UNIVERSITY OF TURKU

Survey and Implementation of Passive SLAM Techniques for Natural Environments

University of Turku
Department of Computing
Master of Science (Tech) Thesis
Robotics and Autonomous Systems
Data Analytics Lab
July 2025

Author:
Vasista Kodumagulla

Supervisors:
Dr. Luca Zelioli
M.Sc. Adrian Borzyszkowski
Professor Jukka Heikkonen

UNIVERSITY OF TURKU
Department of Computing

VASISTA KODUMAGULLA: Survey and Implementation of Passive SLAM Techniques for Natural Environments

Master of Science (Tech) Thesis, 72 p.
Robotics and Autonomous Systems
Data Analytics Lab
July 2025

Autonomous robots, self-driving cars and delivery drones which are viewed as future technological advancements, depend on a specialized class of framework known as Simultaneous Localization and Mapping (SLAM) to enable autonomous navigation. SLAM helps an autonomous vehicle to chart a course through unknown terrain by building a map in real time from features detected by its onboard sensors. However, these frameworks remain under continual refinement to achieve true autonomy, as sensor limitations and computational constraints can occasionally limit the system's capability to capture the critical details from the environment. The challenge becomes even more pronounced when relying solely on passive sensors such as cameras and inertial measurement units (IMUs). Sometimes, these sensors have to be set in power restricted settings, or stealth mode applications, where active sensing could cause harm to living creatures.

This thesis aims to present an extensive study of the diverse sensor modalities employed in mobile robotics for navigation in rural environments, including forested and natural terrains. It encompasses advances in SLAM methodologies that incorporate both passive sensors (e.g. stereo cameras, IMUs, monocular cameras and RGB sensors) and active sensors (e.g. 2D LiDAR, 3D LiDAR and ultrasonic sensors). We then implement the leading passive-based SLAM approaches like ORB-SLAM2 and ORB-SLAM3 to evaluate their performance under realistic conditions in the forest environment. The results indicate that while the average Absolute Trajectory Error (ATE) in indoor environments remains within approximately 10 cm, it increases significantly to 20–30 meters in forest scenarios, demonstrating a clear performance gap. Despite strong reputation, these SLAM algorithms show room for improvement. Choosing better sensors can further enhance their performance. This thesis proposes future research directions, highlighting the potential of multisensor fusion. In particular, integrating stereo cameras and thermal imaging with learning-based SLAM frameworks can achieve more reliable and resilient mapping in natural environments.

Keywords: Autonomous Navigation, Stealth Mode Applications, Forest Navigation, ORB-SLAM, Multisensor Fusion.

Preface

I would like to express my deepest gratitude to Professor Jukka Heikkonen, Dr. Paavo Nevalainen, and my kind supervisors, Dr. Luca Zelioli and M.Sc. Adrian Borzyszkowski, for their continuous support, supervision, and invaluable assistance throughout my thesis. Their expert guidance helped me navigate the intricate details of this research.

I am also grateful to Mr. Timo Vasankari for connecting me with Professor Heikkonen and helping initiate this thesis opportunity.

I remain forever indebted to the University of Turku for granting me the privilege to study at a world-class institution under a scholarship program.

Furthermore, I would like to thank my friends and co-workers for their invaluable support in my ups and downs and their participation in intellectually stimulating discussions that enriched the quality of this thesis.

Lastly, I extend my sincere gratitude to my cherished family for their steadfast encouragement. Their unwavering faith in my academic pursuits has been a wellspring of strength and motivation.

15.07.2025

Vasista Kodumagulla

Contents

List of Figures

List of Tables

List of Acronyms

ATE – Absolute Trajectory Error

BA – Bundle Adjustment

BRIEF – Binary Robust Independent Elementary Features

CPU – Central Processing Unit

DBoW2 – Bag of Binary Words (version 2)

DSO – Direct Sparse Odometry

DSM – Direct Sparse Mapping

EKF – Extended Kalman Filter

FAST – Features from Accelerated Segment Test

GNSS – Global Navigation Satellite System

GPS – Global Positioning System

HDR – High Dynamic Range

IMU – Inertial Measurement Unit

KF – Kalman Filter

KITTI – Karlsruhe Institute of Technology & Toyota Technological Institute

LiDAR – Light Detection and Ranging

LOAM – LiDAR Odometry And Mapping

MAV – Micro Aerial Vehicle

ORB – Oriented FAST and Rotated BRIEF

PF – Particle Filter

PnP – Perspective-n-Point problem

PTAM – Parallel Tracking And Mapping

RGB – Red–Green–Blue colour space

RGB-D – RGB camera with per-pixel Depth

RMSE – Root Mean Square Error

RPE – Relative Pose Error

RTE – Relative Translation Error

SIFT – Scale-Invariant Feature Transform

SLAM – Simultaneous Localization And Mapping

TUM RGB-D dataset – Technische Universität München RGB-D dataset

UGVs – Unmanned ground vehicles

VI – Visual-Inertial

VO – Visual Odometry

1 Introduction

In recent years, the ability of autonomous robots to navigate unknown and unstructured environments has become increasingly critical in a wide range of fields. These include environmental monitoring, precision agriculture, search-and-rescue, defense operations and border surveillance, enabling autonomous patrolling, intrusion detection, and real-time threat assessment in sensitive or remote areas. These operations often unfold in conditions where GPS signals are unreliable or absent, such as under dense forest canopies, inside collapsed buildings, or on extraterrestrial terrains, where maps which are built are often outdated or too coarse to support meaningful interaction. In such scenarios, autonomous agents must rely on their onboard sensors to perceive, interpret, and make real-time decisions about their environment [1]. This operational independence is made possible by the use of SLAM, a computational framework that allows a robot to construct a map of its environment while concurrently estimating its trajectory within that map [2]. SLAM has emerged as a foundational technology in modern robotics, enabling autonomy in both controlled indoor environments and, increasingly, in complex outdoor scenarios [3]. SLAM research has achieved impressive maturity in urban and indoor settings. This is due to high-contrast textures, planar surfaces, and stable lighting.However, the deployment of SLAM in visually cluttered and dynamic natural environments such as forests remains relatively underdeveloped. Forest environments consists of unique challenges, such as self-similar textures, rapid lighting fluctuations, seasonal transformations,

and environmental dynamics such as wind-blown foliage or animal movement [4]. These factors not only degrade the performance of standard SLAM algorithms but also strain the assumptions made in sensor fusion and loop closure modules. Furthermore, concerns over energy consumption, sensor cost, and ecological impact necessitate the use of lightweight, low-power, and non-invasive sensing modalities, conditions ideally suited for passive visual sensors such as monocular and stereo cameras [5]. Motivated by these constraints and opportunities, this thesis explores the robustness, accuracy, and practical trade-offs of passive, vision-based SLAM systems in forest environments. Through empirical evaluations on benchmark datasets and on-site observations, the study aims to identify the limitations of current SLAM pipelines and suggest improvements tailored to natural, unstructured settings.

1.1 Background and Motivation

In the era of autonomous technologies, the capability of robots to navigate without a GPS signal in hostile environments is no longer a mere research ambition, but a strategic necessity. Autonomous navigation has found indispensable utility across various sectors, including large scale farming, environmental surveillance, disaster relief, planetary exploration, and modern defense operations. Each of these applications demands better situational awareness in settings where external location aids, such as GPS, may be unavailable, unreliable, or actively denied [6]. In this context, SLAM has emerged as a cornerstone technology [7], [8]. SLAM enables autonomous navigation in unknown environments without reliance on pre-built maps or external localization infrastructure. Early robotic navigation relied on methods like dead reckoning and beacon-based systems, which suffered from cumulative drift and scalability issues. The advent of SLAM revolutionized autonomous robotics by allowing robots to construct maps of their surroundings while localizing themselves within them [9]. Visual SLAM approaches gained popularity due to their compact

sensor footprint and cost-effectiveness, leveraging image features, pose estimation algorithms, and loop closure techniques to maintain map consistency and reduce drift over time [10]. SLAM has shown considerable success in controlled indoor scenarios and semi-structured urban landscapes. Its deployment in complex, unstructured, and dynamic outdoor environments like forests, mountainous terrain, or post-conflict zones is still under research[11], [12].

1.1.1 The need for SLAM and its importance in Robotics

The achievement of complete autonomy in robotic systems is challenging even under the best of circumstances, and the difficulty increases in natural and unbound settings [8]. Forested environments, in particular, pose a unique set of challenges for SLAM that can potentially hinder feature matching. The presence of dynamic elements, and severe GPS attenuation or outright signal loss under dense vegetation makes it challenging[13], [14]. From a defense perspective, the implications of these challenges are profound. Consider a surveillance drone tasked with silently mapping a forest during a search-and-rescue mission or monitoring enemy movements under the canopy. The drone must rely entirely on its onboard sensors, operating with minimal energy consumption and remaining undetected by avoiding active signal emissions [15]. Unlike static maps, SLAM systems dynamically incorporate new environmental features and re-optimize when revisiting locations, maintaining global consistency [16]. Loop closure, a hallmark of SLAM, refers to the recognition of previously visited locations to correct accumulated drift in the estimated trajectory, thereby enabling long-term autonomy in challenging environments [17]. The evolution from early monocular implementations to modern visual-inertial systems [18] reflects a growing emphasis on their performance in complex real-world environments.

The strategic applications of SLAM in defense extend far beyond patrolling.

UGVs operating in subterranean tunnels, semi-autonomous patrol robots navigating forest perimeters, and surveillance military drones in adversarial territories all require SLAM systems to function effectively where GPS signals are jammed, denied, or spoofed. DARPA's subterranean challenge, for example, emphasized the importance of mapping and localization in visually degraded and GPS-denied environments [19]. This has lead to advanced SLAM architectures suitable for forest search-and-rescue, underground warfare, and disaster response scenarios [20]. These scenarios mirror real-world defense needs where troops may need to deploy robotics in caves, tunnels, or dense jungles for surveillance or rescue operations. SLAM's role also extends to tactical autonomy in missions involving swarms of unmanned aerial vehicles (UAVs). These swarms rely on decentralized SLAM approaches to build shared maps of terrain in real-time while maintaining relative localization among agents [21]. The European Defence Agency (EDA) and NATO ACT (Allied Command Transformation) have outlined the need for resilient, vision-based SLAM systems [22] for use in robotic soldiers, aerial reconnaissance, logistics convoys, and autonomous border surveillance. These systems are expected to operate covertly in GPS-contested areas day and night in adversarial environments without compromising positional awareness.

1.1.2 Motivation

The SLAM research community has historically focused on structured indoor environments (e.g., warehouses, laboratories) or well-mapped urban areas supported by Visual data, LiDAR, and GPS. Benchmark datasets like KITTI, EuRoC, and TUM RGB-D have driven algorithmic progress in these domains [23]. However, these environments are not representative of natural or tactical outdoor deployments. This thesis addresses this critical gap by investigating the deployment of passive, vision-based SLAM systems, specifically ORB-SLAM2 and ORB-SLAM3, in forested envi-

ronments characterized by dynamic, cluttered, less featured, and low-textured visual conditions. It contributes to academic research and defense applications by exploring the trade-offs between performance, computational load, energy consumption, and environmental impact in the context of passive outdoor SLAM [8], [11]. Despite advances in SLAM, deployment in unstructured outdoor environments such as forests remains limited due to the lack of distinct landmarks, which complicates reliable feature matching [24]. Environmental dynamics such as windblown foliage, moving wildlife, and changing sunlight, the typically irregular and non-planar terrain introduces non-rigid variations in the scene, necessitating dynamic scene modeling for consistent tracking [4]. Canopy density can cause frequent GPS signal loss highlighting the importance of autonomous visual localization in such settings, and the unpredicatable weather conditions further degrades the performance of traditional feature detection algorithms.

Moreover, SLAM's utility in military engineering and logistics, such as mapping terrain for infrastructure deployment (e.g., bridges, field hospitals), autonomously guiding supply drones through harsh weather, or assisting in explosive ordnance disposal, underscores its transformative impact. This transformation lies in the ability of SLAM to enable autonomous operations in GPS-denied or high-risk environments, reduce dependence on manual mapping and navigation, and improve both the speed and safety of mission-critical tasks. By shifting the operational paradigm from reactive to proactive and autonomous decision-making, SLAM fundamentally redefines how military field operations are executed. Passive visual sensors, especially stereo and monocular cameras, provide lightweight and energy-efficient alternatives to active sensing technologies such as LiDAR and radar. A detailed comparison of the advantages and limitations of passive and active sensing modalities is presented in Section 2.1. The motivation for this thesis stems from the ecological, operational, and economic advantages of such sensors. Visual SLAM avoids disturbing wildlife,

consumes less energy, and is better suited for prolonged autonomous missions in remote areas. By focusing on vision-based methods, this study lays the foundation for developing SLAM pipelines that are scalable and accurate, ecologically safe, deployable in hostile terrain, and operational on computationally limited platforms commonly used in tactical robotics. The knowledge gained from the evaluation of SLAM in forests has a broader applicability to other complex, dynamic, and GPS-challenged terrains, including urban ruins. These insights contribute to developing adaptable SLAM systems that supports spectrum of civil and military domains. The following research questions guide this investigation.

1.2 Research Questions

SLAM algorithms have matured significantly in the context of indoor settings, warehouse navigation, and urban driving scenarios [8], [25], their reliability in outdoor terrains, such as forests, remains an open research challenge. Forest environments are dynamic, irregular, and have unreliable GPS coverage [11], [13]. This thesis aims to address these gaps by evaluating passive SLAM systems, specifically vision-based methods, under such real-world constraints.

The core research questions that guide this study are elaborated as follows:

- **RQ1:** How robust are state-of-the-art visual SLAM algorithms (e.g. ORB-SLAM2, ORB-SLAM3) in natural forest environments where dense vegetation, varying illumination, seasonal conditions, and dynamic elements can disrupt feature tracking and mapping. Do they work well or poorly in the outdoors compared to feature-rich indoors?

- **RQ2:** What algorithmic adaptations are necessary to enable real-time SLAM in such demanding environments?

- **RQ3:** How do different passive sensors (e.g. stereo, monocular) cope with the challenges posed by dense foliage, dynamic elements, and texture homogeneity?

- **RQ4:** To what extent can low-cost, low-power passive SLAM systems match the performance of more expensive active counterparts (e.g., LiDAR) regarding loop closure and localization drift over the long term?

SLAM algorithms like ORB-SLAM2 [26] and ORB-SLAM3 [23] have shown remarkable performance in various benchmark datasets which are usually collected in controlled or semi-structured settings with stable lighting and clear features. However, their performance in natural environments such as forests, which exhibit variable illumination due to canopy cover, shadows, and changing weather is unknown [12], [14]. Moreover, seasonal variations alters the visual landscape, demanding a SLAM system capable of long-term adaptation and resilience. SLAM systems are computationally intensive, particularly when performing front-end feature extraction, real-time bundle adjustment, and global graph optimization [27]. Forest robots or aerial drones often rely on embedded CPUs with limited computational power and battery capacity, making real-time performance a non-trivial objective. These research questions investigate how frame-rates , simplified loop-closure heuristics, or reduced keypoint densities affect mapping accuracy and localization drift in these SLAM frameworks [23], [26], [28].

Passive sensors can be used due to low power consumption, minimal ecological impact, and small size. However, their performance may degrade under conditions like cold, snow, and dense forests. Difficult environments, such as leaves and bark, vegetation occlusions, and fluctuating lighting, can diminish the reliability of keypoint detection and matching [11], [12], [29]. Monocular SLAM struggles with absolute scale estimation, whereas stereo systems, while more robust, increase the computational burden. Specifically, RQ3 aims to empirically compare these configu-

rations using real-world datasets (e.g., FinnForest, TUM RGBD) and quantify their resilience to these environmental factors, complementing simulation-heavy studies with field-based validation. LiDAR-based SLAM systems described by Zhang et al. [30] offer high-precision depth information, but at the expense of energy, cost, and weight. Visual SLAM alternatives, when properly tuned, offer a promising trade-off between performance and deployability, especially for small ground robots or UAVs used in remote environments [31], [32]. The literature has documented success in using cameras for urban and indoor SLAM, but their capabilities in complex natural environments remain to be rigorously benchmarked. RQ4 address the core limitations of deploying camera-centric SLAM in ecologically sensitive, sensor-hostile, and computationally restricted environments. By systematically analyzing performance under real-world constraints and environmental variability, this thesis contributes to the practical deployment of autonomous mapping systems for outdoor field robotics, pushing the frontier of robotic perception beyond urban and indoor domains.

1.3 Scope and Structural Outline of the Thesis

The study focuses on three core objectives. It aims to evaluate the precision and robustness of stereo-based SLAM in real-world forest conditions, to analyze the trade-offs between computational efficiency and mapping performance, and propose design considerations to enhance SLAM reliability in unstructured natural settings. The thesis is organized into six chapters. Chapter 1 introduces the research context, outlines motivation, and presents key objectives. Chapter 2 reviews the relevant literature, classifying SLAM approaches into filter-based, graph-based, and learning-augmented categories. It also examines the characteristics of passive, active, and hybrid sensor modalities, particularly in autonomous outdoor navigation. Then provides a detailed technical overview of the ORB-SLAM2 and ORB-SLAM3 systems, with emphasis on their architectural components such as feature extrac-

tion, pose estimation, keyframe management, and loop closure. Special attention is paid to the enhancements introduced in ORB-SLAM3, including multimap support and visual-inertial integration. Chapter 3 describes the experimental methodology, including the data sets employed, the evaluation metrics, and the system configurations. Chapter 4 presents and analyzes the empirical results obtained from both controlled indoor datasets and field-collected forest data. Chapter 5 summarizes the main findings, discusses observed limitations, and outlines directions for future research, including the potential integration of semantic and thermal data and the expansion of the evaluation datasets. Finally, Chapter 6 concludes the thesis by summarizing the key takeaways of this research study.

2 Literature review

In this chapter, a comprehensive literature review is presented to explore the development, current state, and application of SLAM technologies. The chapter is structured as follows. Section 2.1 introduces the variety of sensors used in SLAM systems, distinguishing between active and passive modalities. Section 2.2 delves into the algorithmic evolution of SLAM, limitations, and challenges present in state-of-the-art SLAM frameworks, and proposes directions for this research. Section 2.3 covers the detailed architectures and working of ORB-SLAM2 and ORB-SLAM3.

2.1 Survey on Sensors

SLAM systems fundamentally rely on sensory input to estimate both the robot trajectory and the environmental structure. The choice of sensor significantly influences system performance, computational load, and applicability to specific environments.

2.1.1 Active Sensors

Active sensors are those that actively emit energy, such as lasers, sound waves, or modulated infrared light, and interpret the reflected signal to estimate environmental features like depth, structure, or range. These sensors play a crucial role in many SLAM systems, particularly where passive sensors may struggle due to low texture, poor lighting, or the need for fine-grained geometric data. Active sensors, such as

LiDAR, radar, and structured-light cameras (e.g., Kinect), emit energy into the environment and measure its reflection to estimate distances or depth. LiDAR (Light Detection and Ranging), in particular, has been a cornerstone of high-precision mapping in robotics due to its ability to generate accurate 3D point clouds in a wide variety of lighting conditions [30]. Radar systems, though less common, offer good performance in adverse weather conditions (e.g., fog or rain), making them valuable in autonomous driving scenarios. Numerous SLAM implementations leverage active sensors to achieve high accuracy in structured and semi-structured settings. For instance, LOAM (Lidar Odometry and Mapping) Zhang et al. [30] demonstrated real-time, low-drift localization and mapping capabilities, setting a benchmark for LiDAR-based SLAM systems. Similarly, radar-based SLAM has been explored for subterranean and high-speed navigation, where optical sensors may fail [33]. Use cases of active sensors extend to autonomous vehicles navigating in low-light urban environments, inspection robots in underground tunnels, and mining robots operating in harsh, GPS-denied spaces. Moreover, aerial drones equipped with LiDAR are used for topographical surveys in forestry and agriculture. These applications underline the robustness of active sensors across domains that require high spatial accuracy. Despite their advantages, active sensors suffer from limitations that restrict their utility in certain environments. They are often expensive, consume significant power, and may pose ecological concerns when deployed in wildlife-rich areas, such as forests. Below, we outline key categories of active sensors commonly integrated into SLAM frameworks, along with their working principles, advantages, and limitations.

SLAM frameworks such as LOAM [30], Cartographer [34], and LeGO-LOAM [35] rely heavily on LiDAR for robust mapping. Radar (Radio Detection and Ranging) operates by transmitting radio waves and analyzing their reflections to infer object positions. Unlike LiDAR, radar performs reliably in poor visibility conditions such as

fog, smoke, and dust. This reliability makes radar invaluable in SLAM applications, including patrolling the battlefield and long-range threat detection [36]. However, radar suffers from lower resolution and difficulty distinguishing fine features, which limits its utility in scenarios requiring high-precision mapping. Additionally, since radar systems emit detectable radio waves, they can potentially reveal the operator's location to adversaries. This compromises stealth, undermining the primary objective in covert or low-profile missions. Sonar and Ultrasonic Sensors utilize sound waves to estimate distances by measuring the time-of-flight of echoes. These are particularly useful in underwater SLAM or low-cost terrestrial robots operating in short-range settings [37]. While inexpensive and power-efficient, sonar suffers from limitations including slow update rates, beam spreading, and poor performance in complex geometries due to specular reflection.

Structured Light Sensors, such as the Microsoft Kinect v1, emit a known infrared pattern onto a scene and observe distortions to compute depth. This approach is beneficial for indoor SLAM tasks and human-robot interaction. The main advantages are real-time depth estimation and affordability. However, performance degrades significantly in outdoor environments due to ambient sunlight interference. Time-of-Flight (ToF) cameras emit modulated IR light and measure phase shifts or time-of-flight to create depth maps. These sensors are compact and provide depth information at video frame rates, supporting real-time SLAM in confined spaces. Despite these benefits, ToF cameras have limited range (usually under 10 meters), lower resolution, and sensitivity to bright sunlight [38]. 2D Laser Rangefinders represent an early form of LiDAR that scans a single horizontal plane. Although they cannot capture a full 3D structure, these rangefinders are well-suited for flat indoor environments and mobile ground robots [39]. Limitations include a lack of vertical information and challenges with uneven terrain. In sum, active sensors are indispensable in SLAM deployments requiring precision, resilience to lighting conditions,

or operation in visually degraded environments. Nevertheless, their higher power demands, complexity, and visibility to adversaries (in defense contexts) must be carefully considered. In stealth-critical operations, these drawbacks can pose security and logistical risks. Consequently, while active sensing technologies remain indispensable in defense, industrial, and subterranean SLAM, they are often complemented or replaced by passive sensors in lightweight, energy-constrained, or covert operations.

2.1.2 Passive Sensors

Passive sensors, such as monocular, stereo, thermal, or microphone sensors or event-based cameras, rely on ambient light or naturally occurring environmental signals. These sensors are typically low-cost, energy-efficient, and minimally invasive, attributes that make them ideal for long-term field deployment and ecological studies. Visual SLAM (vSLAM) based on passive sensors has gained significant traction due to its applicability in GPS-denied environments and its ability to work with compact mobile platforms. ORB-SLAM [31], ORB-SLAM2 [26], and ORB-SLAM3 [23] exemplify passive visual SLAM pipelines that support monocular, stereo, and RGB-D configurations with the possibility to fuse with other passive modalities. Use cases of passive sensors include consumer-grade drones for photogrammetric mapping, agricultural robots performing crop monitoring, and autonomous lawnmowers navigating domestic gardens. Thermal and event-based cameras have also been employed in surveillance and wildlife tracking, where unobtrusive observation is essential. In space exploration, visual SLAM using stereo cameras has enabled rover navigation on Mars, where energy efficiency is paramount. However, passive sensors are highly susceptible to environmental factors such as illumination changes, motion blur, and scene dynamics. In forested environments, these limitations become particularly pronounced and can confuse feature-based detectors. Nevertheless, passive sensors

remain an attractive choice due to their stealth, affordability, and deployment flexibility. This study builds upon these advantages while seeking to overcome their inherent weaknesses through better algorithmic design and dataset-specific adaptations. Unlike active sensors, passive sensors are well-suited for energy-constrained environments and lightweight platforms. The following types of passive sensors are commonly used in SLAM applications.

Monocular cameras use a single RGB or grayscale lens to capture 2D images of a scene. Feature-based methods (e.g., PTAM, ORB-SLAM) track keypoints across frames to estimate motion, while direct methods (e.g., DSO, LSD-SLAM) use raw pixel intensities to compute camera pose. Despite their wide adoption, monocular SLAM suffers from scale ambiguity, lack of depth information, and sensitivity to environmental changes [31]. These limitations can be partially mitigated by fusing monocular input with inertial measurements or known object sizes. Stereo cameras consist of two or more synchronized lenses with a known baseline, enabling depth estimation through image disparity. They offer a significant advantage over monocular setups by directly inferring scale and depth, making them better suited for unstructured or low-texture environments. Systems like ORB-SLAM2 and ORB-SLAM3 efficiently exploit stereo inputs for improved robustness and loop closure [26]. The primary trade-offs include higher weight, cost, and the need for meticulous calibration and synchronization. Event-based cameras such as the DAVIS or Prophesee sensors detect changes in pixel intensity asynchronously, offering microsecond-level temporal resolution and high dynamic range. They are particularly useful in high-speed or low-light environments where standard frame-based cameras fail. Event-based SLAM approaches like UltimateSLAM and EVIO demonstrate resilience to motion blur and abrupt illumination shifts [40]. However, these systems require novel algorithms and preprocessing pipelines due to the unconventional nature of event data. Thermal cameras capture long-wave infrared radiation emitted by objects, generat-

ing heat maps that represent temperature variations. They are particularly effective in total darkness or through obscurants such as smoke and fog. While thermal cameras offer unique advantages in nighttime operations and search-and-rescue SLAM [41], they also present challenges, including lower spatial resolution, high noise, and inconsistent thermal signatures. Preprocessing techniques and feature adaptation are often necessary to utilize thermal data effectively in SLAM. Additional passive sensors often enhance SLAM performance when fused with visual data.

In summary, passive sensors provide lightweight and efficient alternatives to active sensors in SLAM, particularly when size, cost, stealth, or energy constraints are paramount. While they often lack direct depth measurement capabilities, the fusion of multiple passive modalities, particularly visual and inertial, enables high-accuracy SLAM in diverse scenarios. Their increasing role in field robotics, mobile augmented reality, and military operations underscores their strategic value in both research and deployment contexts.

2.2 Overview of Different SLAM Approaches

This section offers a structured taxonomy of the prevailing SLAM methodologies by building on foundational frameworks outlined by Thrun et al. [42], Durrant-Whyte and Bailey [7], and Cadena et al. [8]. These taxonomies serve to clarify how SLAM has evolved across multiple algorithmic paradigms, including filter-based estimation, graph-based optimization, and learning-enhanced pipelines. Additionally, explored how different sensor modalities, whether active or passive, interface with these paradigms to meet varying environmental and computational challenges. Particular attention is paid to the implications of sensor choice in natural and forested settings, where repetitive textures, dynamic illumination, and occlusions place extreme demands on generalizability of SLAM methods.

2.2.1 Core Components of SLAM

To effectively analyze the diverse SLAM methodologies available today, it is essential to first understand the foundational architecture that underpins all SLAM systems. As illustrated in Figure 2.1, a standard SLAM framework typically consists of four key components, namely front-end, back-end, loop closure, and sensor fusion in modern implementations. These modules interact continuously to enable real-time localization and mapping in dynamic and complex environments. Each component plays a critical role in the accurate and efficient construction of a map and estimation of the camera or robot trajectory. Their coordinated operation becomes especially significant in complex environments such as forests, where visual ambiguity, dynamic elements, and motion-induced artifacts are prevalent [8], [27]. The front end is responsible for raw sensor processing and feature extraction. In feature-based systems such as ORB-SLAM2 and ORB-SLAM3, this involves detecting salient visual keypoints, typically corners or blobs, and computing descriptors for efficient inter-frame matching. The ORB descriptor (Oriented FAST and Rotated BRIEF) [43] is widely adopted due to its computational efficiency and robustness to scale and rotation, making it suitable for embedded and real-time applications. While alternatives like BRISK and AKAZE exist, ORB remains the baseline for real-time keypoint matching due to its balance of speed and accuracy [31].

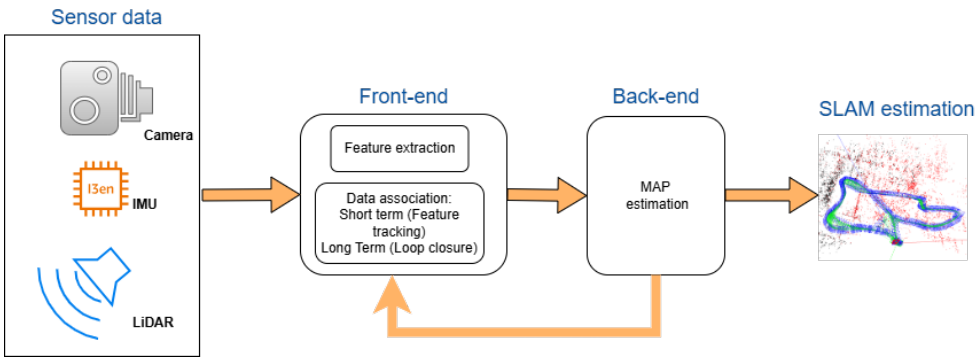

Figure 2.1: General architectural diagram of SLAM and its core components.

The back-end, depicted in the same figure, constructs and optimizes a graph of poses and landmarks using observations from the front-end. This is often modeled as a nonlinear least squares problem where bundle adjustment refines both camera poses and 3D landmark positions. Graph-based SLAM frameworks, such as those employing g2o or GTSAM, have emerged as the dominant approach due to their scalability and better accuracy compared to earlier EKF or particle filter-based methods [42], [44]. In ORB-SLAM2 and ORB-SLAM3, both local and global bundle adjustment are employed. These routines jointly optimize camera poses and map points to minimize reprojection error, maintaining global consistency during long-term operation. This optimization is particularly crucial in natural environments where sensor noise, motion blur, and non-rigid scene changes can introduce significant drift over time. Loop closure is another key component of SLAM that identifies previously visited locations and corrects accumulated drift. ORB-SLAM employs a bag-of-words (BoW) place recognition method using DBoW2 [45], which converts ORB descriptors into visual words and queries a database for likely matches. Upon geometric verification, loop constraints are added to the pose graph and corrected through global optimization. In ORB-SLAM3, this mechanism is further enhanced with improved geometric validation and support for map merging across multiple sessions. Forest environments pose unique challenges for loop closure that can lead to both false positives and missed detections, undermining the system's long-term consistency. As such, better loop detection under these challenging conditions becomes a crucial criterion for SLAM evaluation. Sensor fusion, the fourth pillar of modern SLAM architectures, becomes particularly valuable in degraded visual conditions. ORB-SLAM3 integrates tightly coupled inertial data through pre-integration techniques, allowing the system to maintain tracking even when visual input is poor or temporarily unavailable. For example, when branches occlude the camera or sunlight causes overexposure, the IMU provides a short-term motion prior that sustains

pose estimation until visual tracking resumes [23], [46]. Proper integration requires synchronization, calibration, and bias modeling for a better performance in natural terrain.

Active SLAM systems like LOAM [30] and Cartographer [34] exemplify the strength of depth-first sensing through LiDAR, which simplifies spatial reconstruction but is often unsuitable for energy-constrained or stealth-critical applications. Graph-based SLAM algorithms dominate such systems, exploiting non-linear least squares optimization to deliver globally consistent maps. Passive SLAM techniques, including monocular and stereo approaches like ORB-SLAM2 [26] and ORB-SLAM3 [23], necessitate sophisticated feature extraction and pose estimation pipelines due to the indirect nature of depth computation. These are often supported by visual-inertial fusion methods, as seen in VINS-Mono [32] and DSO [47].

Deploying SLAM in visually ambiguous environments demands selection of appropriate algorithms and sensor configurations since these situations often results in unreliable feature extraction and tracking. Hence, these difficulties have motivated the growing interest in hybrid sensor integration strategies and the development of adaptable feature representations, including those derived from deep learning techniques, to enhance SLAM performance under such adverse conditions. Despite the promise of learning-augmented SLAM systems, they remain brittle due to their dependence on extensive training data and high computational demands. By reviewing these diverse approaches and their sensor dependencies, this thesis sets the stage for a rigorous empirical assessment of passive SLAM performance.

2.2.2 Algorithmic Paradigm

SLAM algorithms have evolved significantly over the past two decades, transitioning from probabilistic filtering approaches to graph optimization and, more recently, learning-based techniques. Early efforts, such as EKF-based SLAM, provided a

foundational probabilistic framework for recursive state estimation [48]. Despite their elegance, EKF-based methods struggle with scalability and suffer from linearization errors, especially when deployed in large or complex environments. These limitations led to particle filter-based FastSLAM, which decouples the estimation of robot trajectory from landmark positions and improves scalability at the cost of increased computational complexity. Graph-based SLAM has since become a dominant paradigm, formulating the SLAM problem as a non-linear least squares optimization over a factor graph. This approach allows the decoupling of the front-end, responsible for data association and feature tracking, from the back-end optimization process [27].

Optimization libraries such as g2o [49] and GTSAM [50] enable robust real-time back-end optimization by minimizing the global reprojection error over the SLAM graph. They help in producing a trajectory and map consistent with all available measurements. Systems like ORB-SLAM and its successors exemplify this approach through keyframe-based tracking, local and global bundle adjustment, and loop closure using a bag-of-words model for place recognition [23], [31]. More recently, learning-based SLAM approaches have emerged, using deep learning to extract features, predict depth, estimate poses, and detect loop closure. Examples include DeepVO, DROID-SLAM, and Neural Bundle Adjustment frameworks. These models perform well on benchmark datasets like KITTI or TUM RGBD but exhibit poor generalization to natural environments such as forests, primarily because they are trained on urban scenes [51]. In addition, deep SLAM systems often require GPU acceleration and substantial memory resources, which limit their deployment on edge devices commonly used in field robotics. While recent advances in learning-based SLAM have demonstrated promising results in structured indoor and urban environments, their generalization to complex natural settings such as forests remains limited. Few deep learning methods have been explicitly trained or evaluated

on forest-specific datasets, and those that exist often struggle with occlusions, seasonal variations, and low-texture vegetation. For example, models such as DeepVO [52] and D3VO [53] have shown strong performance on benchmark datasets like KITTI and TUM RGBD, but their applicability in forested terrain is unproven due to the domain gap. Consequently, this thesis deliberately focuses on classical and graph-based SLAM methods, given their interpretability, hardware efficiency, and adaptability to novel environments where data scarcity and environmental variability are prevalent.

2.2.3 Active Sensor-Based SLAM

Depth data may usually be calculated by LiDAR-based sensors, which have proven indispensable in applications requiring high spatial precision and environmental adaptability. Developed by Google, Cartographer [34] is an open-source, graph-based SLAM library known for real-time performance, submap-based loop closures, and modularity. It supports both 2D and 3D LiDAR and integrates with IMU and odometry data. Cartographer has been widely deployed in autonomous mobile robots (AMRs), logistics automation, warehouse robots, and industrial indoor mapping due to its robustness and scalability. It has also seen applications in the mining and construction sectors, where dynamic obstacles and variable geometry pose significant challenges for SLAM systems. LOAM (LiDAR Odometry and Mapping) [30] initiated a modular SLAM approach that decouples odometry estimation from mapping. Processes 3D LiDAR data to track motion and reconstruct environments with high fidelity.

Recent deployments in large-scale outdoor SLAM, including forestry and agriculture monitoring, have also begun to experiment with lightweight LiDAR-based modules in hybrid sensor settings [54]. GMapping is a classical 2D SLAM algorithm based on Rao-Blackwellized Particle Filters [55]. It remains widely adopted due to

its integration with ROS and its efficacy in producing accurate occupancy grids from 2D LiDAR in structured indoor environments. Although limited to planar navigation, it is still a default option for many indoor service robots, vacuum cleaners, and academic platforms. Hector SLAM [56] offers a high-frequency scan-matching approach that does not depend on external odometry. This makes it valuable in scenarios where wheel odometry is unavailable or unreliable, such as in rescue robotics, rugged terrains, or uneven indoor floors. Its real-time capability and minimal hardware dependency make it a favorite for drone-based SLAM in tightly constrained spaces.

A-LOAM represents an open-source, modular adaptation of the original LOAM algorithm [30], offering a streamlined and accessible implementation frequently utilized in academic and experimental robotics research. While maintaining the two-threaded architecture of LOAM, separating motion estimation from mapping A-LOAM simplifies certain processing steps, thus facilitating integration into real-time robotics platforms for testing and prototyping. Although it does not support sensor fusion natively, its clarity and modularity make it ideal for algorithmic benchmarking and comparative analysis in structured environments. In studies such as [57], A-LOAM is used as a baseline to evaluate the impact of sensor fusion on SLAM performance across urban and semi-structured environments. LeGO-LOAM introduces significant adaptations of the LOAM framework specifically tailored for ground-based mobile robots operating in uneven outdoor terrains [35]. Its core innovation lies in segmenting the point cloud data into ground and non-ground points, allowing for efficient real-time performance even on computationally limited platforms. By utilizing this segmentation, LeGO-LOAM reduces computational overhead and enhances the robustness of mapping in scenarios with elevation variation, such as hillside paths or off-road environments. A practical deployment of LeGO-LOAM is illustrated in agricultural robotics, where autonomous tractors and ground ve-

hicles must navigate complex field geometries without reliance on GPS. In [58], LeGO-LOAM was integrated with crop row detection to support precision farming applications, highlighting its ability to perform accurate terrain mapping and navigation in visually and geometrically ambiguous environments.

LIO-SAM [59] is a state-of-the-art, tightly coupled LiDAR-Inertial SLAM algorithm that leverages factor graph optimization to integrate IMU measurements with LiDAR data in a computationally efficient manner. Unlike loosely coupled systems, LIO-SAM maintains continuous trajectory estimation accuracy by incorporating IMU pre-integration, real-time scan-to-submap matching, and optional GPS integration. The algorithm excels in high-dynamic motion scenarios or feature-sparse environments, where LiDAR or IMU alone may be insufficient. A notable use case is UAV-based urban surveying, where LIO-SAM has been employed for accurate 3D reconstruction in areas such as tunnels and narrow alleyways. In [60], the system demonstrated consistent mapping accuracy under aggressive aerial maneuvers, significantly outperforming traditional LiDAR-only pipelines by reducing drift and improving performance to occlusions. FAST-LIO [61] and its successor FAST-LIO2 [62] are designed for real-time, high-frequency SLAM on embedded robotic systems. These methods adopt an incremental Kalman filter approach to perform tightly coupled LiDAR-inertial odometry, achieving efficient state estimation while maintaining low latency. FAST-LIO2 further improves upon its predecessor by introducing a global map reuse strategy and enhanced robustness against degenerate motions. This makes it particularly suitable for deployment on high-speed platforms such as agile drones, where low latency and high frame-rate processing are essential for stable flight control and autonomous navigation. In the context of infrastructure inspection, FAST-LIO2 has been integrated into quadrotor UAVs tasked with mapping bridge undersides and building façades [62]. The system's low computational burden enabled onboard processing and real-time feedback, allowing human opera-

tors to conduct precise close-range inspection without relying on external processing units or GPS.

These algorithmic advancements reflect a broader trend in active SLAM toward domain-specific optimization, sensor fusion, and computational efficiency. While A-LOAM and LeGO-LOAM prioritize modularity and terrain awareness, respectively, the LIO-SAM and FAST-LIO series exemplify tightly coupled approaches optimized for aggressive maneuvers and embedded systems. Their integration across domains, from agriculture to aerial inspection, reinforces the versatility of LiDAR-based SLAM in real-world field robotics. These active SLAM methods demonstrate the importance of precision sensing in navigation-critical domains. For example, in defense robotics, LiDAR-based SLAM underpins autonomous convoy systems, terrain-aware navigation in combat vehicles, and multirobot coordination in reconnaissance missions [63]. Similarly, mobile mapping units equipped with LiDAR SLAM systems aid in high-fidelity reconstructions of ancient ruins and caves in archaeology and heritage conservation. Despite their advantages, active SLAM systems often incur substantial hardware and energy costs. In forest robotics and long-term monitoring, key use cases for this thesis, their high power draw, sensitivity to dust and fog, and cost limitations, motivate the exploration of passive SLAM solutions. These alternatives promise lightweight deployment and adaptability to resource-constrained settings, which warrants a deeper comparative analysis.

2.2.4 Passive Sensor-Based SLAM

These systems typically require more advanced algorithmic support to infer depth and motion from the sensory data, yet they offer significant benefits in terms of weight, power efficiency, and cost. Their suitability in resource-constrained, visually complex, or rugged environments makes them particularly valuable in domains such as forest robotics, wildlife monitoring, and low-cost UAV applications. Monocular

SLAM uses a single RGB or grayscale camera to capture image sequences, inferring depth through photometric or geometric constraints across frames. Techniques such as ORB-SLAM [31], LSD-SLAM [64], and DSO [47] and DSM[65] represent key milestones in monocular SLAM development. These methods either rely on feature-based keypoint tracking or direct use of pixel intensities. While monocular SLAM is lightweight and hardware-efficient, it suffers from scale ambiguity, motion blur sensitivity, and drift in textureless environments. To overcome these issues, they are typically enhanced with inertial sensors or depth priors, such as known camera height, planar surfaces, or depth information inferred from stereo vision or utilization of learning models. Their low cost and compact design have made monocular setups particularly popular in robotics education and micro-drone navigation.

Stereo SLAM leverages two or more synchronized cameras with a fixed baseline to triangulate depth, removing the scale ambiguity inherent in monocular setups. Algorithms such as ORB-SLAM2 [26] and VINS-Fusion [32] support stereo inputs, benefiting from denser and clear depth cues. Stereo SLAM is better suited for indoor environments, where obstacles and scene geometry demand accurate perception. However, stereo setups are bulkier, require precise calibration, and may struggle in low-light or low-texture environments. Event cameras, also known as neuromorphic sensors, capture asynchronous changes in scene brightness rather than full image frames, making them highly responsive to motion and robust against high dynamic range (HDR) conditions. Algorithms such as EVO [66] and UltimateSLAM [67] exploit these sensors to achieve real-time SLAM in fast-moving or high-contrast environments. Event cameras excel in high-speed robotics, UAVs, and scenarios with repetitive textures where traditional cameras fail. However, the data representation is non-traditional, demanding specialized processing pipelines and often hybrid fusion with conventional sensors. Thermal or infrared SLAM systems use thermal cameras to detect emitted infrared radiation, offering visibility in total darkness

and through visual obstructions such as smoke or foliage. While not common in mainstream robotics, thermal SLAM is gaining traction in search-and-rescue operations, wildlife surveillance, and military applications. Algorithms adapted from standard visual SLAM apply grayscale feature tracking and calibration to handle lower-resolution, high-noise thermal data. Limitations include poor texture consistency, low image resolution, and variable heat signatures, all of which complicate reliable tracking [68].

Though less commonly adopted, some SLAM research explores passive modalities such as ambient light patterns or acoustic cues. For instance, acoustic SLAM using ambient echoes is explored for underwater robots [69], while ambient lighting variations have been used in indoor localization research. These methods remain niche but suggest potential for energy-autonomous SLAM systems in constrained environments. While IMUs, magnetometers, and barometers are often considered supplementary rather than purely passive, they significantly enhance the performance of passive SLAM. Visual-Inertial SLAM systems like VINS-Mono [32] and ORB-SLAM3 [23] demonstrate improved performance through IMU integration, enabling scale recovery and better handling of fast motion or temporary visual loss. Magnetometers provide coarse heading estimates, and barometers support altitude correction, which is particularly relevant in aerial or outdoor SLAM. Passive SLAM systems have found applications in mobile Augmented Reality (AR), planetary exploration rovers, and small-scale autonomous vehicles. With advances in computational photography and lightweight embedded computing, their deployment is expanding in forestry, cave exploration, and agricultural robotics. For example, vision-based SLAM has been applied to monitor vegetation structure, map forest trails, and study animal behavior under canopy cover [70]. Integrating deep learning with passive SLAM, such as depth prediction networks or semantic SLAM, further improves the performance of passive systems under adverse conditions.

In contrast to their active counterparts, passive SLAM solutions offer lower power requirements and more adaptable hardware configurations, making them suitable for long-duration missions in environmentally sensitive or inaccessible regions. This thesis draws upon such advantages to explore passive pipelines as viable, scalable solutions for real-time mapping and localization in unstructured outdoor environments.

2.2.5 Multi-Sensor Fusion SLAM

Multi-sensor fusion SLAM aims to harness the complementary advantages of both active and passive sensors by combining them in a unified SLAM framework. This approach enhances system robustness, accuracy, and adaptability across various environments, especially those characterized by rapid motion, dynamic lighting, or inconsistent texture. By fusing data from LiDAR, RGB cameras, IMUs, GPS, thermal imagers, and depth sensors, multi-sensor SLAM systems are better equipped to handle real-world complexities such as occlusions, feature-poor scenes, or localization drift. A prominent example is LVI-SAM [71], which tightly couples LiDAR and visual-inertial data in a factor-graph-based back-end, improving resilience to LiDAR degradation and poor lighting. Similarly, FAST-LIO2 integrates LiDAR and IMU using iterated Kalman filtering and can optionally incorporate visual data, enhancing performance under aggressive motion or structureless terrain [62].

Another significant example is VINS-Fusion [32], which allows flexible integration of stereo or monocular vision, IMU, and GPS. Such adaptability is critical in outdoor SLAM settings, where GPS intermittency and visual occlusions are common. The system has been successfully deployed in drones and mobile robots for urban and wilderness navigation. Multi-sensor approaches are also gaining traction in autonomous driving, where perception stacks incorporate LiDAR, radar, cameras, and inertial sensors. Systems like Apollo-SLAM and Cartographer Fusion exemplify

how diverse data sources can improve loop closure detection and long-term mapping consistency. In defense applications, multi-modal SLAM enhances situational awareness where the GPS signal is weak or absent, supporting convoy coordination, perimeter patrol in complex terrains [20]. Recent advances have also embraced deep learning in sensor fusion, combining learned depth priors or semantic segmentation maps with traditional geometric SLAM pipelines, as demonstrated in approaches like DVSO [72], SemanticFusion [73], and DeepTAM [74]. This is particularly beneficial in forest environments where occlusion from dense foliage and terrain variability can impair traditional methods.

Despite these advancements, several limitations remain in state-of-the-art SLAM systems. First, sensitivity to lighting conditions, dynamic objects, and textureless scenes continues to pose significant challenges, particularly for passive systems. Second, achieving long-term consistency across multi-session or lifelong mapping remains an open problem. Most SLAM systems struggle with map maintenance, scalability, and relocalization in previously visited environments. Moreover, computational constraints hinder the deployment of advanced SLAM architectures on low-power platforms. While LiDAR-based methods provide high-accuracy mapping, their hardware cost and power requirements make them unsuitable for small-scale or long-duration missions. Conversely, passive camera-based systems are more affordable and energy-efficient but face performance bottlenecks in visually degraded environments.

Below Table 2.1 summarizes the key differences between active, passive, and multi-sensor SLAM approaches. The comparison outlines their typical sensor modalities, representative algorithms, advantages, limitations, and common use cases. As evident from the table, multi-sensor fusion SLAM methods such as ORB-SLAM3 and VINS-Fusion exhibit better performance due to their ability to integrate complementary information from multiple sensing dimensions. This integration enhances

pose estimation accuracy and overall localization, particularly in complex and dynamic environments.

SLAM Type	**Primary Sensors**	**Key Algorithms**	**Advantages**	**Limitation**	**Common Applications**
Active	LiDAR, Radar, Sonar, Structured Light	Cartographer, LOAM, LIO-SAM, GMapping, Hector SLAM	High depth accuracy, robust in varied lighting, suitable for large-scale mapping	Expensive hardware, high power draw, performance degraded in fog, rain	Autonomous vehicles, industrial robots, urban mapping; defense
Passive	Monocular Cameras, Event Cameras, Thermal Cameras	ORB-SLAM, LSD-SLAM, DSO, DSM, EVO, Thermo-SLAM	Lightweight, energy-efficient, low-cost, good for constrained settings	Sensitive to blur and lighting, scale ambiguity in monocular setups	Forest robotics, UAVs, AR/VR, wildlife monitoring, planetary exploration
Multi	Visual, IMU, LiDAR, GPS (in any combination)	ORB-SLAM3, VINS-Fusion, LIO-SAM, DROID-SLAM	Robustness in dynamic scenes, accurate scale and relocalization, works in GPS-denied areas	High computational load, complex sensor synchronization, and calibration	Rescue robotics, drones, underwater SLAM, autonomous exploration, space robotics

Table 2.1: Comparison of Active, Passive, and Multi-Sensor Fusion SLAM Types

To conclude this section, an extensive review of SLAM methodologies encompassing both passive and active sensor-based approaches was conducted. Given the focus of this thesis on enabling stealth-capable and power-efficient navigation in natural, unstructured environments, SLAM systems relying on active sensors such as LiDAR were considered beyond the scope of this study. Hence, emphasis was placed on passive visual SLAM techniques, which offer reduced power consumption and less environmental disturbance. Among these, as seen in Table 2.2, ORB-SLAM2

and ORB-SLAM3 were identified as the most promising candidates based on their proven accuracy, robustness, and widespread adoption across multiple benchmark datasets and most importantly they are open-sourced. These frameworks exemplify state-of-the-art capabilities in passive SLAM, leveraging efficient feature-based tracking and tightly integrated back-end optimization. A detailed explanation of these two SLAM frameworks are explained in the following Section 2.3.

System	Type	Features	Back-end	Loop Closure	Accuracy	Robustness
Visual SLAM Systems						
Mono-SLAM	SLAM	Shi-Tomasi	EKF	–	Fair	Fair
PTAM	SLAM	FAST	BA	–	Very Good	Fair
ORB-SLAM2	SLAM	ORB	Local BA	DBoW2 PG+BA	Excellent	Very Good
LSD-SLAM	SLAM (Direct)	Edgelets	PG	FABMAP PG	Good	Fair
DSO	VO (Direct)	High Gradient	Local BA	–	Fair	Very Good
DSM	SLAM	High Gradient	Local BA	–	Very Good	Very Good
Visual-Inertial SLAM Systems						
ORB-SLAM-VI	SLAM	ORB	Local BA	DBoW2 PG+BA	Very Good	Very Good
VINS-Fusion	VIO	Shi-Tomasi	Local BA	DBoW2 PG	Good	Excellent
Kimera	VIO	Shi-Tomasi	Local BA	DBoW2 PG	Good	Excellent
VI-DSO	VIO (Direct)	High Gradient	Local BA	–	Very Good	Excellent
BASALT	VIO	FAST	Local BA	ORB BA	Very Good	Excellent
ORB-SLAM3	SLAM	ORB	Local BA	DBoW2 PG+BA	Excellent	Excellent

Table 2.2: Summary of representative visual and visual-inertial systems [23]

2.3 Overview of ORB-SLAM Family

Visual keypoint-based SLAM has undergone significant evolution since the introduction of ORB-SLAM in 2015 [31]. This chapter presents an in-depth examination of two influential successors, ORB-SLAM2 [26] and ORB-SLAM3 [23], which form the basis for the empirical evaluations in this thesis. While both leverage the efficiency of ORB descriptors [43] and bag-of-words–based place recognition [45], they differ substantially in sensor compatibility, back-end optimization, and architectural flexibility.

2.3.1 ORB-SLAM2 framework

ORB-SLAM2 extends the capabilities of its monocular predecessor by incorporating stereo and RGB-D configurations, addressing scale ambiguity and depth limitations inherent in monocular systems. The system integrates DBoW2-based loop closure which supports long-term mapping and is implemented in a highly modular, multi-threaded C++ architecture.

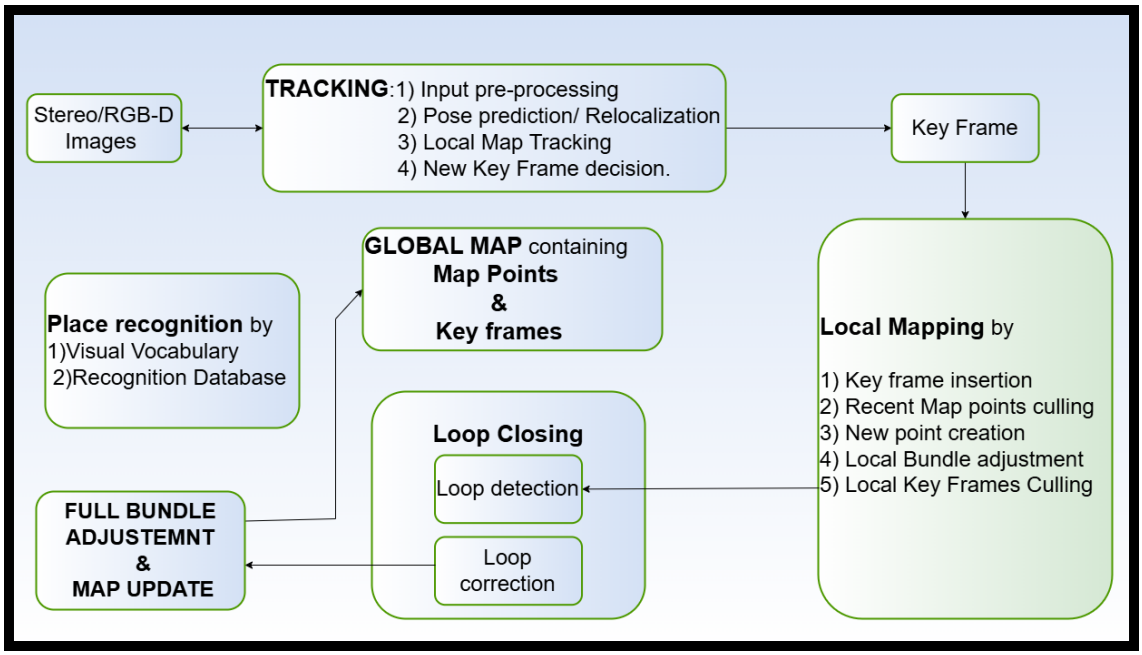

Figure 2.2: High-level architectural pipeline of ORB-SLAM2. [26]

As illustrated in Figure 2.2, the system comprises three parallel modules: tracking, local mapping, and loop closing. The tracking module extracts ORB features, estimates pose using a motion model or depth prior, and decides on keyframe insertion based on parallax and scene change thresholds. Local mapping involves keyframe insertion, point triangulation, local bundle adjustment, and redundancy culling. The loop-closing module performs candidate detection using DBoW2, geometric verification via Sim3 transformation, and global pose graph optimization. In some cases, a full bundle adjustment is triggered in a dedicated thread to refine the global map. ORB-SLAM2 supports monocular, stereo, and RGB-D sensors,

each with specific trade-offs. While monocular setups are lightweight and cost-effective, they suffer from scale drift unless loop closures are frequent. Stereo configurations provide direct triangulation for scale estimation and enhanced robustness in low-texture scenes. RGB-D systems offer dense depth but are often limited to short-range indoor applications due to interference from ambient light. Performance benchmarks show that ORB-SLAM2 achieves sub-percent translational RMSE on the KITTI Odometry dataset and centimeter-level accuracy on EuRoC MAV sequences, demonstrating stability in outdoor driving and indoor drone navigation scenarios [26]. Despite its success, ORB-SLAM2 lacks native support for inertial data and dynamic scene modeling, both critical in environments such as forests. These limitations motivated the development of ORB-SLAM3.

2.3.2 ORB-SLAM3 framework

ORB-SLAM3 generalizes the framework to support monocular, stereo, RGB-D, and tightly coupled visual-inertial (VI) configurations. Unlike ORB-SLAM2, it operates under a unified architecture that seamlessly integrates inertial measurements through IMU pre-integration, improving tracking stability in scenes with rapid motion or poor visual features. As depicted in Figure 2.3, ORB-SLAM3 retains the core modules of tracking, local mapping, and loop closure but incorporates significant architectural enhancements. The tracking module fuses ORB feature-based pose estimation with IMU integration for improved short-term motion prediction. Local mapping now includes IMU initialization and scale refinement, and the system supports multi-map management via an internal structure called the Atlas. When large tracking failures occur, ORB-SLAM3 spawns separate submaps that can be merged upon loop closure detection, improving robustness in fragmented or large-scale environments. A revised DBoW2-based place recognition module with improved geometric verification reduces false positives and enhances relocalization

performance. Loop closures now involve both traditional pose-graph optimization and map merging routines to ensure global consistency. ORB-SLAM3's support for visual-inertial fusion shall potentially benefit forested or dynamic outdoor settings, where texture scarcity, motion blur, and rapid viewpoint changes can degrade visual tracking. Empirical results on datasets such as EuRoC MAV and KITTI Odometry confirm ORB-SLAM3's superior accuracy in such scenarios [23].

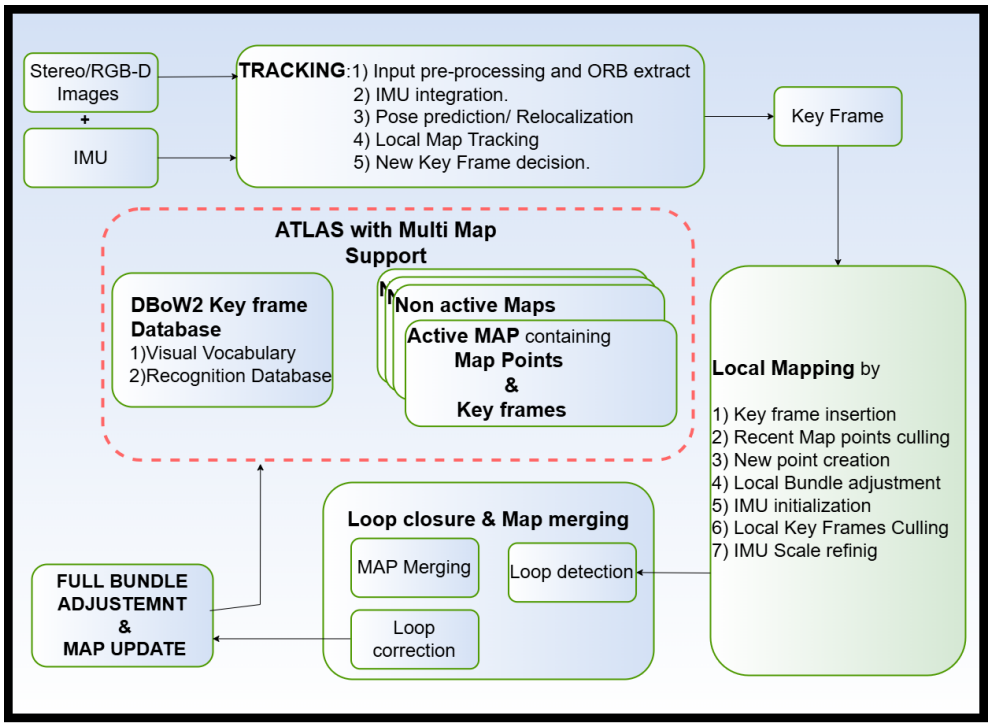

Figure 2.3: High-level architectural pipeline of ORB-SLAM3. [23]

2.3.3 Comparative Summary

Following the detailed discussion of the architectural components of SLAM and a breakdown of the ORB-SLAM2 and ORB-SLAM3 systems, this section synthesizes a comparative evaluation of the two frameworks to highlight their practical trade-offs. This comparative lens is particularly valuable for understanding their applicability to the operational demands of natural, forested environments, where conditions such as visual ambiguity, low parallax, and environmental dynamics present signif-

icant challenges to SLAM performance. Table 2.3 provides a high-level qualitative comparison between ORB-SLAM2 and ORB-SLAM3. Although both frameworks share a common lineage, built on efficient ORB features and a bag-of-words loop closure back-end, their design philosophies diverge to address different operational requirements. ORB-SLAM2 is engineered for visual-only applications, emphasizing computational efficiency and straightforward deployment across monocular, stereo, and RGB-D sensors. In contrast, ORB-SLAM3 introduces inertial fusion, multi-session mapping, and a unified software architecture, making it more suitable for real-world scenarios involving sensor noise, intermittent tracking failure, and long-duration missions. Unlike ORB-SLAM2, which relies on a single continuous map, ORB-SLAM3 employs an Atlas-based structure that manages multiple active and inactive maps. This is highly beneficial in field deployments where temporary localization and tracking failure is inevitable. The ability to spawn and later merge submaps ensures the global map remains coherent, facilitating long-term autonomy in natural terrains with variable geometry and lighting conditions.

Another key difference illustrated in Table 2.3 is the presence of tightly coupled inertial fusion in ORB-SLAM3. This advancement addresses one of the most critical failure modes in visual SLAM which is to lose tracking due to motion blur, and rapid rotations. These are the conditions often encountered in forested paths where feature-rich views may be occluded or unavailable. ORB-SLAM3's ability to integrate IMU data directly into its back-end optimization allows it to maintain accurate motion estimation even in the absence of distinct visual features. In terms of loop closure, ORB-SLAM3 offers enhanced place recognition capabilities, integrating inertial constraints during loop detection and supporting cross-session matching in indoor environments based on the results mentioned in their research work[23]. This enhances the reliability of relocalization, particularly when visual appearance changes due to weather, season, or perspective, key issues in forest applications.

Component	**ORB-SLAM2 [26]**	**ORB-SLAM3 [23]**
Front-End	Uses ORB features (FAST + BRIEF) for keypoint detection and description.	Same ORB features, but integrates IMU constraints earlier for better initialization and tracking.
Camera Support	Monocular, stereo, and RGB-D configurations.	Monocular, stereo, RGB-D, and multi-map support for multi-session and multi-camera setups.
Back-End	Graph-based SLAM with local and global Bundle Adjustment (BA) using `g2o`.	Uses `GTSAM` with visual and inertial data, supporting tightly coupled BA and IMU pre-integration.
Map Representation	Single map with sparse point cloud.	Multi-map representation; supports multi-session mapping and joint optimization of trajectories.
Loop Closure	DBoW2 for place recognition and pose-graph optimization.	Improved loop closure using inertial constraints, DBoW2 and cross-map session recognition.
Sensor Fusion	None (purely visual).	Full visual-inertial fusion; tightly integrated with the optimization pipeline.
Initialization	Visual-only, requires parallax or stereo disparity.	IMU-assisted initialization improves robustness under low-parallax or degraded visual conditions.
Dynamic Environment Handling	Limited assumes a static environment.	Improved handling through multi-session filters, though still primarily assumes static scenes.
Real-Time Performance	Optimized for real-time, especially with stereo inputs.	Real-time capable, but more computationally demanding due to sensor fusion and mapping overhead.
Application Suitability	Suitable for indoor/outdoor robotics, AR/VR, and basic navigation.	Designed for UAVs, GPS-denied operations, dynamic scenes, and long-term multi-session mapping.

Table 2.3: Module-by-Module Comparison of ORB-SLAM2 and ORB-SLAM3

Several additional considerations must be taken into account to complement the insights presented in Table 2.3, particularly when assessing the operational reliability of SLAM systems in forested environments. One critical aspect is robustness to initialization failures, which remains a prominent challenge in monocular SLAM pipelines. Traditional systems such as ORB-SLAM2 rely on observing sufficient parallax between frames to perform successful triangulation and establish a reliable scale estimate. However, in forest environments, especially when navigating narrow trails or low-feature corridors, parallax may be insufficient due to constrained motion or repetitive scenery. This results in unstable or failed initialization, hindering the system's ability to establish a coherent map. ORB-SLAM3 addresses this issue by incorporating tightly coupled inertial priors through IMU preintegration. This allows the system to estimate motion and scale even under conditions of limited visual variability, such as slow forward motion through a densely vegetated path. By reducing dependence on visual-only initialization, ORB-SLAM3 enhances reliability in forest navigation scenarios where initial pose uncertainty is high. Another vital consideration is the system's ability to recover from tracking loss, a common occurrence in unstructured natural environments. Tracking failures may result from temporary visual occlusion, caused by overhanging branches, passing wildlife, or shadows, or from rapid motion blur induced by unstable platform dynamics, such as vibrations in UAVs or uneven terrain in ground vehicles. ORB-SLAM2 incorporates relocalization mechanisms based on keyframe matching via DBoW2, but its performance degrades significantly under substantial viewpoint changes or visual degradation. In contrast, ORB-SLAM3 improves relocalization through enhancements to its place recognition backend, integrating both visual and inertial constraints for more accurate frame reassociation. This advancement enables the system to resume mapping more effectively after temporary failures, thereby supporting long-term autonomy in dynamic forest settings.

The trade-off between accuracy and latency is another critical factor in selecting a SLAM system for real-world deployment. While ORB-SLAM2 is optimized for speed and is well suited to resource-constrained platforms, its accuracy tends to degrade over longer trajectories due to cumulative drift, particularly in monocular configurations. ORB-SLAM3, on the other hand, incurs a higher computational burden due to its multithreaded optimization routines and inertial fusion overhead, especially when handling visual-inertial data. However, this increased complexity yields substantial improvements in mapping accuracy, trajectory stability, and drift reduction characteristics essential for mission-critical tasks such as post-disaster environmental surveying, long-range ecological monitoring, or autonomous exploration in GPS-denied regions. Lastly, ORB-SLAM3 exhibits significant potential for scalability and multi-agent extension, which positions it as a forward-compatible framework for collaborative mapping tasks. Its multimap feature can potentially facilitate the extension of the system to multi-agent scenarios such as a fleet of UAVs or a team of aerial and ground robots operating in parallel across a forested area. Each agent can maintain its local map and periodically synchronize with a shared global map. This is particularly advantageous for applications such as coordinated search and rescue missions or real-time environmental monitoring, where comprehensive spatial coverage and modular deployment is required. The architectural readiness of ORB-SLAM3 to accommodate such extensions supports its role as a foundation for future research aimed at distributed SLAM in complex natural environments. These insights form the analytical foundation for the the methodology and implementation procedures in Chapter 3 which includes a comprehensive description of the datasets used and the performance metrics deployed to assess the SLAM methods and experiments conducted in Chapter 4, where both systems are rigorously evaluated on two contrasting datasets.

3 Methodolgy

This chapter outlines the methodological framework and the implementation steps undertaken in the course of this thesis. Section 3.1 provides a description of the TUM RGB-D dataset [75] and the FinnForest dataset [11], both of which serve as benchmark environments for evaluating the performance of passive visual SLAM methods. Section 3.2 details the hardware and software libraries , and Section 3.3 describes evaluation tools, and metrics used to assess and compare the trajectory accuracy and robustness of the selected SLAM systems, which form the foundation for subsequent analysis and interpretation.

3.1 Datasets

Stereo imagery is essential in this study because a single stereo image pair delivers instantaneous metric depth, eliminating the scale ambiguity that hampers monocular SLAM during rapid motion. Two public datasets were selected that meet this requirement, yet expose the algorithms to markedly different conditions: the laboratory-grade TUM RGB-D benchmark and the forest field-oriented FinnForest dataset collection.

3.1.1 TUM RGB-D Dataset

The TUM RGB-D dataset [75] is a widely recognized benchmark in visual SLAM, particularly for systems leveraging monocular or RGB-D sensors. Developed by the

Technical University of Munich, this dataset has been instrumental in advancing SLAM algorithm development by providing high-quality, synchronized data streams in indoor environments. It offers a rich combination of RGB images, depth maps, and high-frequency ground-truth trajectories captured using a high-precision motion capture system (ART). The availability of precise ground truth 100 Hz and well-calibrated sensor data renders this dataset highly suitable for isolating algorithmic performance from environmental uncertainty. This feature is particularly valuable during the initial phases of system validation. Each sequence in the dataset consists of RGB images with a resolution of 640×480 pixels, recorded at a frame rate of 30 Hz. The depth maps are generated using structured light sensing, which ensures frame-level synchronization with the RGB stream. Furthermore, intrinsic and extrinsic calibration parameters are provided for each sequence, allowing seamless integration with SLAM systems such as ORB-SLAM2 and ORB-SLAM3 through YAML configuration files. These parameters ensure geometric consistency between image measurements and reconstructed 3D structures, particularly during bundle adjustment and trajectory estimation. This level of sensor fidelity and calibration precision makes the TUM RGB-D dataset better suited for benchmarking a SLAM system under controlled conditions.

Beyond its technical specifications, the dataset's organizational structure is conducive to evaluating SLAM performance across various relevant challenges. A sample Figure 3.1 showing different image frames and the indoor factory environment is shown for understanding the dataset. It includes sequences designed to stress-test various components of SLAM pipelines: static indoor scenes with smooth camera motion (e.g., fr1/xyz sequence), sequences with explicit loop closures and larger camera trajectories (e.g., fr2/large_with_loop), and dynamic scenes with human motion and occlusions (e.g., fr3/walking_xyz).

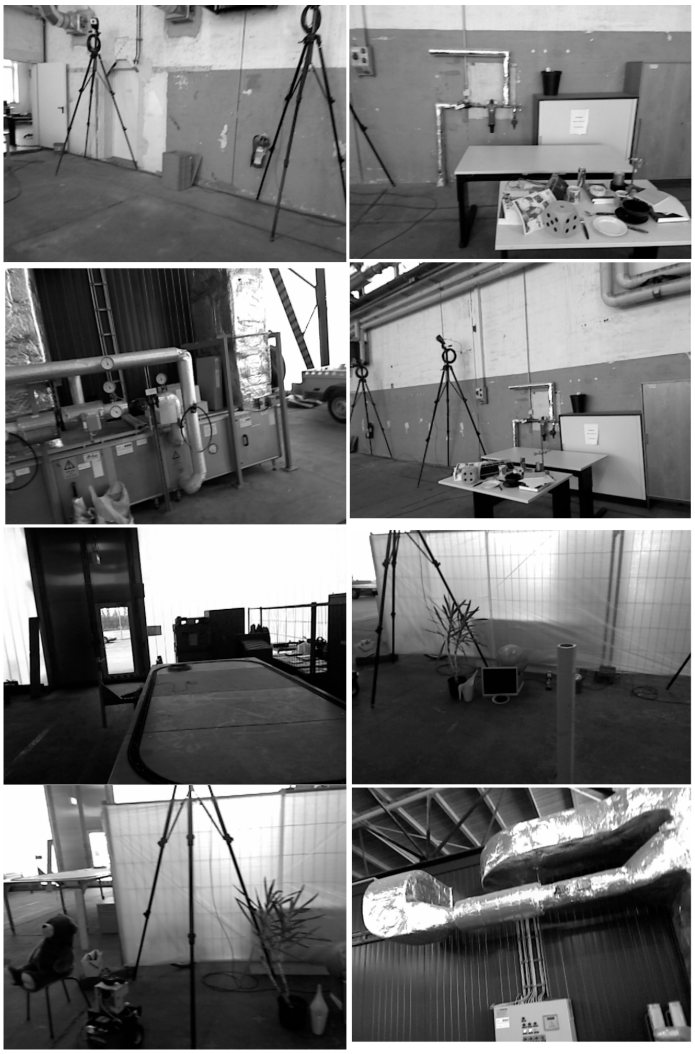

Figure 3.1: Sample images from the TUM RGB-D dataset from various sequences, showing a structured factory environment with distinct features in the mapping area. [75]

Additional sequences incorporate fast camera movement, motion blur, and scenes with sparse or ambiguous visual features. This diversity allows researchers to probe the behavior of feature detection, pose estimation, loop closure, and relocalization under a variety of operational conditions, all while maintaining a consistent baseline for comparative analysis. Although the primary objective of this thesis is to evaluate the performance of passive visual SLAM systems in unstructured outdoor environments, particularly forested terrain, the TUM RGB-D dataset serves a complementary and important role within the experimental framework. It provides a

noise-controlled, repeatable environment for validating the integrity of the SLAM pipeline before deployment in complex real-world settings. In particular, it facilitates the establishment of baseline metrics such as ATE and RTE, which are explained in detail in Section 3.3.

Moreover, the structured nature of the TUM RGB-D sequences enables a focused analysis of loop closure and relocalization behavior in scenes where ground-truth loop candidates are known to exist. This is particularly important for systems such as ORB-SLAM2 and ORB-SLAM3, which depend heavily on bag-of-words place recognition and geometric verification for maintaining global consistency. By testing in such an environment, one can rigorously evaluate the effectiveness of loop closure strategies and the system's ability to recover from deliberate tracking interruptions. These insights inform the calibration of loop closure thresholds and influence the tuning of vocabulary size and feature dimensionality factors that are later stress-tested in natural scenes. In the broader context of this thesis, the use of the TUM RGB-D dataset also facilitates comparison between different sensing modalities. The availability of both RGB and depth data allows for controlled experimentation in monocular and RGB-D modes, which provides depth as an additional parameter similar to stereo images, thereby supporting the thesis's investigation into the trade-offs of passive SLAM configurations.

Although the TUM RGB-D dataset provides three-channel RGB image sequences, both ORB-SLAM2 and ORB-SLAM3 internally convert these images to grayscale. This is because the extracted ORB features are color invariant and operate on grayscale images, so to enable more efficient computation without compromising feature extraction quality, the images are converted to grayscale before hand.

Such comparative studies are critical in assessing how system performance scales across sensor choices, especially when planning for deployments in power- and weight-constrained platforms like UAVs or autonomous ground robots. In this way, the

TUM RGB-D dataset supports not only algorithmic validation but also sensor strategy development, offering a bridge between theoretical performance and practical field deployment. It provides a rigorous testbed for algorithm validation, parameter tuning, and baseline performance measurement, ensuring that any observed degradations in forest environments, such as those reported in the FinnForest dataset, can be directly attributed to environmental complexity rather than intrinsic flaws in the SLAM system. The use of TUM RGB-D data is thus a deliberate methodological choice, one that strengthens the overall rigor of the thesis and facilitates robust conclusions regarding SLAM performance in natural, unstructured, and passive-sensor–based applications.

3.1.2 FinnForest dataset

The FinnForest dataset [11] represents a pivotal contribution to the landscape of SLAM evaluation benchmarks by addressing a critical gap left by urban- and indoor-centric datasets. Collected on logging forest trails near Tampere, Finland, the dataset offers a rich, naturalistic testing ground for autonomous navigation algorithms in unstructured forest environments. Unlike conventional datasets constrained to static scenes or artificial urban settings, FinnForest captures the complexity and variability inherent to woodland terrain, including variable lighting, seasonal transformation, and environmental dynamics. Its design is particularly suited for probing the failure modes of passive visual SLAM systems. Data acquisition was carried out using a vibration-damped roof rack mounted on a vehicle, hosting four Basler acA1920-40uc cameras. The forward-facing stereo pair, separated by a 20-centimeter baseline and capturing high-resolution (2013×1193 pixels) frames at 40 Hz, forms the primary visual input for all SLAM experiments presented in this work. In addition to visual data, the dataset includes centimeter-level ground truth poses generated using inertial measurements from a KVH 1750 fiber-optic IMU sampled

at 200 Hz and fused with a NovAtel PwrPak7 RTK-GNSS receiver operating at 20 Hz. The ground-truth signals are post-processed using tightly coupled smoothing algorithms to provide reference trajectories of high spatial fidelity.

One of the most defining features of FinnForest is its comprehensive coverage of seasonal and illumination variance, which significantly enriches its value as a benchmarking tool. The dataset includes eleven sequences spanning both snowy winter and leafy summer conditions, covering distances from approximately 1.3 km to 6.5 km. These sequences capture a diverse range of ambient lighting scenarios, including overcast skies, direct sunlight, twilight, and even night, thus reflecting the spectrum of real-world conditions that autonomous systems must confront in forested settings. Figure 3.2 illustrates various image frames from summer and winter sequences, highlighting the extent of appearance variation due to foliage density, snow cover, and lighting conditions. Such variations pose severe challenges for SLAM modules like feature matching and loop closure, especially in environments where self-similar textures (e.g., tree trunks, branches) dominate the visual field. As summarized in Table 3.1, the dataset is organized into multiple trajectories labeled by season and sequence number (e.g., W01, S01), with some sequences explicitly designed to contain loop closures. This variety allows for controlled evaluation of algorithmic performance under realistic challenges. Notably, loop closure is most fragile in such woodland environments due to the temporal variability of appearance and the prevalence of visually repetitive patterns, which may lead to false positives or missed detections. This makes FinnForest a suitable testbed for assessing the robustness of visual place recognition algorithms, such as those implemented in ORB-SLAM2 and ORB-SLAM3, and their impact on long-term map consistency.

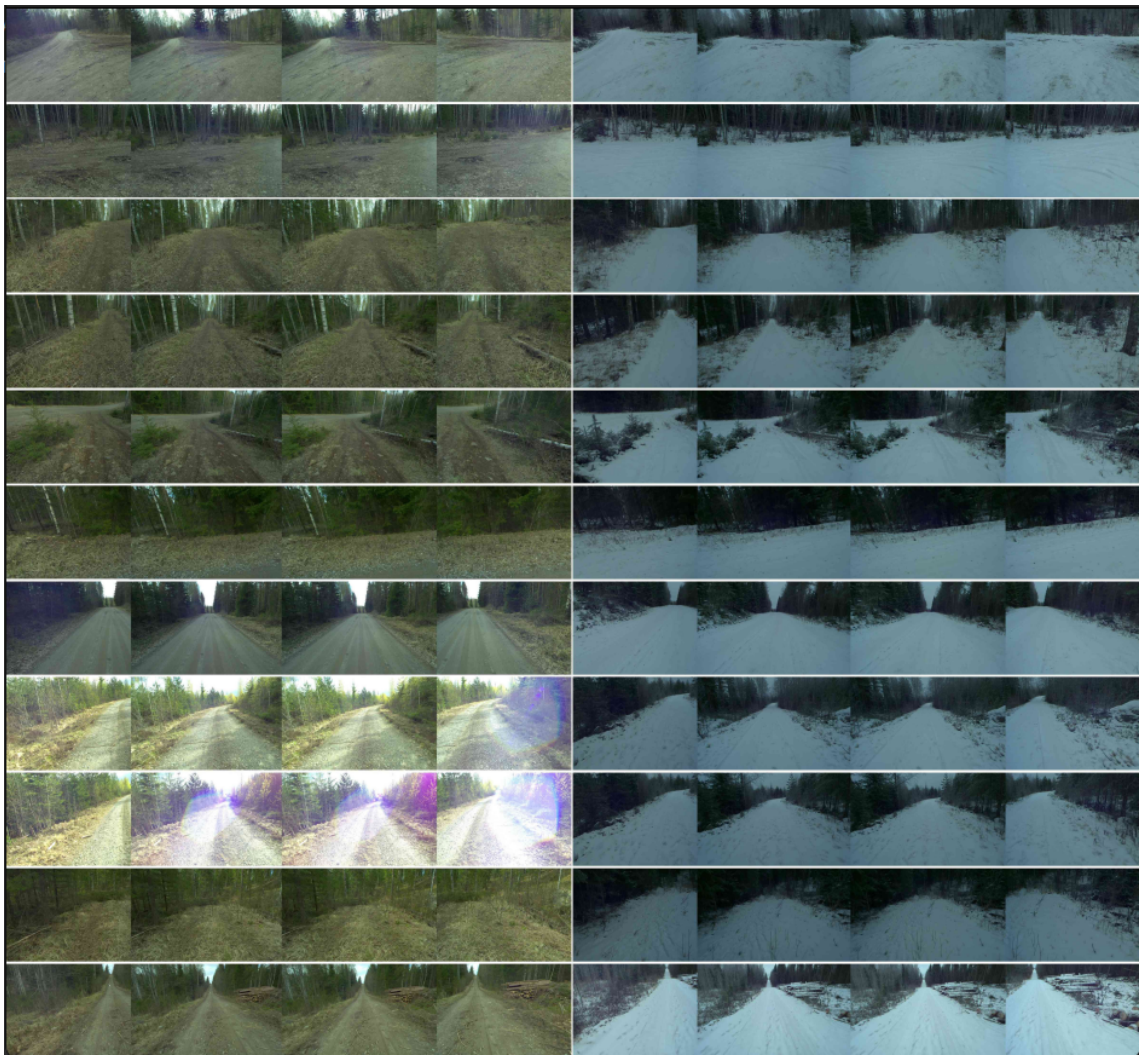

Figure 3.2: Sample images from the Finnforest dataset from various sequences, showing snow-covered trails and varied lighting in summer sequences. [11]

ID	Frames (40 Hz)	Dist. [km]	Loop	Season	Light
W01 / S01	27 630 / 27 960	1.29	Yes	Winter / Summer	Low / High
S02	21 333	1.99	Yes	Summer	High
W03 / S03	23 100 / 15 000	1.69	No	Winter / Summer	Low / High
W04 / S04	37 010 / 30 662	2.3	No	Winter / Summer	Low / High
W05 / S05	57 288 / 61 662	4.74 / 5.84	No	Winter / Summer	Low / High
W06 / W07	20 875 / 43 780	3.59 / 6.48	No	Winter	Low / Low

Table 3.1: FinnForest sequence overview for forward stereo cameras

From a systems design perspective, the high-frequency image data, stereo geometry, and ground-truth accuracy make FinnForest particularly valuable for evaluat-

ing scale consistency, pose drift, and loop closure success rates. These are precisely the failure points that passive SLAM systems tend to encounter in forested terrain, where texture sparsity, rapid appearance change, and motion-induced blur are common. By offering high-resolution stereo input at 40 Hz, the dataset allows the system to leverage fine spatial detail while also demanding real-time performance under computational constraints.

The inclusion of synchronized reference trajectories further enables precise quantification of mapping and localization errors across varying environmental conditions. In the context of this thesis, FinnForest is not merely a performance benchmark but a simulation of deployment reality. It serves as a proxy for scenarios such as autonomous forest patrol, ecological monitoring, and disaster response in GPS-denied regions. The challenges embodied in FinnForest, ranging from low-texture regions and dense canopy occlusion to the need for robust scale recovery, directly reflect the motivating problem space of this research. By benchmarking ORB-SLAM2 and ORB-SLAM3 against this dataset, this thesis can ground its claims in empirical reality, providing rigorous insights into the operational limits and optimization opportunities of passive visual SLAM systems in natural terrain. In conclusion, the FinnForest dataset serves as a key experimental resource in this work, enabling a deep investigation into the behavior of SLAM systems under the exact set of constraints and environmental complexities that real-world forest deployments impose. Its seasonal breadth, high-frequency stereo data, and reliable ground-truth make it well suited to the thesis's aim of advancing robust, passive-sensor SLAM strategies for unstructured outdoor environments.

3.2 Hardware and Software Libraries

This section outlines the hardware and software environments used to implement and evaluate ORB-SLAM2 and ORB-SLAM3 in this thesis. Table 3.2 provides

the hardware specifications of the system used in this study. Although the system includes a powerful GPU, both SLAM systems are primarily designed for CPU multi-threaded execution. As such, GPU acceleration is not critical and is mainly utilized for rendering and visualization tasks via GUI support.

Component	**Specification**
Processor	Intel Core Ultra 7 165H vPro Enterprise (16 cores, 22 threads, 24 MB cache, up to 5.0 GHz)
RAM	32 GB LPDDR5x 7467 MT/s, dual-channel
Storage	1 TB PCIe NVMe M.2 SSD
GPU	NVIDIA RTX 3000 Ada (8 GB GDDR6)
Operating System	Ubuntu 20.04.6 LTS

Table 3.2: Hardware specifications

ORB-SLAM2 and ORB-SLAM3 minimal requirements on the System are 16 giga bytes of RAM and a minimum of intel i7 are recommended but our system is almost twice the mentioned requirement. implemented in `C++` [76] and rely on a set of open-source libraries for key functions including `OpenCV`($\geq$ 3.2) [77] for image processing and ORB feature extraction. Linear algebra operations are handled by `Eigen3` [78], while geometric transformations use Lie groups via `Sophus` [79]. Loop closure and relocalization rely on the `DBoW2` bag-of-words framework [80], and `g2o` is used for graph-based optimization [81]. Real-time 3D visualization is supported by `Pangolin` [82]. Projects are built using `CMake` and compiled with `Make` or `Ninja` [83], [84]. `ROS` is optionally used for robotic integration [85], and trajectory evaluation is done via the `EVO` toolkit [86]. These libraries collectively enable real-time, modular, and accurate SLAM system development.

3.3 Evaluation Tools and Criteria

A methodologically sound and consistent evaluation framework is critical for assessing the effectiveness of SLAM systems, particularly when comparing different

algorithms across varying environmental conditions. Given that visual SLAM algorithms differ significantly in their design assumptions, sensor dependencies, and operational trade-offs, a standardized performance assessment methodology is necessary to ensure rigorous benchmarking and reproducibility. In this study, all trajectory evaluations are conducted using the evo toolset, a widely accepted Python-based library for SLAM and visual odometry analysis developed by Grupp et al. [86]. Its adoption within the robotics and computer vision communities stems from its robust metric definitions, extensibility, and compatibility with several SLAM datasets and file formats, including TUM, KITTI, and EuroC. The toolset facilitates quantitative trajectory analysis by aligning estimated and ground-truth trajectories using rigid-body transformations and evaluating them across standardized metrics. Among these, the Absolute Pose Error (APE), often referred to interchangeably with Absolute Trajectory Error (ATE), serves as a global measure of pose accuracy which is usually calculated in meters(m) . APE quantifies the Euclidean distance between corresponding poses in the estimated and reference trajectories after temporal and spatial alignment. ATE mean value is given by

$$\mathrm{ATE}_{\mathrm{mean}} = \frac{1}{N} \sum_{i=1}^{N} \left\| \mathbf{p}_i^{\mathrm{gt}} - \hat{\mathbf{p}}_i^{\mathrm{est}} \right\| \tag{3.1}$$

Where $\mathbf{p}_i^{\mathrm{gt}}$ is the ground truth position at index i, $\hat{\mathbf{p}}_i^{\mathrm{est}}$ is the aligned estimated position at index i, N is the total number of poses.

This metric offers a global performance overview by reflecting cumulative errors across an entire sequence. Conversely, the Relative Pose Error (RPE), sometimes termed Relative Trajectory Error (RTE), is employed to assess local trajectory consistency. It evaluates the deviation between relative motions over fixed temporal or spatial intervals which is represented in percentage values (%). RPE thus captures drift and short-term inconsistency, both of which are especially important in real-

world applications where precise local tracking is essential for effective navigation, manipulation, or mapping.

$$\mathrm{RTE}_{\mathrm{mean}} = \frac{1}{N}\sum_{i=1}^{N}\left\|\left(\mathbf{P}_i^{\mathrm{gt}^{-1}}\cdot\mathbf{P}_{i+\Delta}^{\mathrm{gt}}\right)^{-1}\cdot\left(\mathbf{P}_i^{\mathrm{est}^{-1}}\cdot\mathbf{P}_{i+\Delta}^{\mathrm{est}}\right)\right\|_{\mathrm{trans}} \tag{3.2}$$

Where $\mathbf{P}_i^{\mathrm{gt}}$ is the ground truth pose at time i, $\mathbf{P}_i^{\mathrm{est}}$ is the estimated pose at time i, Δ is the fixed frame interval (e.g., 1), N is the number of valid pose pairs for RTE calculation, $\|\cdot\|_{\mathrm{trans}}$ denotes the Euclidean norm of the translational component.

Beyond numerical evaluation, the evo toolset incorporates a range of visualization capabilities to support interpretability. These include 2D and 3D plots of estimated and ground-truth paths, error histograms, and time-synchronized overlays of translational and rotational deviations. Such visual analytics facilitate a more nuanced understanding of performance degradations, helping to localize algorithmic failures linked to environmental conditions such as tracking loss in foliage-heavy segments or inaccurate relocalization under repetitive scene structures. The evaluation conducted in this thesis applies evo to both benchmark datasets used. The TUM RGB-D dataset, collected under laboratory conditions with high-precision motion capture, enables controlled analysis of trajectory errors in well-structured, low-noise settings. This environment is particularly effective for validating SLAM pipeline integrity, tuning algorithmic parameters, and establishing reference performance for stereo configurations. Conversely, the FinnForest dataset provides a challenging real-world scenario, characterized by visual aliasing, dynamic lighting, and low-texture regions typical of natural forest environments. Evaluating SLAM performance under these contrasting conditions ensures that the insights derived are not only generalizable but also grounded in operational reality. This dual-dataset strategy underlines a key methodological objective of the thesis, which is to move beyond synthetic or urban-centric evaluations and investigate SLAM performance

in unstructured outdoor terrains. The choice of stereo input alone, absent any inertial or GNSS fusion during testing, ensures that the error metrics computed via evo reflect the isolated behavior of the visual front-end and back-end optimizer. By excluding auxiliary sensors from the runtime configuration and using ground-truth trajectories solely for benchmarking purposes, this evaluation paradigm allows for a like-for-like comparison of visual SLAM systems across both domains. Together, the TUM RGB-D and FinnForest datasets form a complementary evaluation criteria. The former provides laboratory-grade precision for pipeline validation and baseline comparisons. The latter exposes SLAM algorithms to real-world challenges like scale ambiguity, motion blur, environmental occlusions, and seasonal changes. Analyzing both the setups using evo, enables transparent and replicable methodology to assess the robustness, accuracy, and operational limits of passive stereo SLAM in natural environments.

4 Results

This chapter presents comparative results on the performance of ORB-SLAM2 and ORB-SLAM3 across two distinct environments: structured indoor settings, represented by the TUM RGB-D dataset, and unstructured natural outdoor environments, represented by the Finnforest dataset. Section 4.1 outlines the results obtained from the TUM RGB-D dataset, while Section 4.2 focuses on the results derived from the Finnforest dataset. Additionally, a brief interpretation is provided for each sequence to contextualize the system behavior and error characteristics observed during evaluation.

To assess the performance of the systems quantitatively, this study employs two widely recognized metrics namely ATE and RTE which calculates absolute and relative translation errors using the evo evaluation toolkit. The results cover both successful tracking scenarios, where the SLAM algorithms exhibit stability and accuracy, and failure cases marked by loss of tracking or unsuccessful loop closures. Such failures are often linked to challenging environmental factors, including sparse textures, moving vegetation, or variable lighting conditions. A total of six sequences are analyzed: one from the TUM RGB-D dataset, captured inside a spacious industrial hall that allows for a potential loop closure at the end, and five from the Finnforest dataset. Of the Finnforest sequences, three represent summer conditions, while the remaining two are recorded during winter.

The results presented here contribute directly to addressing the central research

question on whether passive visual SLAM systems designed for urban and indoor contexts operate effectively in natural, forested environments.

4.1 TUM RGB-D sequence results

4.1.1 TUM RGB-D Sequence 1:

freiburg2_large_with_loop sequence was recorded using a handheld RGB-D sensor (Microsoft Kinect) in a large industrial hall. The environment is populated with feature-rich objects, which facilitate accurate camera trajectory estimation when visual SLAM algorithms are applied.

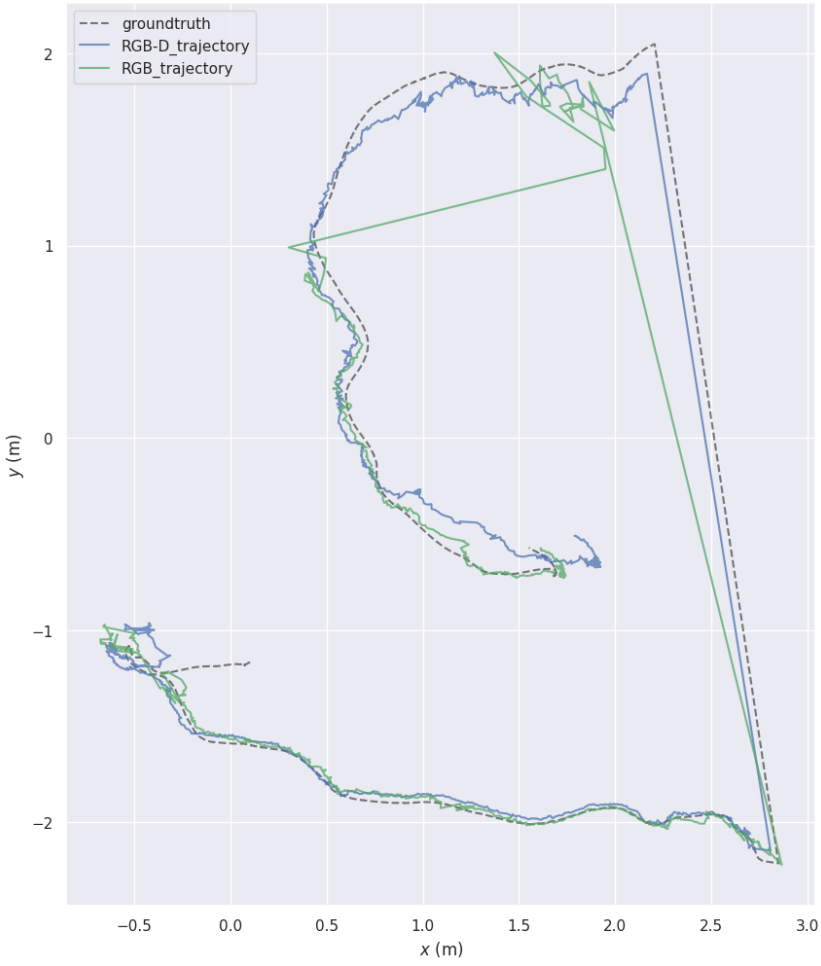

Figure 4.1: Trajectory plot results of ORB-SLAM2 when RGB and RGB-D images are used from the TUM RGB-D dataset

Figure 4.1 illustrates that ORB-SLAM2 achieves higher accuracy when both RGB and depth image sequences are utilized. In contrast, when the modality is restricted to RGB-only input, the estimated trajectory exhibits greater drift and susceptibility to error. In the monocular setup, the system temporarily loses tracking during the sequence. However, once repetitive frame features are encountered, the loop closure mechanism enables the system to relocalize and reconnect the trajectory. This is typically visualized as a straight line bridging the gap between the disjointed segments, indicating the reestablishment of spatial consistency despite prior localization failure. This behavior highlights the limitations of using a simple monocular configuration and underscores the benefit of incorporating depth information, which contributes to improved scale estimation and overall robustness of the SLAM system. The following Table 4.1 shows the ATE and RTE in both scenarios.

Modality	**ATE(m)**			**RTE (%)**		
	Mean	Min	Max	Mean	Min	Max
RGB images	1.069	0.416	2.761	0.012	0.000	3.716
RGB & Depth images (RGB-D)	0.111	0.007	0.636	0.017	0.001	0.371

Table 4.1: Translation and rotation errors for different sensor modalities on the TUM RGB-D dataset

4.1.2 TUM RGB-D Sequence 2:

freiburg2_large_no_loop sequence was also recorded using a handheld RGB-D sensor (Microsoft Kinect) in the same industrial hall but follows a different trajectory. Unlike the previous sequence, this one does not include any intentional loop closures, which makes it suitable for analyzing the performance of visual SLAM algorithms in continuous exploration tasks without opportunities for global correction.

Figure 4.2 presents the estimated trajectory of ORB-SLAM2 comparing RGB with RGB-D input. The system demonstrates stable tracking performance throughout the sequence despite the absence of loop closure opportunities. However, without

loop closure, minor accumulated drift becomes visible, particularly on the straight line part of the trajectory where features are not soo rich.

In contrast, when only RGB input was used for this sequence, ORB-SLAM2 failed to maintain tracking and lost localization entirely just after few seconds. This highlights a significant limitation of monocular setups in long, continuous trajectories without revisits. The results in the Table 4.2 affirm that depth sensing significantly enhances robustness and is particularly valuable in sequences lacking loop closures.

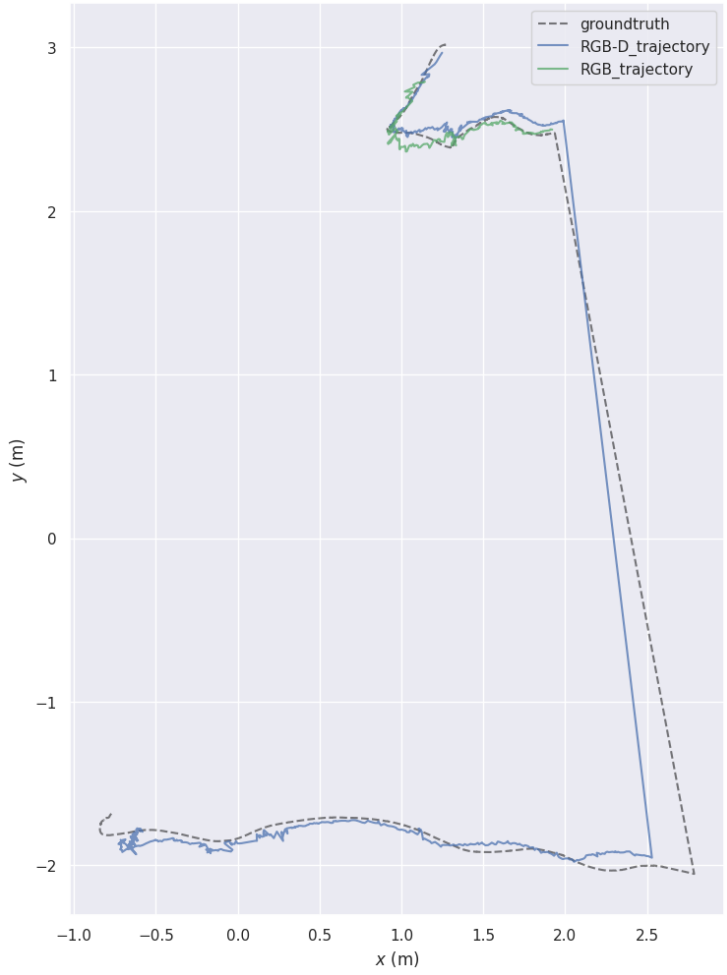

Figure 4.2: Trajectory plot results of ORB-SLAM2 when RGB and RGB-D images are used from the TUM RGB-D dataset with no loop

This is expected, as SLAM systems often rely on loop closure to minimize long-term errors. The reported ATE mean value of approximately 0.16 m and RTE of

Modality	**ATE(m)**			**RTE (%)**		
	Mean	Min	Max	Mean	Min	Max
RGB images	Tracking Lost (TL)					
RGB & Depth images (RGB-D)	0.161	0.063	0.295	0.0161	0.001	0.338

Table 4.2: Comparision of ATE and RTE for different sensor modalities on the TUM RGB-D dataset sequence 2

0.0161% reflect acceptable performance in feature-rich indoor environments.

The results from both the sequences clearly marks an advantage of using depth information in visual SLAM systems and thus draws a foundation for the Finnforest dataset where we had an opportunity to choose between monocular and stereo modes, but the results above clarifies that using stereo imagery can be helpful in obtaining depth information and thus better accuracy of results against the ground truth; thus, we use stereo mode in all our sequences in the Finnforest dataset from now on.

4.2 Finnforest sequence results

4.2.1 Finnforest Sequence S01 and W01:

Sequences S01 and W01 represent the same elliptical trajectory in the Finnforest dataset but were recorded in different seasons, S01 in summer and W01 in winter. This allows for a controlled comparison of seasonal effects on visual SLAM performance while maintaining consistent spatial structure and loop closures. Both sequences feature a closed-loop path with two forward-loop closures and one from the reverse direction, intended to assess relocalization and loop detection reliability.

Figure 4.3 depicts the results of ORB SLAM2 in S01 and W01. Groundtruth is named as reference. In S01, the summer foliage provided a relatively dense and varied texture, which helped ORB-SLAM2 achieve successful tracking throughout the route. Although local drifts were observed, global alignment was significantly

improved through loop closures. In contrast, W01 posed more challenging visual conditions due to snow cover, lower contrast, and a generally more uniform texture, leading to reduced feature richness.

Sequence	**ATE (m)**			**RTE (%)**		
	Mean	Min	Max	Mean	Min	Max
S01 - ORB-SLAM2	8.226	0.000	17.096	0.0185	0.00007	0.0769
W01 - ORB-SLAM2	7.224	0.000	13.529	0.0205	0.00005	1.1999
S01 & W01 - ORB-SLAM3	Tracking Lost (TL)					

Table 4.3: Comparison of ATE and RTE metrics for Sequences S01 and W01 in the Finnforest dataset across different seasons.

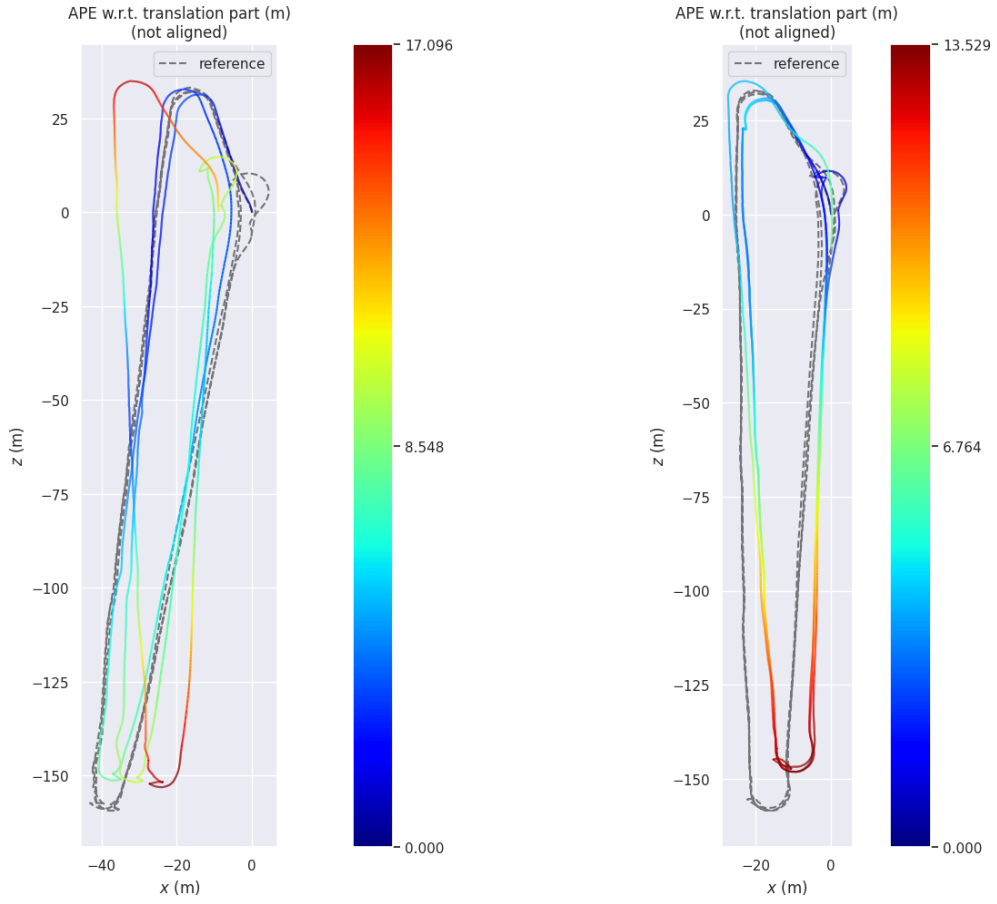

Figure 4.3: Trajectory plot results of ORB-SLAM2 vs ground truth for Finnforest sequences. (S01 on the left and W01 on the right)

Despite this, ORB-SLAM2 still completed the W01 loop with lower drift of mean ATE of 7.224m compared to S01 of 8.226m as shown in table 4.3, possibly due to

reduced motion blur and the sun glare in various situations that can accumulate drift that challenges localization. ORB-SLAM3 failed to maintain consistent tracking in all Finnforest sequences, leading to complete tracking loss and the main reasons for this are explained in the next Chapter 5.1 Discussion of results.

4.2.2 Finnforest Sequence S02:

Unlike the S01 and W01 sequences, S02 features a single-loop trajectory completed only once without revisiting the path. The environment includes moderate forest coverage and uneven terrain but benefits from good lighting conditions due to the summer foliage. Figure 4.4 shows that ORB-SLAM2 was able to complete the sequence without tracking loss. However, the system experienced significant drift, as evident in the ATE, with a mean error of 30.40 meters and a maximum reaching 53.47 meters. This suggests that, while the system was able to localize and close the loop, the estimation was far from the actual ground truth. The drift is most likely due to the lack of strong geometric features and the absence of repeated passes reducing the opportunities for correction.

The RTE for ORB-SLAM2 in this sequence was comparatively better as shown in the Table 4.4, with a mean of 0.036 meters, indicating that local pose consistency was still maintained despite larger global drift. ORB-SLAM3 failed to track in this sequence.

Sequence	**ATE (m)**			**RTE (%)**		
	Mean	Min	Max	Mean	Min	Max
S02 - ORB-SLAM2	30.403	0.000	53.473	0.0361	0.00002	0.3401
S02 - ORB-SLAM3	Tracking Lost (TL)					

Table 4.4: Evaluation of ATE and RTE metrics for Finnforest Sequence S02 using ORB-SLAM2 and ORB-SLAM3.

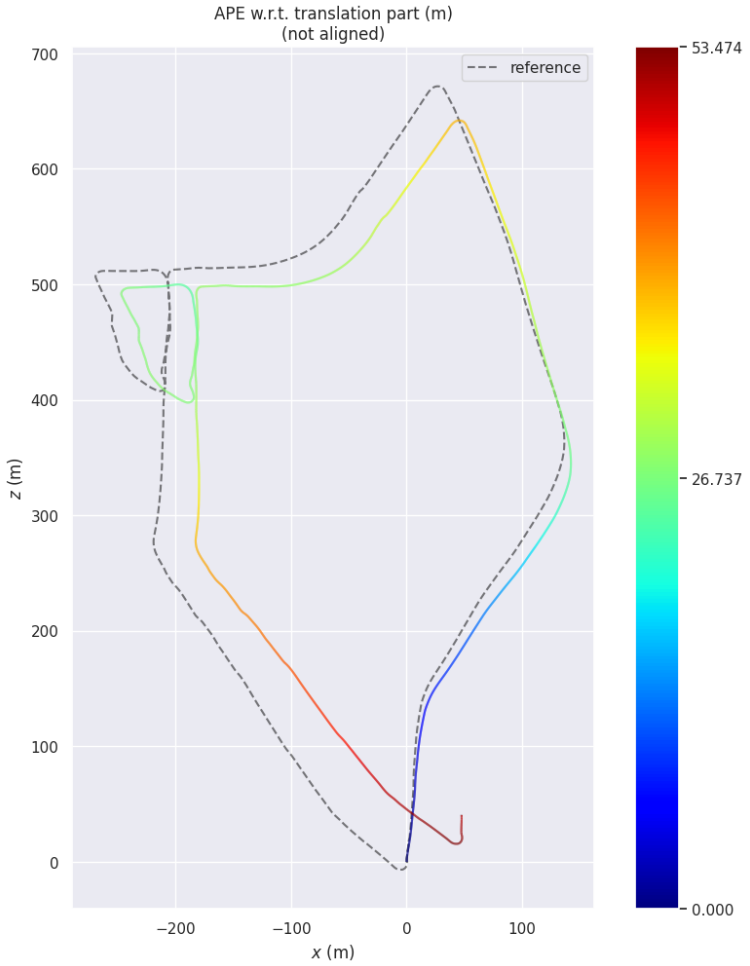

Figure 4.4: Trajectory plot results of ORB-SLAM2 against ground truth in S02 sequence

4.2.3 Finnforest Sequence S03:

Sequence S03 belongs to the group of visual odometry sequences in the Finnforest dataset and represents the shortest trajectory among them. It was designed without same-direction loop closures but does allow relocalization opportunities by revisiting the same path from the opposite direction. This design reflects realistic movement patterns of heavy machinery performing exploratory tasks in forest environments.

Figure 4.5 shows the estimated trajectory of ORB-SLAM2 in comparison with the ground truth. Despite the short distance of the route, ORB-SLAM2 experienced considerable global drift. This is reflected in the high ATE mean of 27.25 meters

and a maximum error of 49.09 meters as shown in the Table 4.5. The RTE was more stable, with a mean of 0.043 meters, suggesting that local pose estimation was relatively consistent. This discrepancy indicates that while ORB-SLAM2 can handle frame-to-frame tracking reasonably well, it struggles to maintain global consistency without reliable loop closures or strong structural cues.

In contrast, ORB-SLAM3 was unable to maintain reliable tracking throughout the entire S03 sequence. While it managed to reconstruct a portion of the trajectory during the initial segment of the loop as shown in Figure 4.6, tracking was lost at a sharp turn midway through the route. The system failed to recover localization in the latter half, and neither single-map tracking nor post-processing with the multi-map merging functionality succeeded in producing a complete and coherent trajectory.

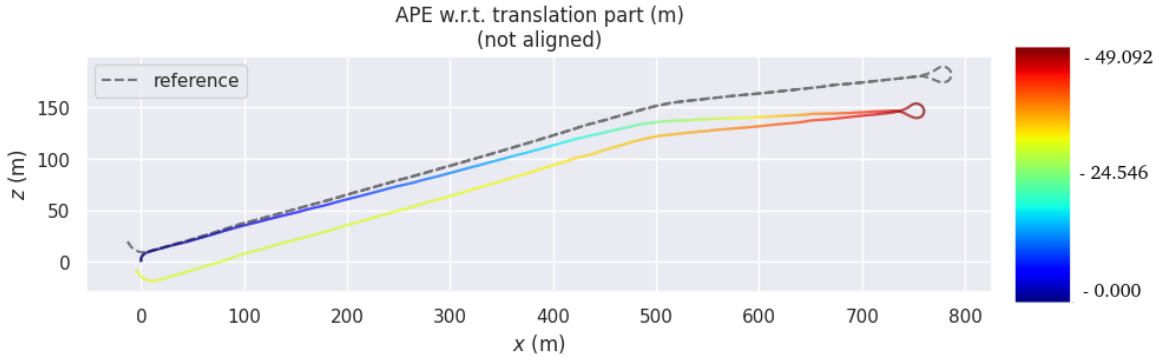

Figure 4.5: Trajectory plot results of ORB-SLAM2 against ground truth in S03 sequence

Sequence	**ATE (m)**			**RTE (%)**		
	Mean	Min	Max	Mean	Min	Max
S03 - ORB-SLAM2	27.255	0.000	49.092	0.0427	0.0002	0.5496
S03 - ORB-SLAM3	Tracking Lost (TL)					

Table 4.5: Evaluation of ATE and RTE metrics for Finnforest Sequence S03 using ORB-SLAM2 and ORB-SLAM3.

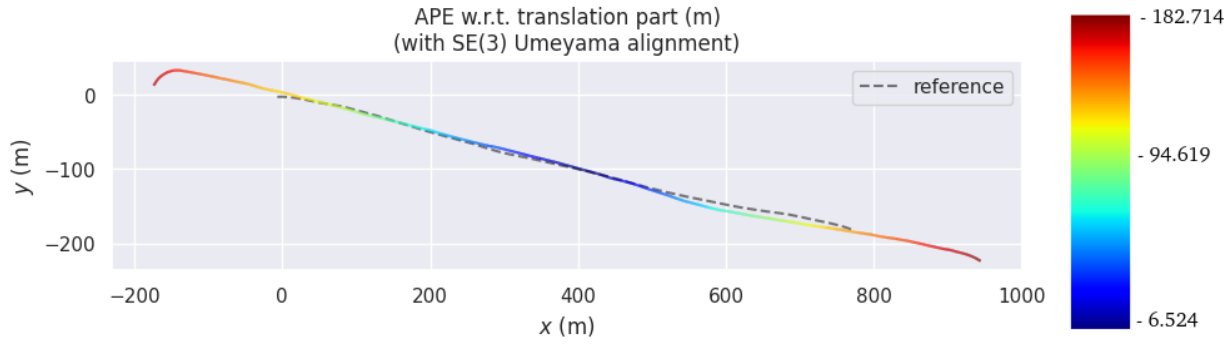

Figure 4.6: Trajectory plot results of ORB-SLAM3 against ground truth in S03

4.2.4 Finnforest Sequence W03:

Sequence W03 is part of the winter visual odometry set and mirrors the same trajectory layout as Sequence S03 but under different seasonal conditions. Despite the winter setting, which introduces additional visual challenges such as low texture, uniform snow-covered surfaces, and subdued contrast, ORB-SLAM2 was able to complete the sequence. However, significant drift was observed as shown in Figure 4.7, particularly along straight segments of the route. Once the trajectory deviated from the true path, the error continued to accumulate over distance, leading to a substantial offset. This effect became most pronounced after approximately 800 meters, where the drift along the X-axis reached its maximum. The Absolute Trajectory Error (ATE) showed a mean of 40.56 meters and a maximum error of 74.16 meters, as summarized in Table 4.6. Such drift likely results from the scarcity of distinct visual features and reliable depth cues in snow-dominated scenes—factors essential for accurate and consistent feature-based tracking in visual SLAM systems.

On the other hand, RTE remained within acceptable limits, with a mean of 0.035 meters. This suggests that while ORB-SLAM2 maintained reasonably consistent short-term tracking between frames, the global accuracy deteriorated progressively

due to error accumulation and the absence of loop closures.

ORB-SLAM3 failed to produce any meaningful trajectory for this sequence, again suffering from complete tracking loss under the given winter conditions.

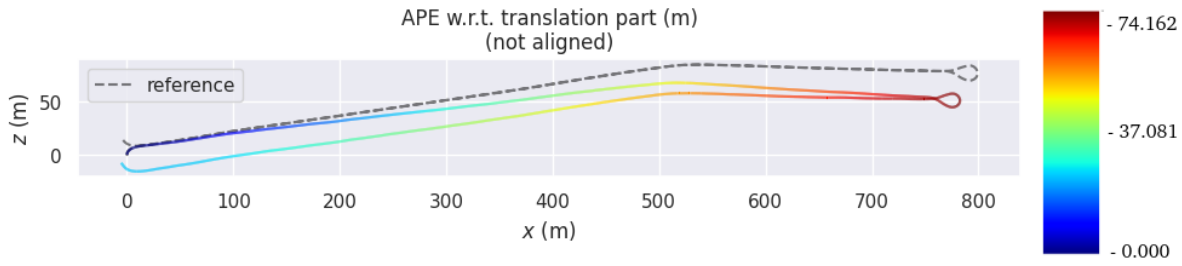

Figure 4.7: Trajectory plot results of ORB-SLAM2 against ground truth in W03 sequence

Sequence	**ATE (m)**			**RTE (%)**		
	Mean	Min	Max	Mean	Min	Max
W03 - ORB-SLAM2	40.561	0.000	74.162	0.0348	0.0003	0.0859
W03 - ORB-SLAM3	Tracking Lost (TL)					

Table 4.6: Evaluation of ATE and RTE metrics for Finnforest Sequence W03 using ORB-SLAM2 and ORB-SLAM3.

The next chapter presents a detailed discussion and interpretation of the results in Section 5.1. Section 5.2 addresses the research questions guiding this study. Subsequently, Section 5.3 outlines the limitations encountered during the research, while Section 5.4 proposes directions for future work.

5 Discussion

This chapter presents a critical analysis of the results obtained in Chapter 4, with a focus on evaluating the performance of state-of-the-art visual SLAM systems when applied to unstructured forest environments. Section 5.1 interprets the experimental findings, identifies key challenges encountered during deployment, and explores the underlying causes of tracking failures and drift. Section 5.2 addresses the main research questions posed in this thesis, synthesizing insights derived from experimental findings. The final two sections discusses the limitations and prospective research directions that aims to improve the robustness and adaptability of SLAM systems in complex outdoor environments.

5.1 Discussion of results

When running ORB-SLAM2 on the Finnforest dataset, the number of ORB features was uniformly set to 1500 across all sequences to maintain consistency and reduce computational burden. Surprisingly, this setting yielded satisfactory results, even in the more visually complex and unstructured outdoor environments. However, ORB-SLAM3 failed to maintain robust tracking in all sequences, regardless of the increased feature count set at as high as 3500 in the configuration file.

The ATE and RTE results, when compared between the TUM RGB-D dataset and the Finnforest sequences, highlight a stark contrast in performance. In the structured, feature-rich indoor settings of the TUM dataset, ORB-SLAM2 demonstrates

excellent accuracy with low drift and stable tracking. In contrast, its performance deteriorates in the unstructured and unpredictable forest settings, as evidenced by the increased drift and tracking errors. Each Finnforest sequence further provides unique insight into how SLAM systems respond to specific environmental conditions. For instance, the sequences S01 and W01, both featuring a loop-based trajectory, show that the availability of repeated visual cues and relocalization opportunities significantly reduces overall drift. In these cases, ORB-SLAM2 successfully closes loops and distributes error effectively across the trajectory. Interestingly, the seasonal variation between S01 (summer) and W01 (winter) did not lead to a dramatic difference in ATE or RTE. This observation suggests that, despite differences in lighting, loop closure has a more substantial impact on SLAM performance than seasonal factors alone. However, winter conditions, with more uniform textures due to snow may result in slightly lower feature richness, which could affect tracking under less controlled circumstances.

In contrast, sequences like S02, S03, and W03 lack robust loop closures and rely more heavily on continuous visual odometry. These sequences showed significantly higher ATE values, revealing the limitations of monocular or stereo SLAM systems in environments without revisitable locations. For example, S02, despite being a closed loop, exhibited a mean ATE of over 30 meters, indicating poor global consistency. Similarly, W03 exhibited the highest ATE in the dataset, reaching a mean of over 40 meters, further confirming that lack of loop closure and variable lighting (such as low sun angle or shadows) impair SLAM robustness. Nonetheless, the RTE values remained relatively low across these sequences, implying that short-term motion estimation remained consistent even though long-term drift accumulated. Additionally, factors such as motion blur, exposure inconsistency, and erratic vehicle motion on uneven terrain introduced further tracking challenges.

ORB-SLAM3's failure in all Finnforest sequences, even with favorable parameter

tuning, emphasizes its current incompatibility with natural environments. One of the critical weaknesses observed was its inability to recover from tracking loss after sharp turns or scene transitions. For example, in sequence S03, the system managed to track part of the trajectory but completely failed during a sharp curve in the middle of the loop, after which it was unable to relocalize or merge submaps. Figure 5.1 illustrates several instances where ORB-SLAM3 fails to maintain localization or tracking. These failures correspond to abnormal drops in the number of feature matches between consecutive image frames, falling below 50, causing the system to lose track of its trajectory and ultimately fail in localizing the camera within the environment. Whereas, ORB-SLAM2 generally sustains a higher number of matches per frame, often exceeding 50, thereby preserving trajectory continuity in most cases unless there is a real localization issue just like we have seen in Sequence 02 of TUM RGB-D dataset where tracking was completely lost.

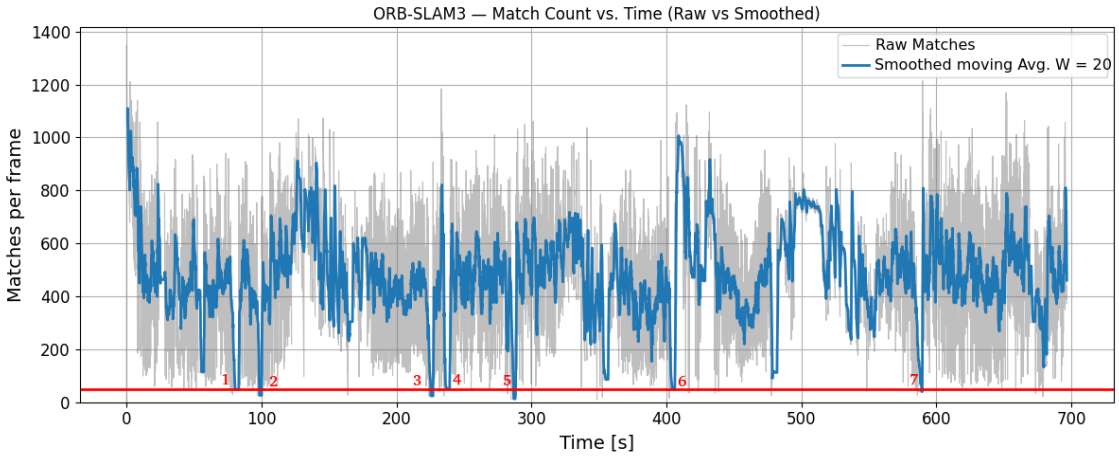

Figure 5.1: Match count vs. Time plot showing why the ORB-SLAM3 is failing in S03 sequence, as the matches drop

This consistent failure across different parameter settings raises concerns about ORB-SLAM3's stability and adaptability in natural environments, particularly in forested terrains with frequent occlusions, illumination variations, and irregular motion patterns.

A key factor contributing to the failure of ORB-SLAM3, in contrast to the more robust performance of ORB-SLAM2, lies in its tightly coupled system architecture. ORB-SLAM3 is inherently optimized for visual-inertial (VI) configurations, where inertial measurements aid in maintaining stable tracking. However, since the Finnforest dataset provides only stereo image sequences without inertial data, the absence of such complementary inputs limits ORB-SLAM3's ability to perform reliably. Moreover, the lack of publicly available VI datasets in natural environments further restricts the opportunity to evaluate and improve its performance under such conditions. ORB-SLAM3 also demands higher-quality and more consistent feature correspondences between the image frames to sustain robust tracking. In scenarios where these stringent requirements are not fulfilled the system is prone to initialization failure or frequent tracking loss and creating multiple maps. In contrast, ORB-SLAM2 exhibits greater tolerance under such conditions since it doesn't support multi map support it attempts to recover and resume tracking within the same map, rather than fragmenting the trajectory. This behavior enables a more coherent trajectory estimation and makes the system more resilient to temporary failures in feature matching or localization.

This work suggests that unless significant feature matching techniques and backend optimizations are introduced, the feasibility of using ORB-SLAM3 in such environments remains limited. This is backed by the fact mentioned in the original work by Campos et al.[23] in the conclusion section that ORB-SLAM3 struggles in low texture scenarios. Forests and natural environments are case of this scenario.

Overall, the results confirm that while state-of-the-art visual SLAM systems like ORB-SLAM2 and ORB SLAM-3 perform well in indoor or structured environments, their direct application in natural forest settings leads to significant challenges. These include difficulty in maintaining scale, increased trajectory drift, and inability to recover from occlusions or erratic motion. Loop closure remains a criti-

cal component for controlling drift, while seasonal variation, though not dominant, adds complexity in terms of illumination and texture.

These findings highlight the need for further development in SLAM algorithms that are resilient to real-world outdoor dynamics, particularly in forestry, agriculture, and autonomous off-road robotics.

5.2 Addressing the Research Questions

RQ1: While ORB-SLAM2 and ORB-SLAM3 remain among the state-of-the-art SLAM frameworks in indoor and structured environments, they face significant challenges when deployed in natural, unstructured settings. These limitations highlight the need for enhanced robustness, which can potentially be achieved through multimodal sensor fusion and the integration of adaptable backend architectures. Furthermore, incorporating learning-based approaches offers promising avenues for improving performance in complex and dynamic outdoor environments.

RQ2: One critical limitation observed in ORB-SLAM3 Algorithm was its strict dependency on a consistently high number of feature matches. This rigid requirement made the system highly sensitive to environments with low texture or frequent occlusions—common in forest scenarios—ultimately causing tracking failures in all Finnforest sequences. In contrast, frameworks like ORB-SLAM2 demonstrated relatively more resilience due to a more flexible frontend.

To address such limitations, future SLAM systems should allow dynamic adjustment of feature thresholds based on environmental conditions, enabling more graceful degradation rather than complete failure. Additionally, integrating a learning-based frontend could significantly enhance adaptability. Neural feature extractors have shown potential in robustly handling varying textures, lighting conditions, and structural ambiguities, thereby providing a more reliable and adaptable input pipeline for SLAM systems operating in unstructured, outdoor environments.

RQ3: Monocular SLAM systems, while lightweight and cost-effective, struggle in forest environments due to scale ambiguity and sensitivity to low-texture, repetitive patterns. Without reliable loop closures or structural cues, they frequently lose tracking. In contrast, stereo SLAM systems offer improved robustness by providing depth through disparity, which enhances pose estimation even in texture-homogeneous regions.

However, stereo setups are not immune to challenges. Dense foliage, dynamic elements like moving branches, and changes in lighting can still cause drift and mapping inconsistencies. Experimental results from the TUM RGB-D and Finnforest datasets confirm that stereo SLAM performs better than monocular in natural settings, yet both have their limitations. These findings highlight the need for multimodal approaches, such as incorporating thermal or event cameras, to enhance reliability and performance in complex outdoor environments.

RQ4: While passive SLAM systems those based on stereo or RGB-D cameras, have demonstrated impressive performance in indoor and controlled settings, their capabilities fall short in natural, unstructured environments. Compared to active sensors like LiDAR, passive systems show limited reliability in maintaining consistent loop closures and are more prone to long-term drift, especially when operating without rich textures or in dynamic lighting conditions.

Active systems inherently benefit from accurate depth sensing and greater robustness to environmental variability, which results in superior mapping accuracy and long-term localization stability. Passive systems can approximate this performance in certain scenarios but generally require careful tuning, ideal lighting, and structured features. Bridging the performance gap would require hybrid systems or learning based passive SLAM pipelines that can adaptively compensate for visual ambiguities and poor feature distributions.

5.3 Limitations

This study encountered several limitations that should be acknowledged. Firstly, the availability and accessibility of open-source SLAM frameworks suitable for outdoor forest environments remain restricted. Consequently, this thesis focuses on ORB-SLAM2 and ORB-SLAM3, both of which are well-documented, widely-used, and publicly available state-of-the-art frameworks. While alternative SLAM methods may potentially perform well in forest settings, the lack of open-source implementations and comprehensive documentation limited their consideration in this study.

Secondly, the scarcity of publicly available datasets that support multisensor fusion and are tailored explicitly for unstructured outdoor environments presents a significant challenge. Such datasets are essential for developing and benchmarking advanced SLAM techniques that integrate multiple sensor modalities to improve accuracy and robustness. The limited dataset availability constrained the scope of this research in exploring multisensor fusion approaches comprehensively.

Furthermore, deep learning-based SLAM methods, which represent a promising direction in robotics, demand substantial computational resources and extensive training data. These requirements hinder their practical deployment in real-time scenarios within complex outdoor environments. Accordingly, this thesis excludes deep learning SLAM techniques due to these prohibitive computational and data demands.

Additionally, implementing multisensor fusion involves several sensors that can perceive the surrounding details in different dimensions. Whether it be thermal cameras or event-based cameras deployed alongside of RGB stereo cameras demands SLAM frameworks capable of supporting these modalities. For example, the M2P2 dataset[12], despite its richness, could not be fully utilized due to the absence of compatible SLAM methods that can simultaneously process all sensor inputs. Moreover, the dataset is provided primarily in ROS bag format, which limits compatibility with

certain open-source SLAM implementations.

Finally, reliance on the Robot Operating System (ROS) for managing sensor data streams introduces performance constraints. ROS systems must publish sensor data from multiple modalities for SLAM frameworks to subscribe and process. This data transmission can introduce latency, especially when handling high-bandwidth data streams, such as those found in the Finnforest dataset[11], where each image frame is approximately 4 megabytes and frames are produced at 40 Hz. High data throughput may degrade system performance and cause occasional failures, thus limiting the effectiveness of ROS-based SLAM implementations in such scenarios.

5.4 Future Work

As a necessity, Future work should prioritise dataset construction of multi-kilometre forest sequences that include time-synchronised stereo, IMU, and ideally thermal or event streams to facilitate sensor-fusion research. Fusing stereo, thermal, and event pipelines add robustness without LiDAR-level power draw. The most publicly available outdoor datasets were recorded outside Finland in comparatively mild conditions. They therefore sidestep the dense snow cover, low-sun illumination, and sub-zero temperatures that can severely degrade feature tracking and inertial calibration. A Finnish data set would fill this gap and provide a more demanding benchmark for algorithms intended to operate in harsh boreal environments.

A possible direction to improve the robustness of feature-based Visual SLAM systems in complex and unstructured environments, such as forests, is the application of Delaunay triangulation techniques. Visual SLAM systems such as ORB-SLAM2 and ORB-SLAM3 rely heavily on repeatable feature detection and matching over time. However, in natural environments with dynamic lighting and ambiguous textures, the churn rate of ORB features (i.e. frequent replacement of tracked keypoints) can compromise both localization accuracy and map consistency. By integrating Delau-

nay triangulation into the feature association pipeline, it may be possible to enforce geometric constraints that preserve the spatial relationships between features across frames.

This concept has been explored by Li et al.[87], who proposed a localization framework for autonomous robots operating in forests, utilizing Delaunay triangulation to match local point cloud structures with a global tree map. Rather than relying solely on direct point cloud matching, their method leverages topological similarities between triangular structures formed by tree trunk positions, yielding impressive localization accuracy (standard deviation of 12 cm) under field conditions in Finnish forests [87]. While their work focuses on LiDAR-based mapping, the underlying principle, using geometric topology for improving map consistency, could be extended to Visual SLAM systems. Incorporating triangulated spatial priors or constraints during ORB feature matching could potentially reduce drift and enhance loop closure detection in dense natural scenes.

Given the consistent failure of ORB-SLAM3 in low-texture environments [23], future research should focus on improving the robustness of SLAM systems under such challenging conditions. One promising direction is the integration of direct methods, which rely on photometric information rather than discrete feature points. Although direct methods tend to be more resilient in low-texture scenarios, their use is currently limited to short- and mid-term tracking due to difficulty in long-term and multi-map data association.

A hybrid approach that combines the long-term consistency of feature-based matching with the short-term robustness of direct photometric methods could offer a more resilient solution. For example, leveraging Lucas–Kanade-based optical flow for real-time tracking and integrating sparse feature descriptors selectively for loop closures and relocalization may help balance performance.

Additionally, further exploration into photometric techniques that support all

four levels of data association namely short-term, mid-term, long-term, and multimap is highly recommended. These techniques must be adapted to the complexities of natural scenes, where lighting variation, texture sparsity, and motion irregularities are prevalent.

Future work may also investigate learning-based or semantic SLAM techniques that can infer scene structure and motion in low-texture environments, possibly using neural networks trained on synthetic or real forest datasets.

Lastly, improving the backend optimization, such as better keyframe selection strategies or adaptive feature thresholds, may help make ORB-SLAM3 more robust in natural, low-featured outdoor environments like forests, thus extending its applicability beyond structured indoor or urban settings.

Addressing these items will close the gap between active-sensor accuracy and the low-power, passive sensing needed for long-duration forest navigation that aids stealth operations.

6 Conclusion

This thesis investigated the feasibility of deploying modern visual SLAM systems specifically ORB-SLAM2 and ORB-SLAM3 in forested environments characterized by dense vegetation, variable illumination, and challenging terrain. The primary research objective was to assess whether these state-of-the-art methods, originally designed for structured indoor or urban settings, can be effectively extended to natural, unstructured environments.

Despite significant advancements in SLAM over the past decade, a clear research gap remains in adapting visual SLAM for forested terrains. Most field robotics applications in such settings still depend on LiDAR or other active sensors. While accurate, these systems are energy-intensive and may interfere with wildlife, making them less ideal for long-term or stealth operations. In contrast, passive vision-based systems offer a more lightweight and non-invasive alternative, but their reliability under forest conditions has not been thoroughly validated.

To bridge this gap, ORB-SLAM2 and ORB-SLAM3 were implemented and evaluated on the FinnForest dataset, which provides stereo image sequences collected in various seasonal and lighting conditions. However, the lack of synchronized inertial measurements in this dataset limited the evaluation of ORB-SLAM3, which is inherently designed for tightly coupled visual–inertial SLAM. Experiments reveal that ORB-SLAM3 frequently lost tracking and failed to recover, particularly in low-texture, high-occlusion scenes typical of forests. ORB-SLAM2, by contrast,

demonstrated better performance, often managing to re-localize and complete sequences, though with notable drift accumulation over time. Results also highlighted the importance of loop closures and feature richness, introduced motion blur or overexposure.

Ultimately, this study underscores that while traditional visual SLAM pipelines such as ORB-SLAM2 can still function in forested environments, their performance remains limited, especially over long distances or in dynamic outdoor conditions. ORB-SLAM3's reliance on inertial data makes it less applicable unless paired with dedicated hardware and datasets.

Further research should explore hybrid approaches. Promising directions include integrating learning-based feature extraction and depth estimation into traditional SLAM pipelines, and optimizing the backend for real-time operation under constrained resources. Developing Multi modal passive vision based data that includes thermal imagery, event cameras and IMU fused forest datasets and adapting SLAM systems to better handle low-texture and high-occlusion scenes will also be critical for real-world deployment. And ultimately addressing computational demands through the use of powerful yet energy-efficient processors (e.g., embedded GPUs, neuromorphic chips) will be critical to deploying these systems on autonomous platforms operating in real-world, forest scenarios.

In conclusion, while current visual SLAM systems show partial success in forests, their limitations call for the next generation of adaptive, learning-enhanced SLAM methods tailored to the complexities of natural environments.

References

[1] M. T. Ohradzansky, E. R. Rush, D. G. Riley, *et al.*, "Multi-agent autonomy: Advancements and challenges in subterranean exploration", *Field Robotics*, vol. 2, pp. 1068–1104, 2022.

[2] S. Thrun, "Simultaneous localization and mapping", in *Robotics and cognitive approaches to spatial mapping*, Springer, 2008, pp. 13–41.

[3] C. Fan, Z. Li, W. Ding, H. Zhou, and K. Qian, "Integrating artificial intelligence with slam technology for robotic navigation and localization in unknown environments", *Applied and Computational Engineering*, vol. 77, pp. 245–250, 2024.

[4] N. M. Nadkarni, R. O. Lawton, K. L. Clark, T. J. Matelson, and D. Schaefer, "Ecosystem ecology and forest dynamics", *Monteverde: ecology and conservation of a tropical cloud forest. Oxford University Press, New York*, pp. 303–350, 2000.

[5] A. Wilson, K. A. Gupta, B. H. Koduru, A. Kumar, A. Jha, and L. R. Cenkeramaddi, "Recent advances in thermal imaging and its applications using machine learning: A review", *IEEE Sensors Journal*, vol. 23, no. 4, pp. 3395–3407, 2023.

[6] A. LaMarca and E. De Lara, *Location systems: An introduction to the technology behind location awareness*. Springer Nature, 2022.

[7] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part i", *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006. DOI: 10.1109/MRA.2006.1638022.

[8] C. Cadena, L. Carlone, H. Carrillo, *et al.*, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age", *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016. DOI: 10.1109/TRO.2016.2624754.

[9] F. A. Cheein, G. Scaglia, F. di Sciasio, and R. Carelli, "Feature selection criteria for real time ekf-slam algorithm", *International Journal of Advanced Robotic Systems*, vol. 6, no. 3, p. 21, 2009.

[10] F. Schmidt, C. Blessing, M. Enzweiler, and A. Valada, "Visual-inertial slam for agricultural robotics: Benchmarking the benefits and computational costs of loop closing", *arXiv preprint arXiv:2408.01716*, 2024.

[11] I. Ali, A. Durmush, O. Suominen, *et al.*, "Finnforest dataset: A forest landscape for visual slam", *Robotics and Autonomous Systems*, vol. 132, p. 103 610, 2020. DOI: 10.1016/j.robot.2020.103610.

[12] A. Datar, A. Pokhrel, M. Nazeri, *et al.*, "M2p2: A multi-modal passive perception dataset for off-road mobility in extreme low-light conditions", *arXiv preprint arXiv:2410.01105*, 2024.

[13] A. Elfes, "Using occupancy grids for mobile robot perception and navigation", *Computer*, vol. 22, no. 6, pp. 46–57, 1989. DOI: 10.1109/2.30720.

[14] D. Schubert, N. Demmel, V. Usenko, *et al.*, "The tum VI benchmark for evaluating visual–inertial odometry", *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1680–1687, 2018. DOI: 10.1109/IROS.2018.8593654.

[15] G. E. M. Abro, S. A. B. Zulkifli, R. J. Masood, V. S. Asirvadam, and A. Laouiti, "Comprehensive review of uav detection, security, and communication advancements to prevent threats", *Drones*, vol. 6, no. 10, p. 284, 2022.

[16] Y. Wang, Y. Tian, J. Chen, K. Xu, and X. Ding, "A survey of visual slam in dynamic environment: The evolution from geometric to semantic approaches", *IEEE Transactions on Instrumentation and Measurement*, 2024.

[17] K. Ebadi, M. Palieri, S. Wood, C. Padgett, and A.-a. Agha-mohammadi, "Dare-slam: Degeneracy-aware and resilient loop closing in perceptually-degraded environments", *Journal of Intelligent & Robotic Systems*, vol. 102, pp. 1–25, 2021.

[18] K. Eckenhoff, P. Geneva, and G. Huang, "Mimc-vins: A versatile and resilient multi-imu multi-camera visual-inertial navigation system", *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1360–1380, 2021.

[19] M. Tranzatto, T. Miki, M. Dharmadhikari, *et al.*, "Cerberus in the darpa subterranean challenge", *Science Robotics*, vol. 7, no. 66, eabp9742, 2022.

[20] M. Lyu, Y. Zhao, C. Huang, and H. Huang, "Unmanned aerial vehicles for search and rescue: A survey", *Remote Sensing*, vol. 15, no. 13, p. 3266, 2023.

[21] P.-Y. Lajoie and G. Beltrame, "Swarm-slam: Sparse decentralized collaborative simultaneous localization and mapping framework for multi-robot systems", *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 475–482, 2023.

[22] S. R. Soare, "What if... the military ai of nato and eu states is not interoperable?", *What If... Not*, pp. 18–22, 2021.

[23] C. Campos, R. Elvira, J. J. Gómez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and

multi-map SLAM", *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021. DOI: 10.1109/TRO.2021.3075644.

[24] A. Gupta and X. Fernando, "Simultaneous localization and mapping (slam) and data fusion in unmanned aerial vehicles: Recent advances and challenges", *Drones*, vol. 6, no. 4, p. 85, 2022.

[25] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): Part ii—state of the art", *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 108–117, 2006. DOI: 10.1109/MRA.2006.1678147.

[26] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras", *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017. DOI: 10.1109/TRO.2017.2705103.

[27] G. Grisetti, C. Stachniss, and W. Burgard, "A tutorial on graph-based SLAM", *IEEE Intelligent Transportation Systems Magazine*, vol. 2, no. 4, pp. 31–43, 2010. DOI: 10.1109/MITS.2010.939928.

[28] A. J. Davison, "Real-time simultaneous localization and mapping with a single camera", in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2003, pp. 1403–1410. DOI: 10.1109/ICCV.2003.1238654.

[29] B. Bescos, M. Fábrega, C. Rodríguez, J. Neira, and R. Mur-Artal, "Dynaslam: Tracking, mapping, and inpainting in dynamic scenes", in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2018, pp. 4265–4272. DOI: 10.1109/ICRA.2018.8460661.

[30] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real time", in *Robotics: Science and Systems (RSS)*, 2014. [Online]. Available: http://www.cs.cmu.edu/~jingjiez/papers/RSS2014_LOAM.pdf.

[31] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: A versatile and accurate monocular SLAM system", *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015. DOI: 10.1109/TRO.2015.2463671.

[32] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual–inertial state estimator", *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[33] Z. Zeng, X. Dang, Y. Li, Y. Huang, and X. Liang, "A mmwave radar slam method in subterranean tunnel for low visibility and degradation", *IEEE Robotics and Automation Letters*, 2024.

[34] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2d lidar slam", in *2016 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2016, pp. 1271–1278.

[35] T. Shan and B. Englot, "Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain", in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 4758–4765.

[36] A. Rashid, A. Kausik, A. Sunny, and M. Bappy, "Artificial intelligence in the military: An overview of the capabilities, applications, and challenges", *International Journal of Intelligent Systems*, vol. 2023, pp. 1–31, Nov. 2023. DOI: 10.1155/2023/8676366.

[37] M. Heshmat, L. Saad Saoud, M. Abujabal, *et al.*, "Underwater slam meets deep learning: Challenges, multi-sensor integration, and future directions", *Sensors*, vol. 25, no. 11, p. 3258, 2025.

[38] S. Foix, G. Alenya, and C. Torras, "Lock-in time-of-flight (tof) cameras: A survey", *IEEE Sensors Journal*, vol. 11, no. 9, pp. 1917–1926, 2011.

[39] B. D. S. Howarth, "Real time 3d mapping for small wall climbing robots", Ph.D. dissertation, UNSW Sydney, 2012.

[40] M. Shahraki, A. Elamin, and A. El-Rabbany, "Event-based visual simultaneous localization and mapping (evslam) techniques: State of the art and future directions", *Journal of Sensor and Actuator Networks*, vol. 14, no. 1, p. 7, 2025.

[41] A. S. Narouz, A. Ismail, A. Atef, *et al.*, "A review of features and characteristics of rescue robot with ai", *Advanced Sciences and Technology Journal*, vol. 1, no. 2, pp. 1–18, 2024.

[42] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005, ISBN: 978-0-262-20162-9.

[43] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF", in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2011, pp. 2564–2571. DOI: 10.1109/ICCV.2011.6126544.

[44] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with rao–blackwellised particle filters", in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2007, pp. 2432–2437. DOI: 10.1109/ROBOT.2007.363194.

[45] D. Gálvez-López and J. D. Tardós, "Bags of binary words for fast place recognition in image sequences", *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012. DOI: 10.1109/TRO.2012.2197158.

[46] Y. Xiao, W. Xu, B. Li, H. Zhang, B. Xu, and W. Zhou, "Research on visual–inertial measurement unit fusion simultaneous localization and mapping algorithm for complex terrain in open-pit mines", *Sensors*, vol. 24, no. 22, p. 7360, 2024.

[47] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry", *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2017.

[48] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, "Contact-aided invariant extended kalman filtering for robot state estimation", *The International Journal of Robotics Research*, vol. 39, no. 4, pp. 402–430, 2020.

[49] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G 2 o: A general framework for graph optimization", in *2011 IEEE international conference on robotics and automation*, IEEE, 2011, pp. 3607–3613.

[50] F. Dellaert, "Factor graphs and gtsam: A hands-on introduction", *Georgia Institute of Technology, Tech. Rep*, vol. 2, no. 4, 2012.

[51] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite", in *2012 IEEE conference on computer vision and pattern recognition*, IEEE, 2012, pp. 3354–3361.

[52] V. M. S. A. S. Datta, A. Ghosh, and V. D. S. D. Chakravarty, "Deepvo: A deep learning approach for monocular visual odometry",

[53] N. Yang, L. v. Stumberg, R. Wang, and D. Cremers, "D3vo: Deep depth, deep pose and deep uncertainty for monocular visual odometry", in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1281–1292.

[54] X. Xu, L. Zhang, J. Yang, *et al.*, "A review of multi-sensor fusion slam systems based on 3d lidar", *Remote Sensing*, vol. 14, no. 12, p. 2835, 2022.

[55] Y. Li, F. Zhun, Z. Guijie, *et al.*, "A slam with simultaneous construction of 2d and 3d maps based on rao-blackwellized particle filters", in *2018 tenth international conference on advanced computational intelligence (ICACI)*, IEEE, 2018, pp. 374–378.

[56] S. Kohlbrecher, O. Von Stryk, J. Meyer, and U. Klingauf, "A flexible and scalable slam system with full 3d motion estimation", in *2011 IEEE international symposium on safety, security, and rescue robotics*, IEEE, 2011, pp. 155–160.

[57] Y. Cong, C. Chen, B. Yang, *et al.*, "3d-cstm: A 3d continuous spatio-temporal mapping method", *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 186, pp. 232–245, 2022.

[58] G. Liu, K. Huang, X. Lv, *et al.*, "Innovations and refinements in lidar odometry and mapping: A comprehensive review", *IEEE/CAA Journal of Automatica Sinica*, 2025.

[59] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping", in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, IEEE, 2020, pp. 5135–5142.

[60] B. Đikić, "Automotive lidar technology for marine applications–determining and increasing the accuracy of simultaneous localisation and mapping", 2023.

[61] W. Xu and F. Zhang, "Fast-lio: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter", *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3317–3324, 2021.

[62] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry", *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2053–2073, 2022.

[63] F. Khalid, J. E. Pickering, and Z. Dai, "Using 2d lidar and rgb camera for human agnostic mapping", in *2024 29th International Conference on Automation and Computing (ICAC)*, IEEE, 2024, pp. 1–6.

[64] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam", in *European conference on computer vision*, Springer, 2014, pp. 834–849.

[65] J. Zubizarreta, I. Aguinaga, and J. M. M. Montiel, "Direct sparse mapping", *IEEE Transactions on Robotics*, vol. 36, no. 4, pp. 1363–1370, Aug. 2020, ISSN: 1941-0468. DOI: 10.1109/tro.2020.2991614. [Online]. Available: http://dx.doi.org/10.1109/TRO.2020.2991614.

[66] H. Rebecq, T. Horstschäfer, G. Gallego, and D. Scaramuzza, "Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real time", *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 593–600, 2016.

[67] A. R. Vidal, H. Rebecq, T. Horstschaefer, and D. Scaramuzza, "Ultimate slam? combining events, images, and imu for robust visual slam in hdr and high-speed scenarios", *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 994–1001, 2018.

[68] K. Celik, "Autocalibrating vision guided navigation of unmanned air vehicles via tactical monocular cameras in gps denied environments", Ph.D. dissertation, Iowa State University, 2012.

[69] A. Mallios, P. Ridao, D. Ribas, M. Carreras, and R. Camilli, "Toward autonomous exploration in confined underwater environments", *Journal of Field Robotics*, vol. 33, no. 7, pp. 994–1012, 2016.

[70] R. J. a. L. Hartley, I. L. Henderson, and C. L. Jackson, "Bvlos unmanned aircraft operations in forest environments", *Drones*, vol. 6, no. 7, p. 167, 2022.

[71] T. Shan, B. Englot, C. Ratti, and D. Rus, "Lvi-sam: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping", in *2021 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2021, pp. 5692–5698.

[72] N. Yang, R. Wang, J. Stuckler, and D. Cremers, "Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry", in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 817–833.

[73] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "Semanticfusion: Dense 3d semantic mapping with convolutional neural networks", in *2017 IEEE International Conference on Robotics and automation (ICRA)*, IEEE, 2017, pp. 4628–4635.

[74] H. Zhou, B. Ummenhofer, and T. Brox, "Deeptam: Deep tracking and mapping with convolutional neural networks", *International Journal of Computer Vision*, vol. 128, no. 3, pp. 756–769, 2020.

[75] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems", in *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[76] B. Stroustrup, *The C++ Programming Language*, 4th. Addison-Wesley, 2013.

[77] G. Bradski, "The opencv library", *Dr. Dobb's Journal of Software Tools*, 2000.

[78] G. Guennebaud and B. J. et al., *Eigen v3*, `http://eigen.tuxfamily.org`, 2010.

[79] H. Strasdat, *Sophus: C++ implementation of lie groups*, `https://github.com/strasdat/Sophus`, 2023.

[80] D. Galvez-Lopez and J. D. Tardós, "Bags of binary words for fast place recognition in image sequences", *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.

[81] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G2o: A general framework for graph optimization", in *Proc. IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2011, pp. 3607–3613.

[82] S. Lovegrove, *Pangolin: A lightweight portable rapid development library for managing opengl display / interaction and abstracting video input*, https://github.com/stevenlovegrove/Pangolin.

[83] *Cmake: Cross-platform build-system generator*, https://cmake.org.

[84] *Ninja: A small build system with a focus on speed*, https://ninja-build.org.

[85] M. Quigley, K. Conley, B. Gerkey, *et al.*, "Ros: An open-source robot operating system", in *ICRA Workshop on Open Source Software*, 2009.

[86] M. Grupp, R. Grimm, J. Schwendner, F. Schilling, and P. L. Fischer, *evo: Python package for the evaluation of odometry and SLAM*, version v1.14.0, Accessed: 2025-06-02, Jan. 2024. DOI: 10.5281/zenodo.10515152. [Online]. Available: https://github.com/MichaelGrupp/evo.

[87] Q. Li, P. Nevalainen, J. Peña Queralta, J. Heikkonen, and T. Westerlund, "Localization in unstructured environments: Towards autonomous robots in forests with delaunay triangulation", *Remote Sensing*, vol. 12, no. 11, 2020, ISSN: 2072-4292. DOI: 10.3390/rs12111870. [Online]. Available: https://www.mdpi.com/2072-4292/12/11/1870.