SLAM Pipeline Benchmarking
Team 13

Meemansa Pandey
2022102036

Sidhi Sankalp Panda
2021111024

Harshavardhan P
2021111003

I. Problem Statement

The performance of Simultaneous Localization and Mapping (SLAM) systems can vary significantly based on factors such as the type of sensors used, the environment in which the system operates, and the specific SLAM pipeline implemented. With a growing range of SLAM solutions available, choosing the right pipeline for a given application remains a challenging task. Current SLAM frameworks often rely on visual inputs, but the introduction of additional sensors like Inertial Measurement Units (IMUs) and Wheel Odometry has opened up new possibilities for improving the accuracy and robustness of localization.

However, despite their potential, not all SLAM solutions can effectively integrate such sensor data or maintain consistent performance in dynamic or resource-constrained environments. Furthermore, the accuracy of SLAM localization directly impacts subsequent processes such as 3D reconstruction, which relies on precise position tracking for building accurate environmental models.

This project addresses the need for a comprehensive performance evaluation of three prominent SLAM pipelines—RTAB-Map, ORB-SLAM3, and SVO (Semi-Direct Visual Odometry). The core problem is to benchmark these systems under both simulated and real-world conditions, assessing their strengths, weaknesses, and performance limitations in various environments and sensor configurations. The specific challenges addressed include:

Sensor Integration: How well do these SLAM systems utilize different sensor modalities, such as monocular, RGBD, stereo, and IMU, to enhance localization accuracy? Localization Performance: What is the impact of using monocular versus RGBD configurations on localization accuracy and real-time performance? 3D Reconstruction: How do SLAM pipelines affect the quality of dense 3D reconstructions, and to what extent do localization errors propagate into the reconstruction process? Real-World Data Challenges: How do these SLAM systems perform when processing real-world datasets collected from mobile platforms, considering the presence of noise, dynamic elements, and sensor imperfections? By addressing these challenges, this project aims to provide a clearer understanding of the operational trade-offs between different SLAM solutions, guiding the selection of the most appropriate system for specific applications in robotics.

II. Introduction

Simultaneous Localization and Mapping (SLAM) has become fundamental to autonomous robotics, with applications ranging from indoor navigation to autonomous driving. While numerous SLAM approaches exist, their performance can vary significantly based on environmental conditions, sensor configurations, and implementation specifics. This variability presents a significant challenge in selecting and optimizing SLAM systems for specific applications.

Traditional SLAM evaluation approaches typically rely on either real-world datasets, which offer authentic conditions but limited control over variables, or synthetic datasets that may not adequately represent real-world challenges. This dichotomy creates a significant gap in SLAM system evaluation, where researchers and developers struggle to conduct controlled, reproducible experiments while maintaining real-world relevance.

The challenge is particularly evident when comparing fundamentally different SLAM approaches, such as direct methods like Semi-Direct Visual Odometry (SVO) versus feature-based approaches like ORB-SLAM3 and graph-based systems like RTAB-Map. Each system responds differently to environmental variations in lighting, texture, motion patterns, and sensor noise, making fair comparison particularly challenging using existing datasets.

This paper presents a novel SLAM evaluation framework that bridges the gap between synthetic and real-world testing. Our key contribution is a comprehensive platform that leverages Blender's 3D animation capabilities to generate controlled, reproducible datasets with precise ground truth data. The platform enables:

- Generation of synchronized RGB, depth, and IMU data from customizable Blender scenes
- Automated conversion of Blender animations to ROS bag files for seamless integration with SLAM pipelines
- Support for both monocular and stereo camera configurations with controllable depth generation
- Fine-grained control over environmental variables such as lighting, camera motion, and scene complexity
- Direct comparison capabilities between perfect (synthetic) and real-world sensor configurations

We demonstrate our framework's utility by conducting a comprehensive evaluation of three distinct SLAM approaches: SVO, ORB-SLAM3, and RTAB-Map. Through systematic

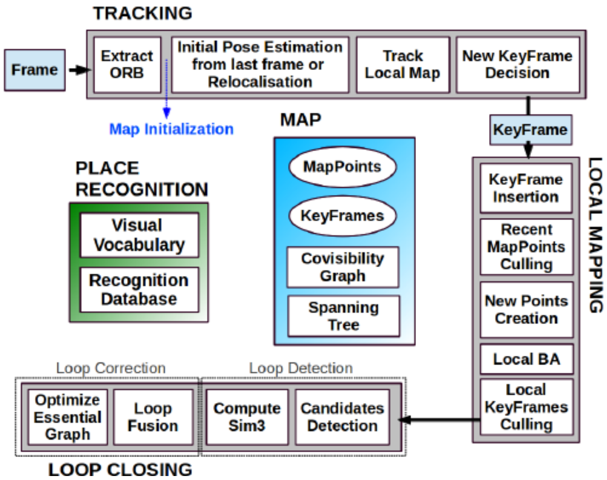

Fig. 1. ORBSlam Architecture.

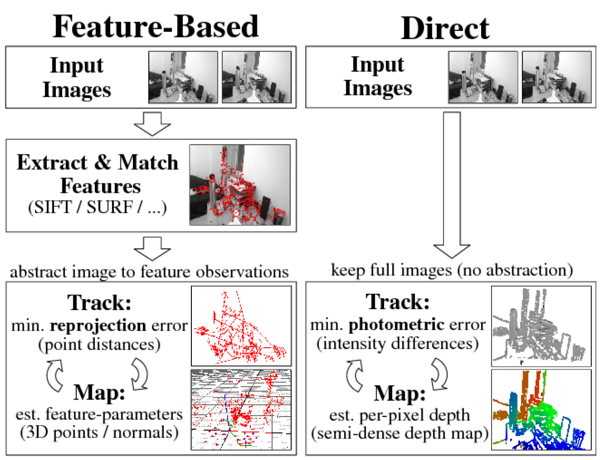

Fig. 2. Feature Based and Direct SLAM.

variation of environmental parameters, we analyze how each system responds to specific challenges, providing insights into their operational characteristics and limitations.

III. Related Work

A. SLAM Frameworks and Approaches

The evolution of Simultaneous Localization and Mapping (SLAM) has witnessed significant paradigm shifts, transitioning from traditional filter-based approaches to modern feature-based and direct methods. Within this evolution, the Robot Operating System (ROS) has established itself as the de facto standard for SLAM implementation and deployment. ROS provides a flexible middleware architecture that facilitates sensor integration and algorithm deployment, enabling researchers and developers to focus on algorithm development rather than system integration challenges.

B. Direct Visual Methods

SVO (Semi-Direct Visual Odometry) represents a fundamental departure from conventional feature-based approaches in visual SLAM. Unlike traditional methods that rely on

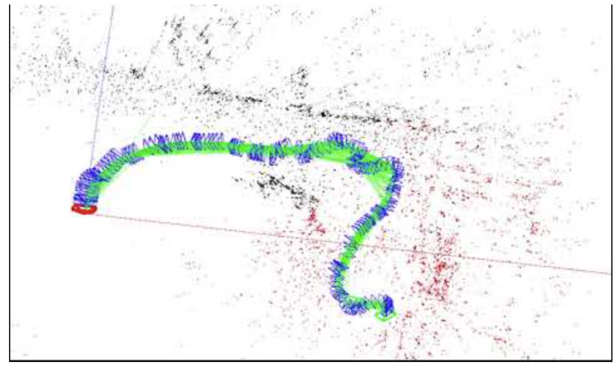

Fig. 3. ORBSlam3 on RRC dataset.

explicit feature extraction and matching, SVO operates directly on image intensities, minimizing photometric error between frames. This direct approach allows the system to leverage semi-dense information from the images while maintaining computational efficiency.

The algorithm's core operation involves sparse image alignment between frames using direct methods, followed by a joint optimization that considers both photometric and geometric errors. A probabilistic depth filter is employed to estimate semi-dense depth maps, enabling robust tracking while maintaining computational efficiency. This innovative approach enables SVO to achieve remarkable frame rates, reaching up to 100 Hz on embedded platforms, making it particularly valuable for resource-constrained applications such as micro aerial vehicles.

However, SVO's reliance on photometric consistency between frames introduces specific vulnerabilities. The system's performance can significantly degrade in environments with poor texture or during rapid illumination changes, as these conditions violate the underlying assumptions of photometric consistency.

C. Feature-Based Methods

The evolution of ORB-SLAM has culminated in ORB-SLAM3, representing the current state-of-the-art in feature-based SLAM systems. This system builds upon its predecessors with significant architectural improvements and innovative features. At its core, ORB-SLAM3 employs a sophisticated multi-map atlas architecture that enables robust operation across disconnected environments, a significant advancement over traditional single-map approaches.

The system's feature extraction and matching capabilities leverage the ORB (Oriented FAST and Rotated BRIEF) descriptor, combined with a bag-of-words approach for efficient place recognition. This foundation is enhanced by several key innovations:

- Multi-map atlas architecture with seamless switching between environments
- Tightly-integrated visual-inertial odometry with online calibration

- Advanced loop closure detection with cross-map place recognition

ORB-SLAM3's visual-inertial integration enables scale-aware operation in monocular configurations and significantly improves robustness during rapid motions or temporary visual tracking failures. The system demonstrates remarkable versatility, effectively handling monocular, stereo, RGB-D, and visual-inertial configurations within a unified framework.

D. Graph-Based Methods with Multi-Sensor Fusion

RTAB-Map (Real-Time Appearance-Based Mapping) represents a comprehensive approach to SLAM that emphasizes robust multi-sensor fusion within a graph-based framework. The system's architecture is designed to handle large-scale environments effectively through sophisticated memory management and multi-session mapping capabilities.

At its core, RTAB-Map employs a hierarchical memory model that intelligently manages computational resources by transferring nodes between working and long-term memory based on their operational relevance. This approach enables consistent performance even in extensive environments where memory constraints might otherwise become problematic.

The system's multi-sensor fusion capabilities are particularly noteworthy, incorporating visual, inertial, and laser data within a unified framework. Real-time loop closure detection is achieved through an efficient bag-of-words approach, enabling robust place recognition and map correction. The framework provides:

- Advanced memory management for scalable mapping
- Sophisticated sensor fusion across multiple modalities
- Real-time performance optimization

E. Synthetic Data Generation and Evaluation Frameworks

The importance of controlled testing environments in SLAM evaluation has led to significant developments in synthetic data generation. Notable contributions include Virtual KITTI and KITTI-SF datasets for autonomous driving scenarios, ICL-NUIM for indoor RGB-D SLAM evaluation, and platforms like UnrealCV and AirSim for synthetic data generation. However, these solutions often present limitations in terms of environmental parameter control or require extensive expertise in computer graphics and simulation.

Evaluation frameworks for SLAM systems have evolved to address specific performance aspects. SLAMBench provides comprehensive performance characterization, while the TUM RGB-D benchmark tools and EuRoC MAV datasets offer standardized evaluation methodologies. These frameworks typically focus on specific performance metrics such as trajectory accuracy or computational efficiency, though they may lack tools for comprehensive environmental variable control and sensor configuration testing.

F. ROS Integration Challenges

Despite its powerful capabilities, the ROS ecosystem presents significant challenges for SLAM implementation and testing. Version compatibility issues between ROS1 and ROS2 create substantial integration complexities, while complex dependencies and system requirements often lead to deployment difficulties. Limited documentation for advanced use cases further complicates development efforts, particularly in cross-platform deployment and containerization.

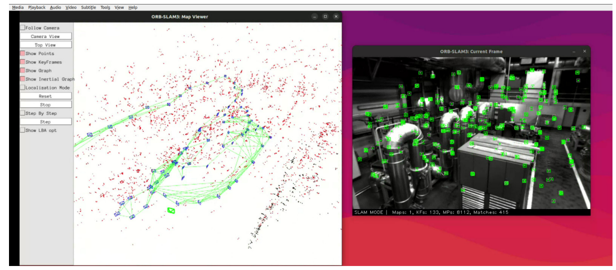

Fig. 4. ORBSlam3 on EuRoC dataset.

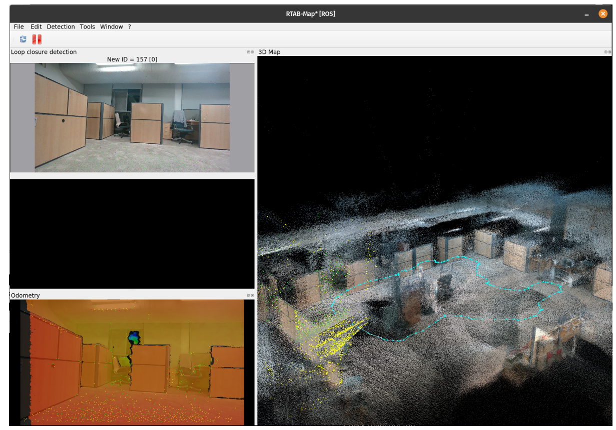

Fig. 5. RTabMap on the RRC dataset.

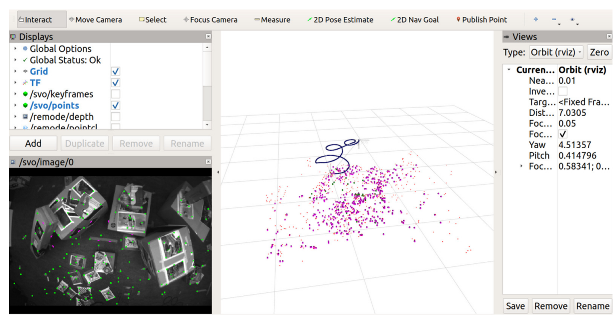

Fig. 6. SVO Localisation.

Recent developments have focused on addressing these challenges through improved containerization strategies and enhanced build systems. However, significant barriers remain in achieving comprehensive SLAM testing and deployment, particularly in maintaining consistency across different platform configurations and managing complex sensor integrations.

IV. Methodology and Implementation

All the codes, documentation and other resources part of the project can be found here here in our github.

A. System Architecture

Our implementation addresses the significant challenge of maintaining compatibility across different SLAM systems while ensuring reproducible testing environments. The strict version dependencies of ROS1 and various SLAM implementations necessitated the development of a containerized architecture using Docker to manage these constraints effectively.

The system architecture employs a multi-container approach, comprising three main containerized environments: a primary ROS1 container for RTAB-Map implementation, a specialized container with pre-configured ORB-SLAM3, and a dedicated container for SVO with its dependencies. This approach was driven by the complexity of building these SLAM systems from source, particularly due to specific version dependencies and compilation challenges that made traditional unified deployment impractical.

The multi-container architecture required sophisticated configuration for seamless operation. A critical aspect was establishing reliable inter-container communication while maintaining system security and performance. We implemented comprehensive X11 forwarding configurations for visualization tools like RViz, ensuring that users could effectively monitor and debug SLAM operations across containers. Port forwarding mechanisms were carefully designed to enable efficient ROS topic communication between containers, while resource management systems were implemented to optimize container performance and prevent resource conflicts.

One of the key technical achievements was developing a unified approach to dataset handling across different SLAM systems. This involved creating standardized interfaces for various data formats and implementing efficient data sharing mechanisms between containers, ensuring consistent access to both real-world and synthetic datasets across all SLAM implementations.

B. Dataset Selection and Processing

Our evaluation framework incorporates a diverse selection of both real-world and synthetic datasets, carefully chosen to provide comprehensive testing scenarios. The dataset selection process was guided by the specific requirements and capabilities of each SLAM system, ensuring meaningful evaluation across different operational contexts.

For RTAB-Map evaluation, we utilized the RRC datasets, which feature rich RGBD and IMU data collected using RealSense cameras mounted on P3DX robots. These datasets provide excellent test cases for indoor navigation and mapping scenarios. ORB-SLAM3 testing employed the EuRoC MAV dataset, offering challenging stereo and IMU data sequences that effectively test the system's robustness in dynamic environments. SVO evaluation utilized a specialized dataset provided with the implementation, optimized for direct methods and offering specific challenges for photometric tracking.

A significant advantage of our framework is the standardization of all datasets to the ROS bag format. This standardization simplifies integration with our synthetic data generation pipeline and ensures consistent handling of temporal synchronization and sensor calibration across different SLAM systems.

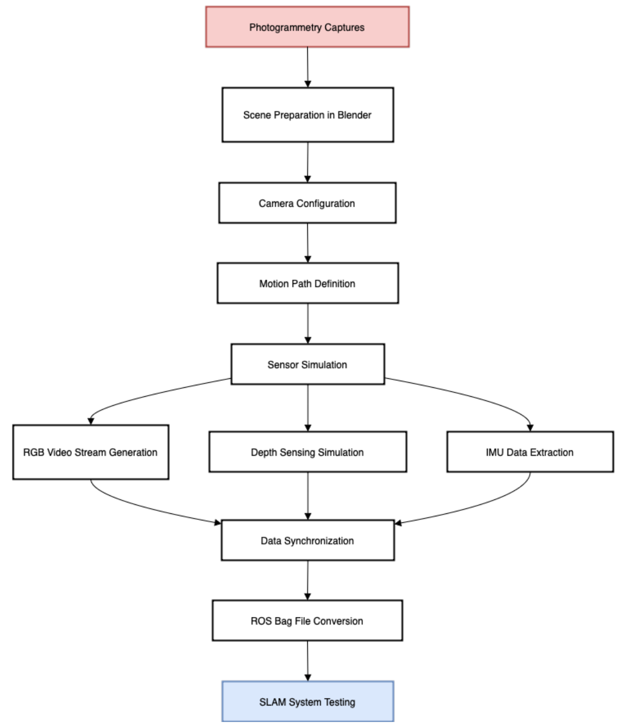

Fig. 7. Overview of the blender dataset generation.

C. Synthetic Dataset Generation Framework

We developed a novel synthetic dataset generation pipeline utilizing Blender, creating a powerful tool for generating controlled, reproducible testing environments. The framework's architecture is designed to provide precise control over all aspects of data generation while maintaining realistic environmental conditions.

Scene preparation forms the foundation of our synthetic data generation process. We utilize high-quality photogrammetry captures from open-source repositories, carefully processing and scaling them to match real-world dimensions. This attention to physical accuracy extends to the implementation of realistic motion properties, ensuring that generated datasets reflect authentic robotic movement patterns.

The camera and sensor simulation system represents a significant technical achievement. Virtual cameras are configured to match real-world sensor parameters precisely, supporting both monocular and stereo configurations. Depth sensing is implemented using Blender's mist layer with configurable near and far planes, providing accurate depth information that closely mimics real sensor behavior. A custom scripting system extracts IMU data from camera motion, ensuring physically accurate sensor readings.

Data generation and synchronization presented unique challenges that we addressed through innovative solutions. The framework generates perfectly synchronized RGB and depth video streams, with custom implementations for stereo depth computation in multi-camera configurations. Camera intrinsics and motion parameters are extracted with high precision, while Python scripts utilizing the rosbags library ensure proper data integration and timestamp management.

Environmental control capabilities represent a key advantage of our framework. Users can modify lighting conditions, apply motion blur effects, introduce sensor noise, and adjust camera movement patterns with precise control. This fine-grained manipulation of environmental parameters enables systematic evaluation of SLAM performance under varying conditions, providing insights that would be difficult to obtain with real-world data alone.

The resulting framework offers several distinct advantages for SLAM evaluation:

- Complete control over environmental variables with precise parameter adjustment
- Perfect ground truth data availability for accurate performance assessment
- Reproducible testing scenarios enabling systematic comparison
- Isolation and evaluation of individual factors affecting SLAM performance

Through the integration of these components, we have created a comprehensive testing environment that enables systematic evaluation of SLAM systems under controlled conditions while maintaining the capability to reproduce real-world challenges. This framework represents a significant step forward in SLAM system evaluation methodology, providing tools for both detailed analysis and comparative benchmarking.

V. Experimental Results

A. Preliminary SLAM Evaluation

Our evaluation of the three SLAM systems revealed distinct characteristics and operational constraints across different testing scenarios. Initial testing with RTAB-Map using the RRC datasets demonstrated robust performance in indoor environments with RGBD and IMU data. The system effectively maintained localization across varying lighting conditions and showed reliable loop closure detection in corridor environments. However, the computational overhead of processing dense RGBD data resulted in notable latency on resource-constrained systems.

ORB-SLAM3, tested with the EuRoC MAV dataset, exhibited impressive tracking stability and accurate trajectory estimation in feature-rich environments. The system's multi-threading architecture efficiently handled parallel tracking and mapping operations, though we observed occasional tracking losses in regions with repetitive patterns or limited visual features. The visual-inertial integration proved particularly effective in maintaining scale consistency and improving robustness during rapid camera motions.

SVO testing, while limited to the provided dataset, demonstrated the system's capability for high-frame-rate operation and efficient direct tracking. The semi-direct approach showed particular strength in scenarios with good illumination and clear intensity gradients. However, the system's performance was notably sensitive to motion blur and rapid illumination changes, reflecting the challenges inherent to direct methods.

B. Technical Challenges and Solutions

The implementation and testing process revealed several significant technical challenges that required innovative solutions. Docker containerization, while solving compatibility issues, introduced complexity in system integration and resource management. The primary challenges centered around three key areas:

Container Communication and Visualization: Establishing reliable communication between dockerized SLAM systems required careful configuration of network interfaces and port mappings. We developed a systematic approach to X11 forwarding that enabled consistent access to visualization tools across containers while maintaining system security. This involved creating dedicated scripts for managing display server access and container initialization.

Dataset Integration and Synchronization: Each SLAM system exhibited unique requirements for data preprocessing and format conversion. Real-world datasets often contained varying sensor configurations and timing patterns, necessitating the development of flexible data preprocessing pipelines. We implemented custom synchronization modules to ensure consistent temporal alignment of sensor data across different testing scenarios.

Resource Management and Performance Optimization: Running multiple containerized SLAM systems simultaneously posed significant challenges in terms of resource allocation and system performance. We addressed this through careful container resource limiting and implementation of monitoring tools to track system utilization. This was particularly critical when processing dense RGBD data or running computationally intensive optimization routines.

The development of our synthetic dataset generation pipeline also presented unique challenges. While Blender provided powerful tools for scene creation and animation, extracting accurate sensor data required deep understanding of both computer graphics principles and robotics sensor models. The integration of this data into ROS-compatible formats demanded careful attention to timestamp synchronization and coordinate frame transformations.

Camera parameter extraction and IMU data generation proved particularly challenging, requiring custom scripts to bridge the gap between Blender's animation system and real-world sensor characteristics. We developed specialized tools to extract and convert camera motion data into realistic IMU measurements, including proper noise modeling and bias characteristics.

These challenges and their solutions have contributed significantly to our understanding of both the technical requirements for comprehensive SLAM evaluation and the practical considerations in implementing such testing frameworks. The experience gained has informed our ongoing development of more robust and automated testing procedures.

VI. Analysis and Discussion

A. SLAM Performance Analysis

The comparative analysis of SVO, ORB-SLAM3, and RTAB-Map revealed distinct operational characteristics that highlight the trade-offs inherent in different SLAM approaches. Through our testing framework, we observed that environmental factors significantly influence each system's performance in unique ways.

ORB-SLAM3's feature-based approach demonstrated robust performance across varied environments, particularly excelling in feature-rich scenes with good illumination. The system's visual-inertial integration proved especially valuable in maintaining consistency during rapid motions, a scenario where purely visual systems often struggle. However, the system's reliance on distinctive visual features makes it vulnerable in environments with repetitive patterns or minimal texture, even with IMU integration.

The direct method employed by SVO showed remarkable efficiency in computational resource utilization, achieving higher frame rates compared to feature-based approaches. This efficiency comes with trade-offs in robustness, particularly in scenarios with significant illumination changes or motion blur. Our testing revealed that SVO's performance is highly dependent on the quality of the initial pose estimation and the presence of strong intensity gradients in the scene.

RTAB-Map's graph-based approach with multi-sensor fusion demonstrated superior robustness in long-term operation scenarios. The system's memory management capabilities effectively handled large-scale environments, though at the cost of increased computational overhead. The integration of RGB-D data provided reliable depth estimation, making it particularly suitable for indoor navigation tasks where accurate depth perception is crucial.

B. Framework Analysis

Our containerized testing environment proved effective in managing the complexity of running multiple SLAM systems with different dependencies. The Docker-based approach successfully isolated system-specific requirements while maintaining the flexibility needed for comprehensive testing. However, this architecture introduced overhead in terms of system resource utilization and increased complexity in data sharing between containers.

The novel Blender-based dataset generation framework demonstrated significant potential for controlled SLAM evaluation. The ability to precisely control environmental parameters while maintaining ground truth data provides valuable insights into system behaviours under specific conditions. This capability is particularly valuable for isolating the effects of individual factors such as lighting, motion patterns, and sensor noise.

Several key findings emerged from our framework implementation:

The ability to generate synchronized sensor data streams with perfect ground truth enables a more precise evaluation of SLAM performance than typically possible with real-world datasets. This is particularly valuable for understanding system behaviour under specific challenging conditions that may be difficult to reproduce consistently in real-world settings.

However, our framework also revealed limitations that warrant further development. The current implementation requires significant manual intervention in scene preparation and camera path definition. While this allows for precise control over testing scenarios, it limits the scalability of our testing approach. Additionally, the computational overhead of rendering high-quality synthetic data can be substantial, particularly for complex scenes or long sequences.

In considering the trade-offs between synthetic and real-world testing, we found that synthetic data provides invaluable insights for initial system validation and parameter tuning. However, real-world testing remains essential for understanding system performance under genuine environmental conditions. The complementary nature of these approaches suggests that a hybrid testing methodology, combining both synthetic and real-world data, may provide the most comprehensive evaluation framework.

Performance characteristics observed across different datasets highlighted the importance of environmental diversity in SLAM evaluation. While standard benchmark datasets provide valuable reference points, our synthetic data generation capability enables exploration of edge cases and specific challenging scenarios that may be underrepresented in existing datasets.