosas ou inconsistentes, como as da odometria visual em condições de iluminação deficiente ou as do LIDAR em ambientes com baixa estrutura geométrica, comprometeram a performance da fusão sensorial, revelando uma limitação típica das abordagens do tipo *loosely-coupled*. Ainda assim, a introdução de mecanismos adaptativos, como o ajuste dinâmico das matrizes de covariância em função da qualidade das medições, mostrou-se eficaz na mitigação desses efeitos adversos.

A análise das trajetórias revelou que a inclusão de todos os sensores disponíveis nem sempre resulta em melhorias significativas. Em vários cenários, configurações sensoriais parciais, cuidadosamente selecionadas, superaram a combinação total. Esta constatação reforça a importância de estratégias adaptativas de seleção e ponderação dos sensores empregar ajustadas ao ambiente e às condições operacionais.

A integração de correções GNSS esporádicas demonstrou ser benéfica para estabilizar a trajetória e mitigar a acumulação de erro, desde que essas correções sejam cuidadosamente sincronizadas e validadas. Contudo, foram identificados picos de erro associados à sua introdução, sublinhando a necessidade de mecanismos mais robustos de sincronização temporal e espacial.

Os principais desafios decorrem do facto de o estado ser estimado a partir de sensores relativos; com exceção do GNSS, que fornece posição absoluta, a ausência de medições absolutas conduz inevitavelmente à acumulação temporal de erro. Adicionalmente, a fusão de múltiplos sensores definidos em referenciais distintos

aumenta a complexidade do sistema, exigindo transformações rigorosas e consistentes para um referencial comum. Por fim, a calibração dos parâmetros do ESKF é particularmente exigente, sobretudo na determinação de *bias* e na modelação das incertezas das medições, sendo que parametrizações inadequadas podem induzir comportamentos instáveis ou divergência.

De forma global, os métodos desenvolvidos evidenciaram um desempenho promissor na garantia de navegação autónoma em ambientes adversos com disrupção prolongada do sinal GNSS. Apesar de subsistirem limitações os resultados obtidos constituem uma base sólida para desenvolvimento futuro, tanto em termos de robustez como de fiabilidade do sistema.

A secção seguinte apresenta propostas de trabalho futuro, com o objetivo de ultrapassar as limitações identificadas e alargar o domínio de aplicação da abordagem desenvolvida.

## 6.1 Trabalho Futuro

Ao longo do desenvolvimento, implementação e validação do sistema, foram identificadas diversas limitações e oportunidades de melhoria que justificam a continuidade da investigação. Em particular, destaca-se a necessidade de reforçar a robustez, adaptabilidade e fiabilidade da solução em contextos operacionais exigentes, especialmente em cenários com disrupção do sinal GNSS e elevada variabilidade ambiental. As propostas que se seguem visam ultrapassar os constrangimentos detetados e potenciar a aplicação prática da arquitetura desenvolvida em ambientes reais.

Uma das linhas de investigação prioritárias consiste na melhoria do algoritmo de odometria visual, nomeadamente através da implementação com diferentes métodos de deteção e descrição de *features*, com o objetivo de reduzir a deriva acumulada quando o sistema opera de forma isolada. A robustez da estimativa poderá ser significativamente reforçada com a adoção de técnicas mais resilientes à variação de iluminação e textura do solo.

Adicionalmente, a substituição do sensor LIDAR 2D por um LIDAR tridimensional, devidamente adaptado à operação em ambientes exteriores, representa uma evolução promissora. A maior densidade e riqueza de informação geométrica permitiria melhorar substancialmente a qualidade das estimativas de pose, mesmo em contextos pouco estruturados.

Outra extensão natural do trabalho seria a inclusão explícita do *bias* sistemático da odometria das rodas no vetor de estado do ESKF, permitindo que o filtro o estime dinamicamente e, assim, compense de forma mais eficaz os erros acumulados ao longo do tempo.

A afinação das matrizes de covariância associadas ao ruído dos sensores constitui igualmente uma vertente crítica. A calibração cuidadosa destes parâmetros por métodos adaptativos é essencial para garantir o equilíbrio entre a propagação e a correção no filtro, especialmente em situações de receção esporádica de sinal GNSS ou durante a calibração inicial.

Uma direção de desenvolvimento particularmente relevante consiste na comparação sistemática da abordagem adotada com metodologias alternativas, nomeadamente arquiteturas de fusão do tipo *tightly-coupled* e algoritmos de SLAM. Estas abordagens, embora mais complexas do ponto de vista computacional e de implementação, oferecem um nível de integração mais profundo entre sensores, o que pode traduzir-se numa melhor consistência e precisão na estimação da pose, especialmente em ambientes degradados. A realização de testes comparativos entre estas diferentes arquiteturas, aplicadas aos mesmos dados experimentais recolhidos com o Leo Rover, permitiria aferir de forma objetiva os ganhos e compromissos associados a cada paradigma, contribuindo para decisões mais informadas na escolha de soluções para cenários operacionais concretos.

Por fim, seria pertinente conduzir testes que envolvam transições entre ambientes interiores e exteriores, com o objetivo de validar a capacidade do sistema para lidar com mudanças abruptas nas fontes de observação. A utilização de uma trajetória de referência obtida por um sistema externo fiável, permitiria quantificar com maior rigor os erros e validar o comportamento do sistema em cenários realistas.

O material resultante desta dissertação encontra-se disponível no repositório GitHub<sup>5</sup>, estando igualmente disponíveis vídeos de cada cenário experimental no YouTube<sup>6</sup>

---

<sup>5</sup>[https://github.com/frederico-rasinhas/Dissertacao\\_de\\_Mestrado](https://github.com/frederico-rasinhas/Dissertacao_de_Mestrado)

<sup>6</sup>[https://youtube.com/playlist?list=PLIHwYXXCh30rJB9YSrMBbW2YKTRc4kW\\_j&si=MYgxfIzkiPLRrFJc](https://youtube.com/playlist?list=PLIHwYXXCh30rJB9YSrMBbW2YKTRc4kW_j&si=MYgxfIzkiPLRrFJc)

# Bibliografia

- Alteriis, Giorgio de, Verdiana Bottino, Claudia Conte, Giancarlo Rufino e Rosario Schiano Lo Moriello (2021). «Accurate Attitude Initialization Procedure based on MEMS IMU and Magnetometer Integration». Em: pp. 1–6. DOI: 10.1109/MetroAeroSpace51421.2021.9511679.
- Aqel, Mohammad OA, Mohammad H Marhaban, M Iqbal Saripan e Napsiah Bt Ismail (2016). «Review of visual odometry: types, approaches, challenges, and applications». Em: *SpringerPlus* 5, pp. 1–26.
- Bay, Herbert, Andreas Ess, Tinne Tuytelaars e Luc Van Gool (2008). «Speeded-up robust features (SURF)». Em: *Computer vision and image understanding* 110.3, pp. 346–359.
- Bayer, B. E. (jul. de 1976). «Color Imaging Array». Pat. US 3971065.
- Behroozpour, Behnam, Phillip A. M. Sandborn, Ming C. Wu e Bernhard E. Boser (2017). «Lidar System Architectures and Circuits». Em: *IEEE Communications Magazine* 55.10, pp. 135–142. DOI: 10.1109/MCOM.2017.1700030.
- Bloesch, Michael, Sammy Omari, Marco Hutter e Roland Siegwart (2015). «Robust visual inertial odometry using a direct EKF-based approach». Em: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 298–304. DOI: 10.1109/IROS.2015.7353389.
- Campos, Carlos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel e Juan D Tardós (2021). «Orb-slam3: An accurate open-source library for visual, visual-inertial, and multimap slam». Em: *IEEE transactions on robotics* 37.6, pp. 1874–1890.
- Canon (2025). *Image Sensors Explained*. Acessado em: 6 de janeiro de 2025. URL: <https://www.canon.pt/pro/infobank/image-sensors-explained/>.
- Defense Mapping Agency (set. de 1987). *Department of Defense World Geodetic System 1984: Its Definition and Relationships with Local Geodetic Systems*. Technical Report TR8350.2. Revisions published in 1991, 1997 and amended through 2004. USA: Defense Mapping Agency.
- Dellaert, Frank, Michael Kaess et al. (2017). «Factor graphs for robot perception». Em: *Foundations and Trends® in Robotics* 6.1-2, pp. 1–139.

- Du, Shaoyi, Nanning Zheng, Shihui Ying, Qubo You e Yang Wu (2007). «An extension of the ICP algorithm considering scale factor». Em: *2007 IEEE International Conference on Image Processing*. Vol. 5. IEEE, pp. V–193.
- Dudek, Gregory e Michael Jenkin (2010). *Computational Principles of Mobile Robotics*. 2nd. Cambridge: Cambridge University Press.
- Egea-Roca, Daniel, Markel Arizabaleta-Diez, Thomas Pany, Felix Antreich, José A. López-Salcedo, Matteo Paonni e Gonzalo Seco-Granados (2022). «GNSS User Technology: State-of-the-Art and Future Trends». Em: *IEEE Access* 10, pp. 39939–39968. DOI: 10.1109/ACCESS.2022.3165594.
- Fictionlab (2025). *Leo Rover 1.9 Specification*. <https://docs.fictionlab.pl/leo-rover/documentation/specification>. Acedido a 10 de junho de 2025.
- Filip, Iulian, Juhyun Pyo, Meungsuk Lee e Hangil Joe (2023). «LiDAR SLAM with a Wheel Encoder in a Featureless Tunnel Environment». Em: *Electronics* 12.4, p. 1002. DOI: 10.3390/electronics12041002.
- Fischler, Martin A e Robert C Bolles (1981). «Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography». Em: *Communications of the ACM* 24.6, pp. 381–395.
- Forster, Christian, Luca Carlone, Frank Dellaert e Davide Scaramuzza (2016). «On-manifold preintegration for real-time visual–inertial odometry». Em: *IEEE Transactions on Robotics* 33.1, pp. 1–21.
- Gad-el-Hak, Mohamed (2001). *The MEMS handbook*. CRC press.
- Gonzalez, Rafael C. e Richard E. Woods (2008). *Digital Image Processing*. 3rd. Pearson Education. URL: [https://www.researchgate.net/publication/333856607\\_Digital\\_Image\\_Processing\\_Second\\_Edition](https://www.researchgate.net/publication/333856607_Digital_Image_Processing_Second_Edition).
- Gouveia e Melo, Henrique (jul. de 2019). *Uma Marinha útil e minimamente significativa para Portugal no início do séc. XXI*. Cadernos Navais 53. Ensaio estratégico publicado no âmbito do CEEM da Marinha Portuguesa. Lisboa: Comissão Cultural da Marinha / Centro de Estudos Estratégicos da Marinha. ISBN: 978-989-8159-92-2.
- Gräter, Johannes, Tobias Schwarze e Martin Lauer (2015). «Robust scale estimation for monocular visual odometry using structure from motion and vanishing points». Em: *2015 IEEE Intelligent Vehicles Symposium (IV)*, pp. 475–480. DOI: 10.1109/IVS.2015.7225730.
- Grupp, Michael (2017). *evo: Python package for the evaluation of odometry and SLAM*. <https://github.com/MichaelGrupp/evo>. Accessed: July 2025.

- Hamilton, William Rowan (1844). «Ii. on quaternions; or on a new system of imaginaries in algebra». Em: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 25.163, pp. 10–13.
- Harris, Chris, Mike Stephens et al. (1988). «A combined corner and edge detector». Em: *Alvey vision conference*. Vol. 15. 50. Citeseer, pp. 10–5244.
- Hartley, Richard e Andrew Zisserman (2003). *Multiple View Geometry in Computer Vision*. Cambridge University Press. ISBN: 9780521540513. URL: <https://www.cambridge.org/9780521540513>.
- Hoppe, Hugues, Tony DeRose, Tom Duchamp, John McDonald e Werner Stuetzle (1992). «Surface reconstruction from unorganized points». Em: *Proceedings of the 19th annual conference on computer graphics and interactive techniques*, pp. 71–78.
- Huber, Peter J (1992). «Robust estimation of a location parameter». Em: *Breakthroughs in statistics: Methodology and distribution*. Springer, pp. 492–518.
- ISO 20473:2007 - Optics and photonics – Spectral bands (2007). International Organization for Standardization. URL: <https://www.iso.org/standard/37777.html>.
- Julier, Simon J e Jeffrey K Uhlmann (1997). «New extension of the Kalman filter to nonlinear systems». Em: *Signal processing, sensor fusion, and target recognition VI*. Vol. 3068. Spie, pp. 182–193.
- Kalman, Rudolph Emil (1960). «A new approach to linear filtering and prediction problems». Em.
- Ke, Qifa e Takeo Kanade (2003). «Transforming camera geometry to a virtual downward-looking camera: robust ego-motion estimation and ground-layer detection». Em: *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings*. Vol. 1, pp. I–I. DOI: 10.1109/CVPR.2003.1211366.
- Konolige, Kurt, Motilal Agrawal e Joan Sola (2011). «Large-scale visual odometry for rough terrain». Em: *Robotics Research: The 13th International Symposium ISRR*. Springer, pp. 201–212.
- Li, Mingyang e Anastasios I Mourikis (2013). «High-precision, consistent EKF-based visual-inertial odometry». Em: *The International Journal of Robotics Research* 32.6, pp. 690–711.
- Li, You e Javier Ibanez-Guzman (2020). «Lidar for Autonomous Driving: The Principles, Challenges, and Trends for Automotive Lidar and Perception Systems». Em: *IEEE Signal Processing Magazine* 37.4, pp. 50–61. DOI: 10.1109/MSP.2020.2973615.

- Lowe, David G (1999). «Object recognition from local scale-invariant features». Em: *Proceedings of the seventh IEEE international conference on computer vision*. Vol. 2. Ieee, pp. 1150–1157.
- Lucas, Bruce D e Takeo Kanade (1981). «An iterative image registration technique with an application to stereo vision». Em: *IJCAI'81: 7th international joint conference on Artificial intelligence*. Vol. 2, pp. 674–679.
- Lukac, R., K.N. Plataniotis e D. Hatzinakos (2005). *Color image zooming on the Bayer pattern*. DOI: 10.1109/TCSVT.2005.856923.
- Meyers, Drew, Wei Wang, Tixiao Shan, Brendan Englot, Carlo Ratti e Daniela Rus (2020). «LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping». Em: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5135–5142. DOI: 10.1109/IR0S45743.2020.9341176.
- Mohamed, Sherif A.S., Mohammad Hashem Haghbayan, Tomi Westerlund, Jukka Heikkonen, Hannu Tenhunen e Juha Plosila (2019). «A Survey on Odometry for Autonomous Navigation Systems». Em: *IEEE Access* 7, pp. 97466–97486. ISSN: 21693536. DOI: 10.1109/ACCESS.2019.2929133.
- Mourikis, Anastasios I e Stergios I Roumeliotis (2007). «A multi-state constraint Kalman filter for vision-aided inertial navigation». Em: *Proceedings 2007 IEEE international conference on robotics and automation*. IEEE, pp. 3565–3572.
- Narasimhappa, Mundla, Arun D Mahindrakar, Vitor Campagnolo Guizilini, Marco Henrique Terra e Samrat L Sabat (2019). «MEMS-based IMU drift minimization: Sage Husa adaptive robust Kalman filtering». Em: *IEEE Sensors Journal* 20.1, pp. 250–260.
- Nistér, David, Oleg Naroditsky e James Bergen (2004). «Visual odometry». Em: *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004*. Vol. 1. Ieee, pp. I–I.
- Pizer, Stephen M, E Philip Amburn, John D Austin, Robert Cromartie, Ari Gesselowitz, Trey Greer, Bart ter Haar Romeny, John B Zimmerman e Karel Zuiderveld (1987). «Adaptive histogram equalization and its variations». Em: *Computer vision, graphics, and image processing* 39.3, pp. 355–368.
- Plataniotis, Konstantinos N. e Anastasios N. Venetsanopoulos (2000). *Color Image Processing and Applications*. Berlin, Heidelberg: Springer. DOI: 10.1007/978-3-662-04186-4.

- Qin, Tong, Peiliang Li e Shaojie Shen (2018). «Vins-mono: A robust and versatile monocular visual-inertial state estimator». Em: *IEEE transactions on robotics* 34.4, pp. 1004–1020.
- Rosten, Edward e Tom Drummond (2006). «Machine learning for high-speed corner detection». Em: *Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part I 9*. Springer, pp. 430–443.
- Roumeliotis, Stergios I, Gaurav S Sukhatme e George A Bekey (1999). «Circumventing dynamic modeling: Evaluation of the error-state kalman filter applied to mobile robot localization». Em: *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*. Vol. 2. IEEE, pp. 1656–1663.
- Rublee, Ethan, Vincent Rabaud, Kurt Konolige e Gary Bradski (2011). «ORB: An efficient alternative to SIFT or SURF». Em: *2011 International conference on computer vision*. Ieee, pp. 2564–2571.
- Rusinkiewicz, Szymon e Marc Levoy (2001). «Efficient variants of the ICP algorithm». Em: *Proceedings third international conference on 3-D digital imaging and modeling*. IEEE, pp. 145–152.
- Rusu, Radu B. e Steve Cousins (2011). «3D is here: Point Cloud Library (PCL)». Em: *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–4.
- Scaramuzza, Davide e Friedrich Fraundorfer (2011). «Visual odometry [tutorial]». Em: *IEEE robotics & automation magazine* 18.4, pp. 80–92.
- Shan, Tixiao e Brendan Englot (2018). «LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain». Em: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4758–4765. DOI: 10.1109/IROS.2018.8594299.
- Shi, Jianbo et al. (1994). «Good features to track». Em: *1994 Proceedings of IEEE conference on computer vision and pattern recognition*. IEEE, pp. 593–600.
- Siciliano, Bruno e Oussama Khatib (2016). *Springer Handbook of Robotics*. Berlin, Heidelberg: Springer.
- SLAMTEC (2025). *RPLIDAR A3 Laser Range Scanner*. <https://www.slamtec.com/en/Lidar/A3/>. Acedido a 16 de Junho de 2025.
- Sola, Joan (2017). «Quaternion kinematics for the error-state Kalman filter». Em: *arXiv preprint arXiv:1711.02508*. URL: <https://arxiv.org/abs/1711.02508>.

- Song, S., M. Chandraker e C. C. Guest (2015). «High accuracy monocular SfM and scale correction for autonomous driving». Em: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.4, pp. 730–743. DOI: 10.1109/TPAMI.2015.2470659.
- Sturm, Jürgen, Nikolas Engelhard, Felix Endres, Wolfram Burgard e Daniel Cremers (2012). «A benchmark for the evaluation of RGB-D SLAM systems». Em: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 573–580. DOI: 10.1109/IRoS.2012.6385773.
- Super-MP3D, Starter Set (2025). Acedido a 16 de Junho de 2025. URL: <https://marvelmind.com/product/starter-set-super-mp-3d/>.
- Szeliski, Richard (2022). *Computer Vision: Algorithms and Applications*. Springer Nature. URL: <https://link.springer.com/book/10.1007/978-3-030-34372-9>.
- Tomasi, Carlo e Takeo Kanade (1991). «Detection and tracking of point». Em: *Int J Comput Vis* 9.137-154, p. 3.
- UBlox EVK-M8N GPS Module (2025). <https://www.leorover.tech/shop/gps-module>. Acedido a 16 de Junho de 2025.
- Wan, E.A. e R. Van Der Merwe (2000). «The unscented Kalman filter for nonlinear estimation». Em: *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, pp. 153–158. DOI: 10.1109/ASSPCC.2000.882463.
- Woodman, Oliver J. (2007). *An introduction to inertial navigation*. Rel. téc. UCAM-CL-TR-696. University of Cambridge, Computer Laboratory. DOI: 10.48456/tr-696. URL: <https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-696.pdf>.
- Xu, Wei, Yixi Cai, Dongjiao He, Jiarong Lin e Fu Zhang (2022). «Fast-lio2: Fast direct lidar-inertial odometry». Em: *IEEE Transactions on Robotics* 38.4, pp. 2053–2073.
- Xu, Wei e Fu Zhang (2021). «Fast-lio: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter». Em: *IEEE Robotics and Automation Letters* 6.2, pp. 3317–3324.
- Zhang, Ji, Sanjiv Singh et al. (2014). «LOAM: Lidar odometry and mapping in real-time.» Em: *Robotics: Science and systems*. Vol. 2. 9. Berkeley, CA, pp. 1–9.

- Zheng, Chunran, Wei Xu, Zuhao Zou, Tong Hua, Chongjian Yuan, Dongjiao He, Bingyang Zhou, Zheng Liu, Jiarong Lin, Fangcheng Zhu et al. (2024). «Fast-livo2: Fast, direct lidar-inertial-visual odometry». Em: *IEEE Transactions on Robotics*.
- Zheng, Chunran, Qingyan Zhu, Wei Xu, Xiyuan Liu, Qizhi Guo e Fu Zhang (2022). «Fast-livo: Fast and tightly-coupled sparse-direct lidar-inertial-visual odometry». Em: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 4003–4009.
- Zhou, D., Y. Dai e H. Li (2019). «Ground-plane-based absolute scale estimation for monocular visual odometry». Em: *IEEE Transactions on Intelligent Transportation Systems* 21.2, pp. 791–802. DOI: 10.1109/TITS.2019.2894300.
- Zuo, Xingxing, Patrick Geneva, Woosik Lee, Yong Liu e Guoquan Huang (2019). «Lic-fusion: Lidar-inertial-camera odometry». Em: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 5848–5854.



## Anexo I - Classificação e Exemplo de Sensores

A Tabela I.1 apresenta vários exemplos de sensores, classificados segundo a natureza da medição, o tipo de percepção que proporcionam, designadamente interna (PI ou *proprioception*), quando aferem o estado do próprio sistema, ou externa (PE ou *exteroception*), quando aferem o ambiente envolvente, e ainda o seu modo de operação, distinguindo entre sensores ativos (A) e passivos (P).

Classificação	Exemplo	PI / PE	Modo
Táteis	Interruptores	PE	P
	Sensores Óticos	PE	A
Motores/Eixo	Encoders	PI	A/P
	Potenciómetros	PI	P
Direção	Magnetómetro	PE	P
	Giroscópio	PI	P
	Inclinómetro	PE	A/P
Baseados em <i>Beacons</i>	GPS	PE	A
	<i>Beacons</i> RF	PE	A
	<i>Beacons</i> Ultrassónicos	PE	A
	<i>Beacons</i> Óticos	PE	A
Medição de Distância	RADAR	PE	A/P
	Sonar	PE	A
	Laser	PE	A
	Câmara	PE	P
Velocidade/Aceleração	Acelerómetro	PE	P
	RADAR Doppler	PE	A
	Sonar Doppler	PE	A
	Câmara	PE	P
Identificação	RFID	PE	A
	RADAR	PE	A/P
	Câmara	PE	P
	Laser	PE	A

TABELA I.1: Classificação e Exemplos de Sensores

Fonte: Adaptado de (Siciliano e Khatib 2016)



## Anexo II - Referencial LIDAR

A Figura II.1 ilustra o referencial para as medições, apresentado no *datasheet* do LIDAR.

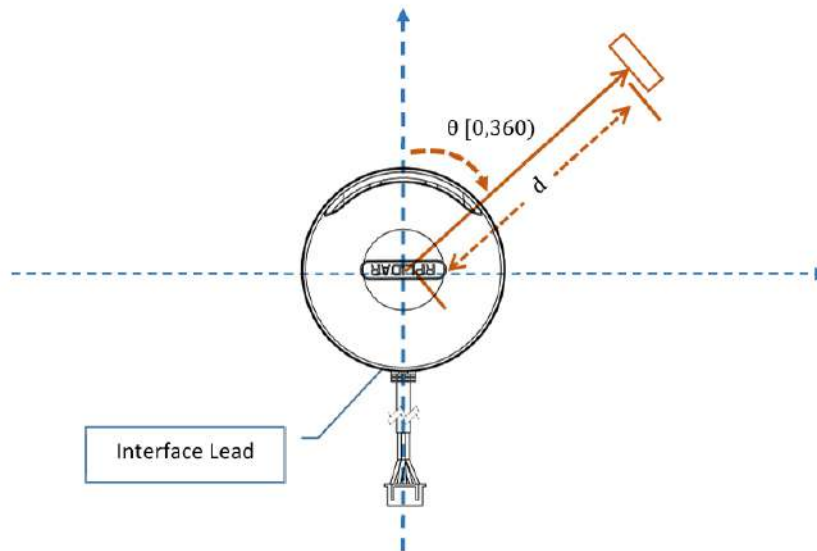


FIGURA II.1: Referencial Lidar  
Fonte: SLAMTEC rplidar datasheet A3M1



## Anexo III - Comparação de Trajetórias

Este anexo reúne figuras correspondentes às trajetórias estimadas nos diferentes ensaios realizados, servindo de complemento visual à análise apresentada no corpo do documento.

### III.1 Trajeto 1

A Figura III.1 apresenta as trajetórias estimadas para o Trajeto 1 com recurso ao filtro ESKF, com base na propagação da IMU e nas correções fornecidas por sensores adicionais. Estão representadas as trajetórias resultantes da fusão entre a IMU e cada um dos sensores individualmente (odometria visual, LIDAR ou rodas), a combinação simultânea dos três sensores (IMU+ODO+LIDAR) e por fim, a combinação de todos os sensores.

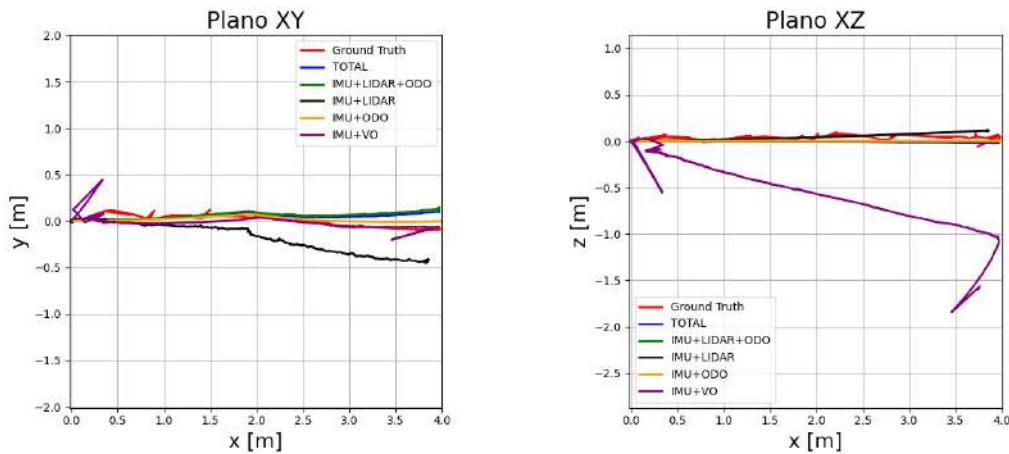


FIGURA III.1: Trajeto 1 — Estimação da pose com o filtro ESKF e diferentes sensores de correção

### III.2 Trajeto 2

A Figura III.2 apresenta as trajetórias estimadas pelos diferentes métodos aplicados de forma isolada no Trajeto 2.

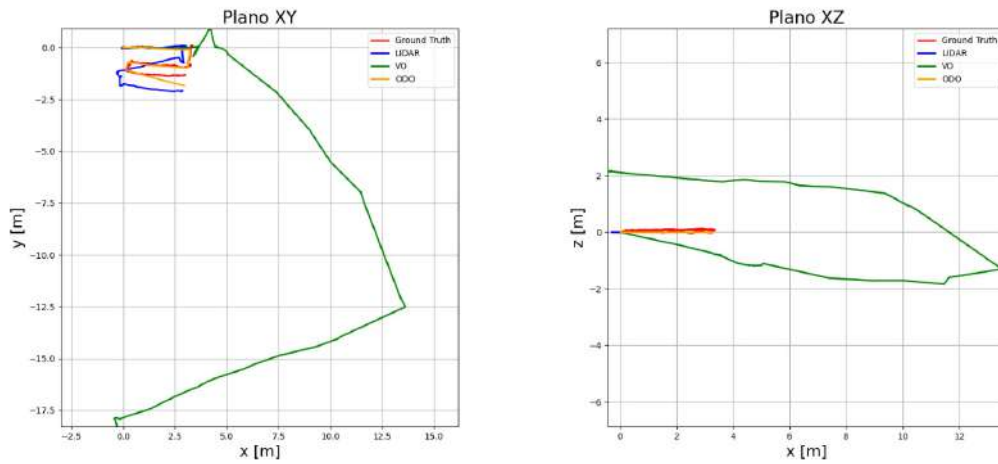


FIGURA III.2: Trajeto 2 — Estimação da pose com os métodos isolados

A Figura III.3 apresenta a trajetória estimada para o Trajeto 2, resultante da aplicação do filtro ESKF com a combinação da informação da IMU e odometria visual.

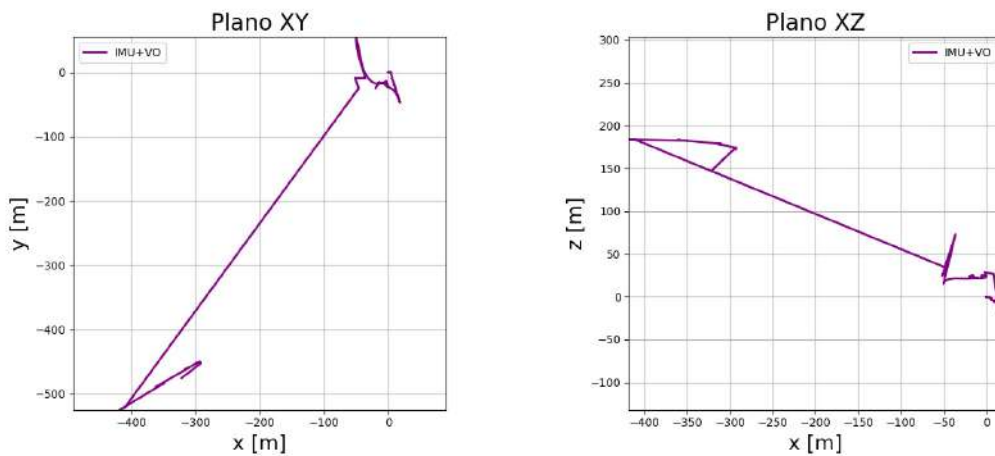


FIGURA III.3: Trajeto 2 — Estimativa da pose com a combinação IMU+VO

### III.3 Trajeto 3

A Figura III.4 apresenta as trajetórias estimadas pelos diferentes métodos aplicados de forma isolada no Trajeto 3.

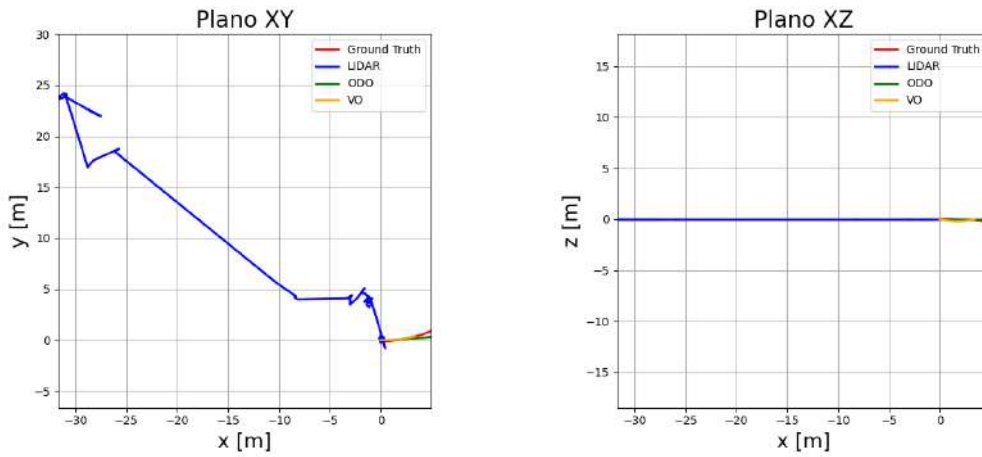


FIGURA III.4: Trajeto 3 — Estimação da pose com os métodos isolados

A Figura III.5 apresenta as trajetórias estimadas pelo método de odometria de rodas e odometria visual no Trajeto 3.

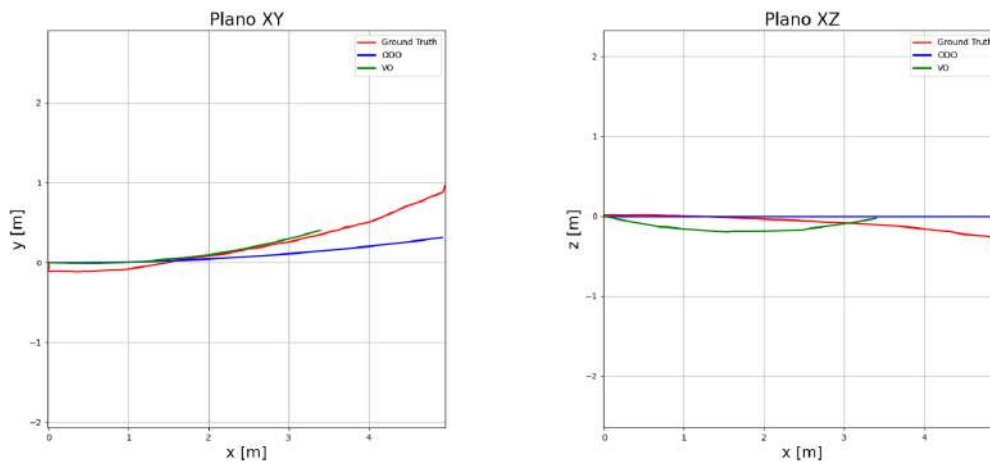


FIGURA III.5: Trajeto 3 — Estimação da pose com os métodos isolados, sem distorção introduzida pelo LIDAR

### III.4 Trajeto 5

A Figura III.6 apresenta as estimativas de posição obtidas pelos métodos de odometria visual, LIDAR e rodas, aplicados isoladamente, ao longo do Trajeto 5.

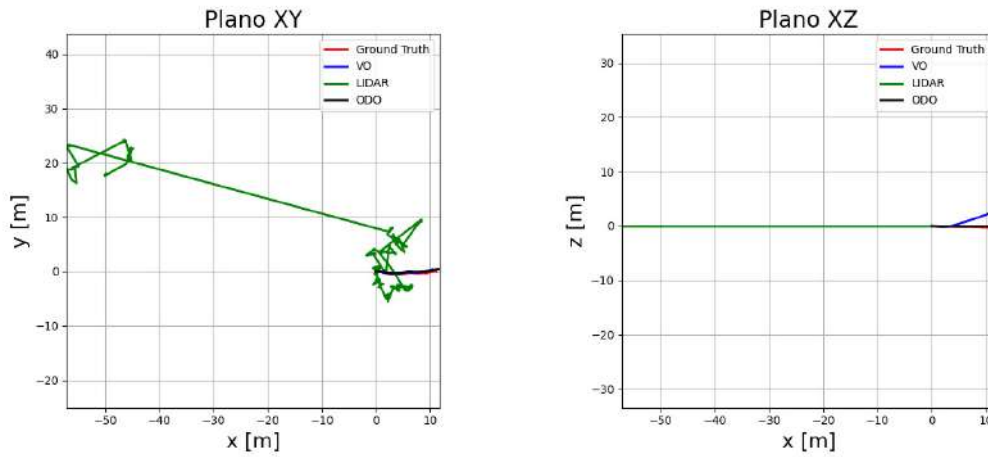


FIGURA III.6: Estimativas com métodos isolados no Trajeto 5

A Figura III.7 apresenta a comparação entre a estimativa obtida pela odometria visual e odometria tradicional.

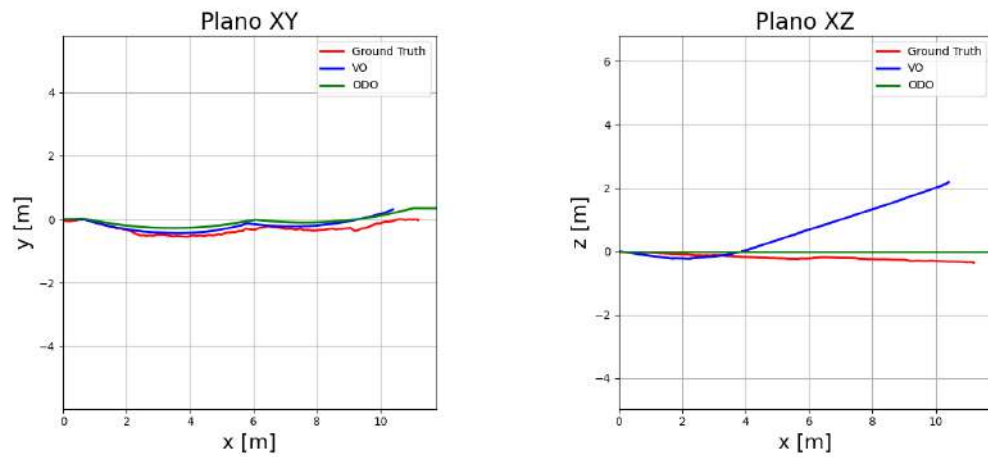


FIGURA III.7: Estimativas com métodos isolados no Trajeto 5