Metric-Semantic SLAM, Trajectory Evaluation, and Object-Level Segmentation on the TCS-X Ground Floor Using Intel RealSense

Falak Fatima(24349) and Ganga Nair b(25565)
Course: Robot Perception

May 5, 2025

1 Introduction

Recent advancements in robotics emphasize the significance of Simultaneous Localization and Mapping (SLAM) systems, particularly those integrating metric and semantic capabilities. Metric-semantic SLAM enables robots not only to build geometrically accurate representations of their environment but also to assign semantic labels to structures and objects. This combined geometric and semantic understanding is essential for tasks requiring intelligent robot interaction, such as precise navigation, effective object localization, and comprehensive scene interpretation in indoor environments like offices and industrial complexes.

In this project, we perform metric-semantic SLAM to reconstruct a detailed map of a specific portion of the ground floor of the TCS-X building using an Intel RealSense depth camera. We specifically leverage Kimera, a robust SLAM library integrating Visual-Inertial Odometry (VIO), Robust Pose Graph Optimization (RPGO), and dense semantic mapping capabilities. Our objectives include: generating accurate maps; quantitatively evaluating map quality using Absolute Trajectory Error (ATE); and accurately localizing a predefined piece of furniture designated couch or person within the mapped scene.

Additionally, we undertake an extensive comparative analysis between Kimera and RTAB-Map, two prominent metric-semantic SLAM frameworks, to highlight their respective strengths, computational efficiencies, and robustness when applied to indoor mapping tasks. Insights from this analysis provide valuable guidance on the practical deployment and potential enhancements of robotic perception systems in complex indoor scenarios.

2 Map Demonstration

2.1 SLAM Using Kimera

Figures 1 and 2 demonstrate the metric-semantic reconstruction of the TCS-X ground floor using the Kimera SLAM framework. The reconstructions illustrate Kimera's capabilities in creating dense, accurate, and semantically meaningful maps using stereo camera images and inertial data from the Intel RealSense depth camera. As observed, the map captures intricate geometric details such as walls, pillars, and furniture clearly. The spatial layout, including both structural features and movable objects, is accurately represented, demonstrating Kimera's robust visual-inertial odometry and mesh reconstruction capabilities as highlighted in the literature.

The visualizations demonstrate Kimera's efficacy in providing both geometric accuracy and semantic clarity, essential for tasks such as navigation, obstacle avoidance, and targeted object localization.

2.2 SLAM Using RTAB-Map

Figures 4 and 5 illustrate the mapping results obtained using the RTAB-Map SLAM approach. RTAB-Map (Real-Time Appearance-Based Mapping) leverages visual odometry and RGB-D sensors to produce a comprehensive 3D reconstruction. It employs loop closure detection to enhance map consistency and accuracy, effectively correcting drift in the trajectory.

The generated 3D maps highlight RTAB-Map's ability to clearly reconstruct detailed spatial structures, capturing significant visual features such as furniture, wall boundaries, and specific indoor landmarks. The loop closure detection, as visually represented in the figures, significantly enhances the

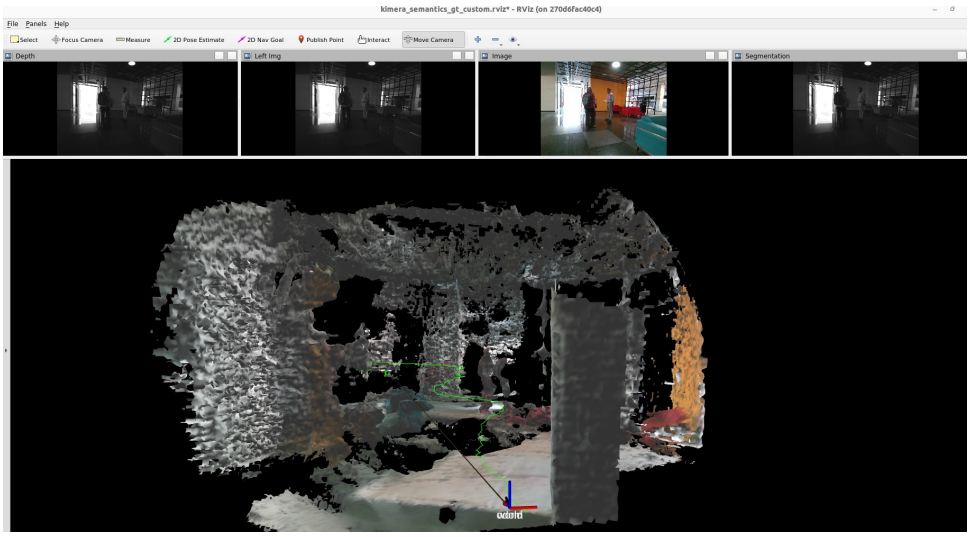


Figure 1: Metric-semantic reconstruction highlighting structural elements and indoor layout(with CPU).

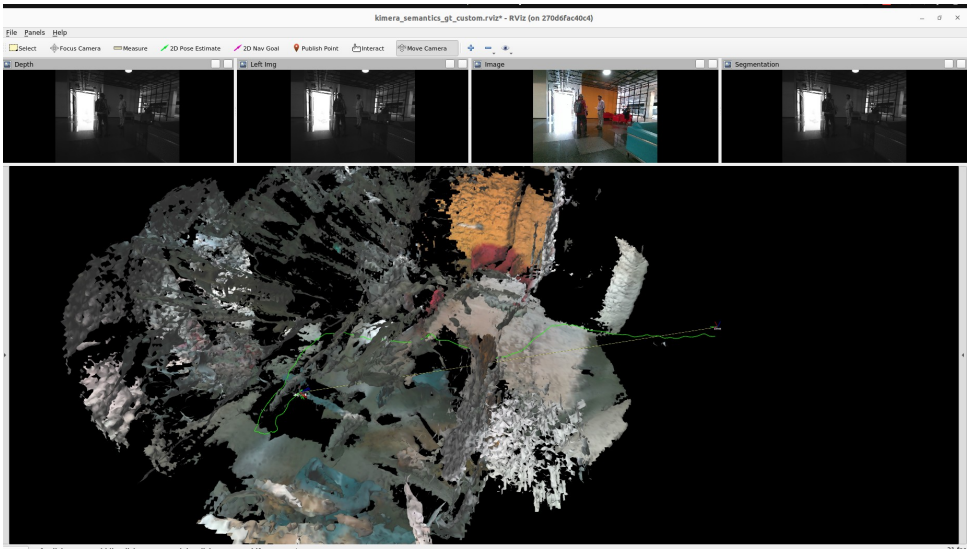


Figure 2: Detailed reconstruction map of indoor environment(With CPU).

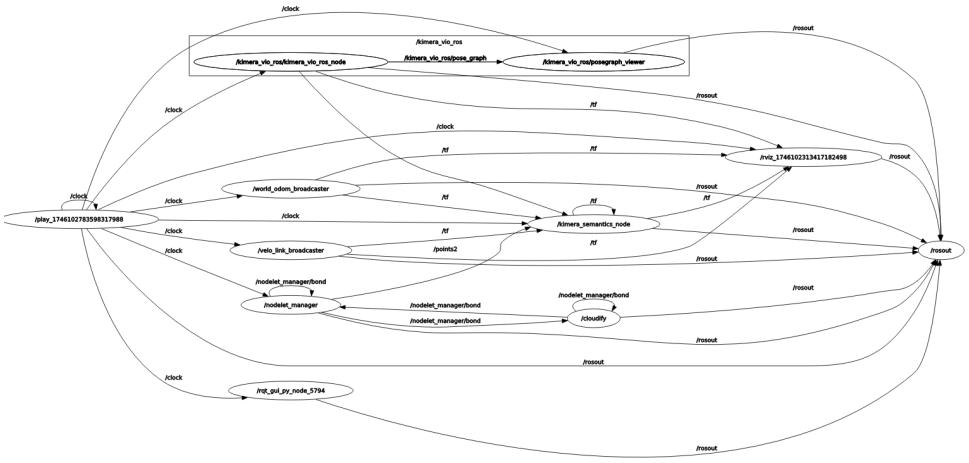


Figure 3: Rqt graph of Kimera SLAM for reconstruction of indoor environment.

robustness and accuracy of the final map by recognizing previously visited locations and correcting the accumulated drift over the robot's path.

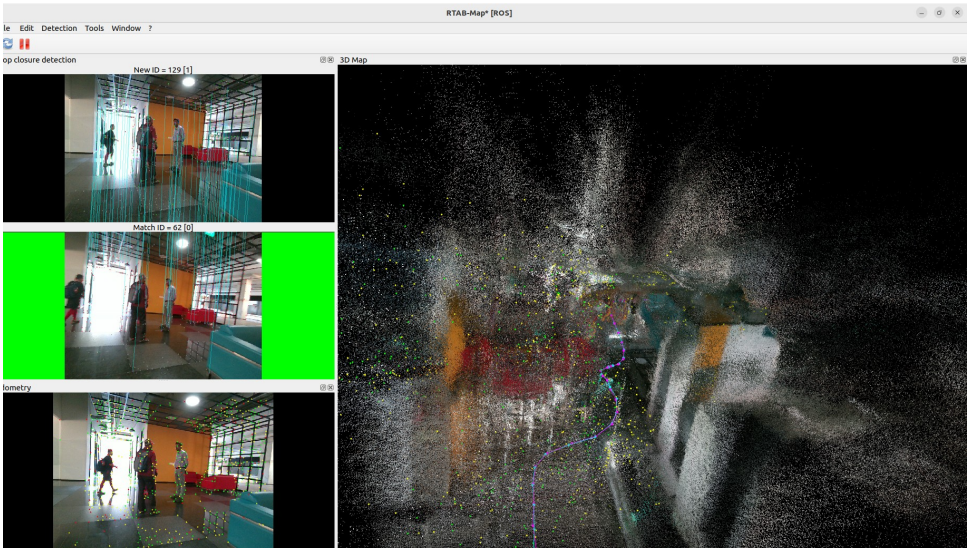


Figure 4: RTAB-Map reconstruction demonstrating detailed spatial features and loop closures.(with CPU)

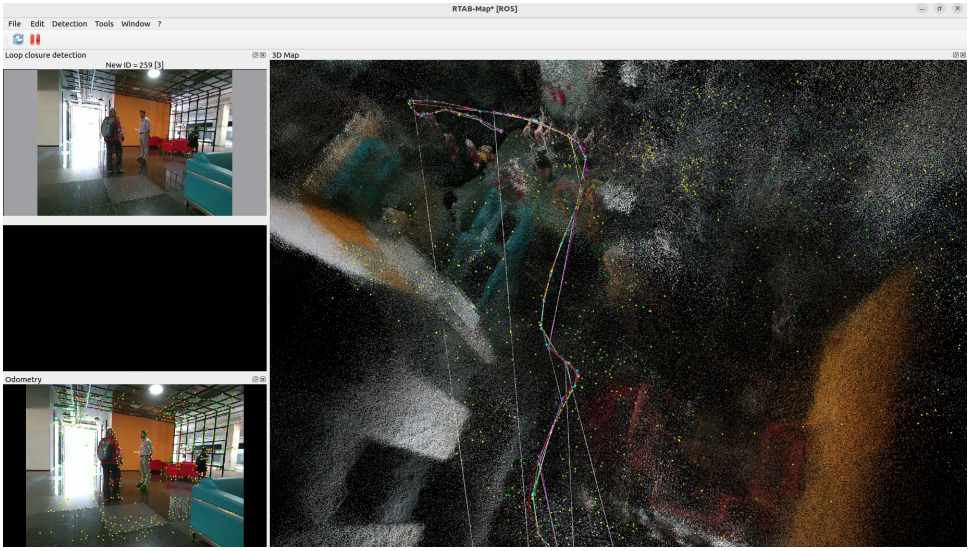


Figure 5: Visualization of robust loop closure detection and comprehensive 3D map using RTAB-Map.(with CPU)

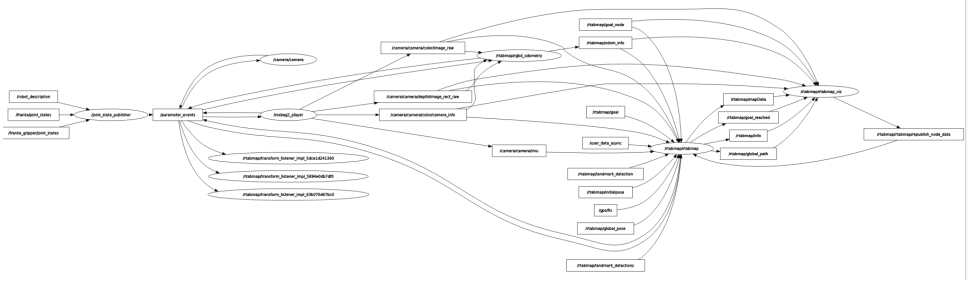
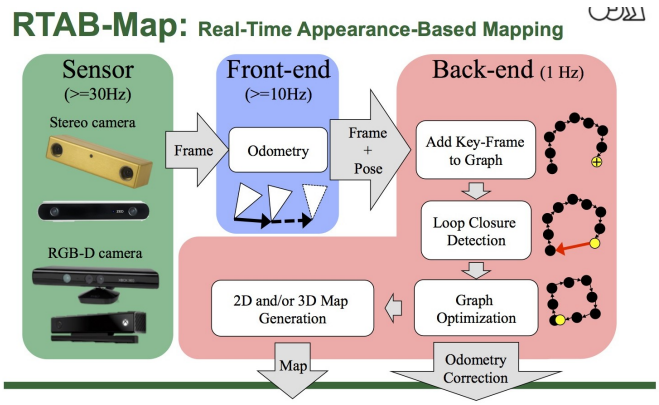


Figure 6: Rqt graph of RTABMAP SLAM for reconstruction of indoor environment.

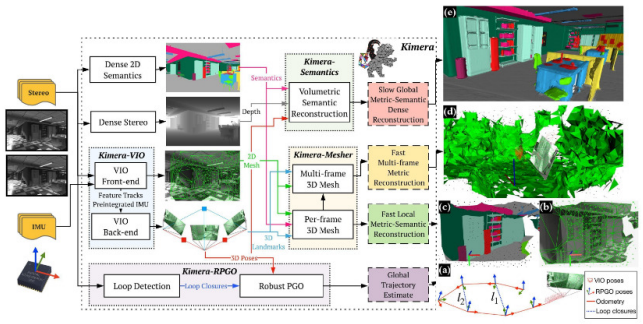
These visualizations underscore RTAB-Map's strength in real-time metric mapping and reliable drift

correction via visual loop closures, making it a robust solution for indoor SLAM applications.

3 Comparison between RTAB-Map and Kimera



(a) RTAB-Map architecture: shows sensor input, odometry, loop closure, and map generation pipeline.



(b) Kimera framework: integrates visual-inertial odometry, semantic mapping, and 3D mesh generation.

Figure 7: Architectural comparison between RTAB-Map and Kimera SLAM frameworks.

Both RTAB-Map and Kimera are widely adopted SLAM frameworks with distinct architectures and strengths, tailored for different use-cases in robotics perception. In this project, we evaluated both approaches on the same portion of the TCS-X ground floor using data collected from an Intel RealSense camera.

RTAB-Map, short for Real-Time Appearance-Based Mapping, is a visual SLAM system based on loop closure detection and memory management. It creates dense point clouds and integrates RGB-D data efficiently in real-time. It is particularly effective in maintaining global consistency through robust loop closure detection mechanisms.

Kimera, in contrast, is a modular, research-oriented SLAM library that fuses visual-inertial odometry, mesh reconstruction, and semantic labeling. It excels in producing metrically and semantically rich reconstructions using minimal drift, thanks to its back-end optimization via RPGO (Robust Pose Graph Optimization).

While RTAB-Map is easier to set up and provides visually rich results using RGB-D sensors alone, Kimera delivers more refined and semantically-aware mesh maps, crucial for applications requiring environmental understanding.

4 Absolute Trajectory Error (ATE)

The Absolute Trajectory Error (ATE) is a widely used metric to evaluate the global consistency of a SLAM system. It quantifies the difference between the estimated trajectory produced by the SLAM algorithm and the ground truth trajectory, typically provided by an external tracking system or a pre-measured reference.

Feature	RTAB-Map	Kimera
Sensors	RGB-D cameras, Stereo vision	Monocular/Stereo cameras with IMU
Backend	Graph-based optimization with loop closure detection	Robust Pose Graph Optimization (RPGO) with loop closure detection
Mapping Output	Dense 3D point cloud maps	Dense semantic 3D mesh
Computational Efficiency	Real-time operation, moderate computational load	Real-time with high computational efficiency, optimized for CPU use
Semantic Capability	Limited semantic annotation, primarily geometric	Extensive semantic annotation integrated.0 with 3D mesh
Robustness	Reliable drift correction via visual loop closure	Highly robust to perceptual aliasing, robust outlier rejection

Table 1: Comparative analysis of RTAB-Map and Kimera SLAM frameworks

Procedure: A trajectory with predefined stopping points was designated as the ground truth and marked in advance (as shown in Figure 8). A ROS bag was then recorded while traversing this trajectory. At each designated stopping point, a timestamp was manually recorded via the command line to create a ground truth file. Manual measurements of the trajectory was added to this later. The first data point was considered to be equal to the kimera trajectory value. After processing the rosbag using Kimera-Semantics, we subscribed to the `/optimized_trajectory` topic to extract the estimated trajectory. From this data, we extracted the timestamp along with the x, y, and z position values. The final step involved computing the difference between the estimated and ground truth trajectories.

Table 2: Ground Truth Trajectory Points

Label	Time	X	Y	Z
Point A	1745938804	0.0	1.48	0.0
Point B	1745938818	0.0	3.38	0.0
Point C	1745938831	0.0	5.28	0.0
Point D	1745938840	0.0	7.18	0.0
Point E	1745938849	0.0	9.08	0.0

In the figure 10 the red trajectory corresponds to the path estimated by RTAB-Map using RGB-D inputs. The green trajectory represents the ground truth, which was measured manually separately under controlled conditions. Ideally, both trajectories should align perfectly. However, due to sensor noise, drift in odometry, or imperfect loop closures, the estimated path often deviates from the true path. This deviation is known as drift.

The ATE is computed as the root mean square error (RMSE) of the Euclidean distances between the aligned estimated poses and ground truth poses:

$$ATE_{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \|p_i^{est} - p_i^{gt}\|^2}$$

where p_i^{est} and p_i^{gt} are the estimated and ground truth positions at time i, respectively.

The error plots indicate that the trajectory estimated by the SLAM system begins to diverge from the ground truth over time. This increasing error suggests an accumulation of small drift errors typical in odometry-based systems. Despite this drift, the overall accuracy remains reasonable, as evidenced by the Absolute Trajectory Error (ATE) metrics, which report a low RMSE of 0.2788m. The trajectory visualization also highlights that the system follows the ground truth path closely, with only minor spatial discrepancies. These results demonstrate that while the system performs well in general, long-term drift accumulation remains a challenge in accurate localization.

Figure 8: Depicts Ground truth used for trajectory estimation green color path followed with red colour marking points

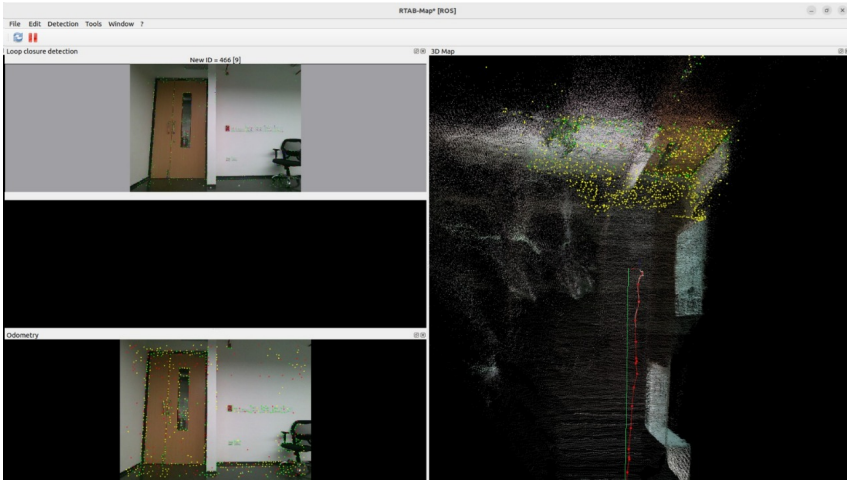


Figure 9: Trajectory visualization from RTAB-Map: the red line indicates the estimated trajectory from SLAM, while the green line represents the ground truth.

Table 3: Absolute Trajectory Error metrics for the RTAB-Map run.

Metric	**Value (m)**
RMSE	0.2788
Mean Error	0.2627
Median Error	0.2342

5 Localization of Specific Couch and Person

Semantic segmentation and localization are key tasks in robotic perception that enable understanding and interaction with the environment. While semantic segmentation classifies each pixel in an image into meaningful categories (e.g., road, pedestrian, building), localization determines the precise position within a map or environment.

We have decided to use RTABMap for achieving loaclisation. Our aim was to implement real time

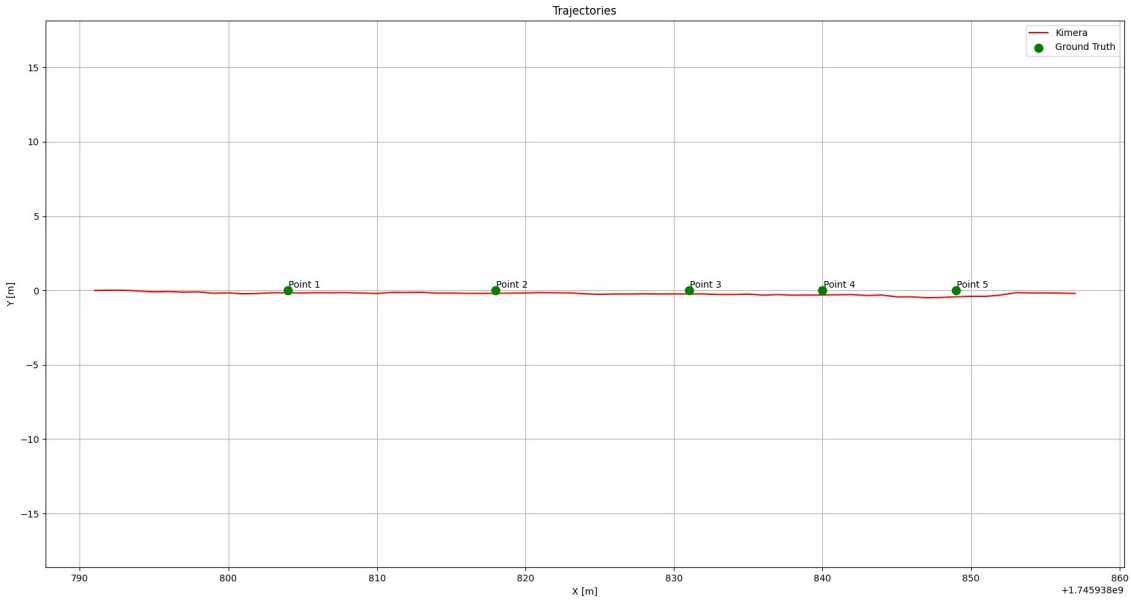
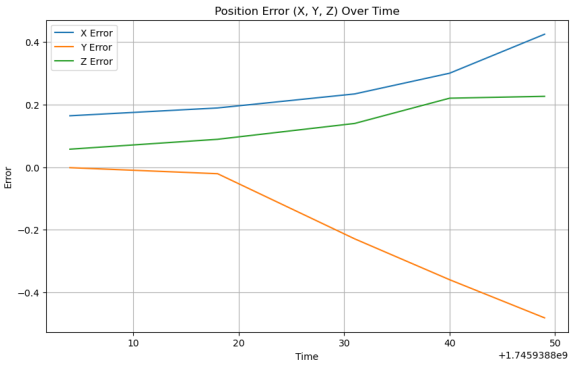
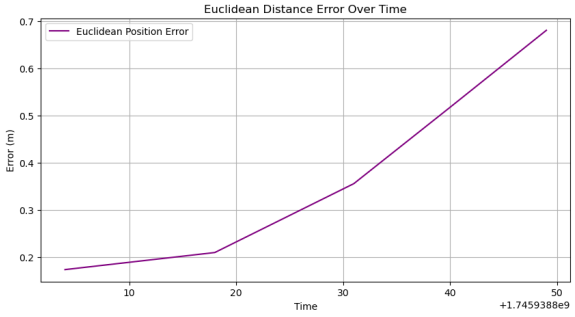


Figure 10: Comparison of Trajectory from SLAM system vs ground truth.



(a) Error along x, y, z coordinates.



(b) Total euclidean distance between ground truth and trajectory at given timestamp.

localisation of a couch (and other indoor objects) as the rosbag was playing. The following were keypoints to achieve localisation:

- Localisation of specific entities such as couches and people was achieved through semantic segmentation integrated with RTAB-Map SLAM as shown in figure 12.
- We employed a pretrained **DeepLabV3 ResNet-50 model** from the TorchVision library.
- The segmentation was applied frame-by-frame on RGB images extracted from the rosbag dataset.

The pipeline for segmentation was as follows:

1. To perform segmentation, each RGB frame from the $/camera/camera/color/image_raw$ topic was passed through a preprocessing pipeline that included resizing, normalization, and conversion to tensor format compatible with the model's input expectations.
2. Inference was then carried out using the DeepLabV3 model, which classifies every pixel in the image into one of the COCO dataset's classes.
3. The resulting segmentation mask was decoded into a colorized overlay, and bounding boxes were drawn for recognized instances of person, sofa, and couch.
4. Semantic labels were rendered using OpenCV to aid in visual interpretation. Objects smaller than a threshold (100 pixels) were filtered out to reduce noise.

5. A custom ROS 2 bag-processing pipeline was implemented using Python, *rosbag2py*, and *cvbridge*, allowing real-time segmentation and generation of semantic overlays.

6. The processed data was published back into a new rosbag topic /*camera*/*camera*/*color*/*semantic*$_l$*abels*, preserving timestamps and enabling synchronized playback with odometry and point cloud data.

This approach enabled visual confirmation of the specific couch and people locations within the mapped scene, aiding downstream robotic decision-making tasks like path planning, obstacle avoidance, or human-aware navigation.

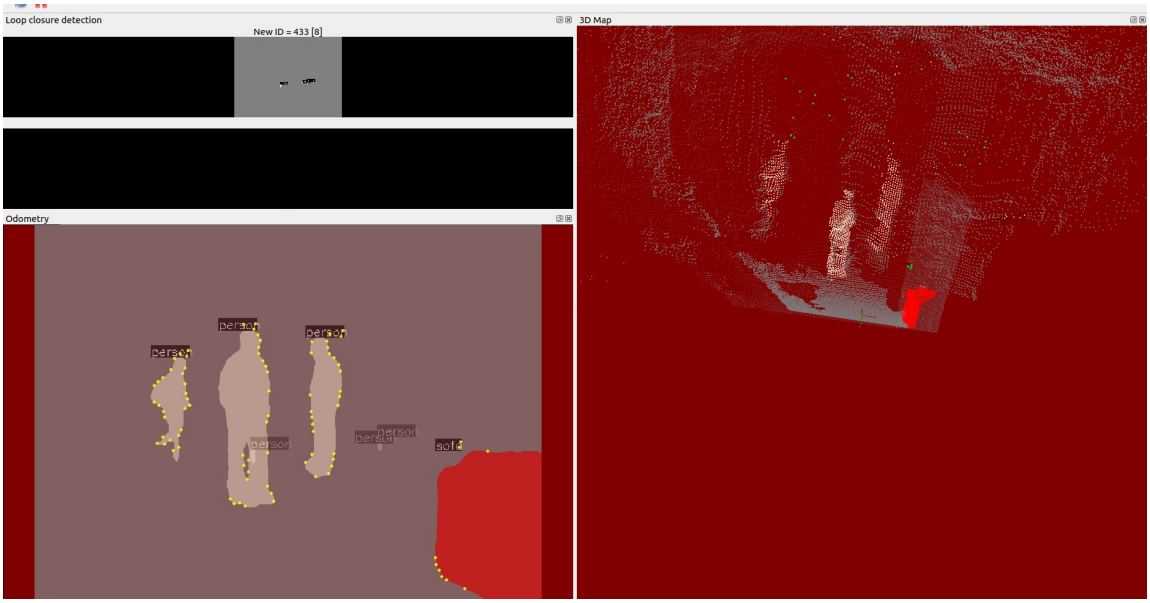


Figure 12: Metric-semantic reconstruction highlighting structural elements and indoor layout.

6 Summary

This report presents a metric–semantic SLAM study on the TCS-X ground floor using an Intel RealSense camera, comparing two leading frameworks—Kimera and RTAB-Map. We first demonstrate Kimera's visual-inertial odometry and dense semantic mesh reconstruction, highlighting its ability to produce accurate, semantically rich maps with minimal drift. Next, we show RTAB-Map's real-time RGB-D mapping, where loop-closure detection yields consistent 3D point clouds. A side-by-side architectural comparison and quantitative evaluation via Absolute Trajectory Error (ATE) reveal that RTAB-Map achieves an RMSE of 0.2788 m (mean 0.2627 m, median 0.2342 m), while Kimera offers tighter semantic integration. Finally, we integrate DeepLabV3-based segmentation to localize specific objects (couch, person) within the reconstructed scene.