Data paper

The International Journal of
Robotics Research
2024, Vol. 43(7) 899–917
© The Author(s) 2023
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/02783649231222507
journals.sagepub.com/home/ijr
Sage

# CID-SIMS: Complex indoor dataset with semantic information and multi-sensor data from a ground wheeled robot viewpoint

Yidi Zhang[1,2,*], Ning An[3,*], Chenhui Shi[2], Shuo Wang[1,2], Hao Wei[2], Pengju Zhang[2], Xinrui Meng[3], Zengpeng Sun[3], Jinke Wang[3], Wenliang Liang[3], Fulin Tang[2] and Yihong Wu[2,1]

**Abstract**

*Simultaneous localization and mapping (SLAM) and 3D reconstruction have numerous applications for indoor ground wheeled robots such as floor sweeping and food delivery. To advance research in leveraging semantic information and multi-sensor data to enhance the performances of SLAM and 3D reconstruction in complex indoor scenes, we propose a novel and complex indoor dataset named CID-SIMS, where semantic annotated RGBD images, inertial measurement unit (IMU) measurements, and wheel odometer data are provided from a ground wheeled robot viewpoint. The dataset consists of 22 challenging sequences captured in nine different scenes including office building and apartment environments. Notably, our dataset achieves two significant breakthroughs. Firstly, semantic information and multi-sensor data are provided meanwhile for the first time. Secondly, GeoSLAM is utilized for the first time to generate ground truth trajectories and 3D point clouds within two-centimeter accuracy. With spatial-temporal synchronous ground truth trajectories and 3D point clouds, our dataset is capable of evaluating SLAM and 3D reconstruction algorithms in a unified global coordinate system. We evaluate state-of-the-art SLAM and 3D reconstruction approaches on our dataset, demonstrating that our benchmark is applicable. The dataset is publicly available on https://cid-sims.github.io.*

**Keywords**

Dataset, ground wheeled robots, semantic segmentation, multi-sensor data, simultaneous localization and mapping, 3D reconstruction

Received 04 July 2023; Revised 14 November 2023; Accepted 01 December 2023
Senior Editor: Tim Barfoot
Associate Editor: Ayoung Kim

## 1. Introduction

Simultaneous localization and mapping (SLAM) and 3D reconstruction are key problems in the field of mobile robotics (Appleton and Williams (2012); Hägele et al. (2016); Oleynikova et al. (2020); Wu et al. (2018)). With the advantages of convenience and low cost, vision-based approaches have received extensive researches (Davison et al. (2007); Forster et al. (2014); Mur-Artal et al. (2015); Schonberger and Frahm (2016); Moulon et al. (2016)). Complementing vision with other information, such as depth (Whelan et al. (2015); Mur-Artal and Tardós (2017); Chen et al. (2022); Li et al. (2022); Shi et al. (2023)), inertial measurement unit (IMU) (Mourikis and Roumeliotis (2007); Qin et al. (2018); Geneva et al. (2020); Campos et al. (2021); Wei et al. (2021a, 2022); Xu et al. (2023)), and prior semantic annotations (Rosinol et al. (2020); Fan et al. (2022); Zimmerman et al. (2022, 2023)), tremendously improves performances in accuracy and robustness, and thus has spawned much interest.

One of the most crucial factors for these advancements and interest can be attributed to the public data suites (Geiger et al. (2013); Burri et al. (2016); Schubert et al. (2018)). In order to fairly compare the performances of algorithms under the same metric, public benchmarks for evaluating SLAM and 3D reconstruction approaches are indispensable. Current

[1] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[2] State Key Laboratory of Multimodal Artificial Intelligence, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[3] Horizon Robotics, Beijing, China

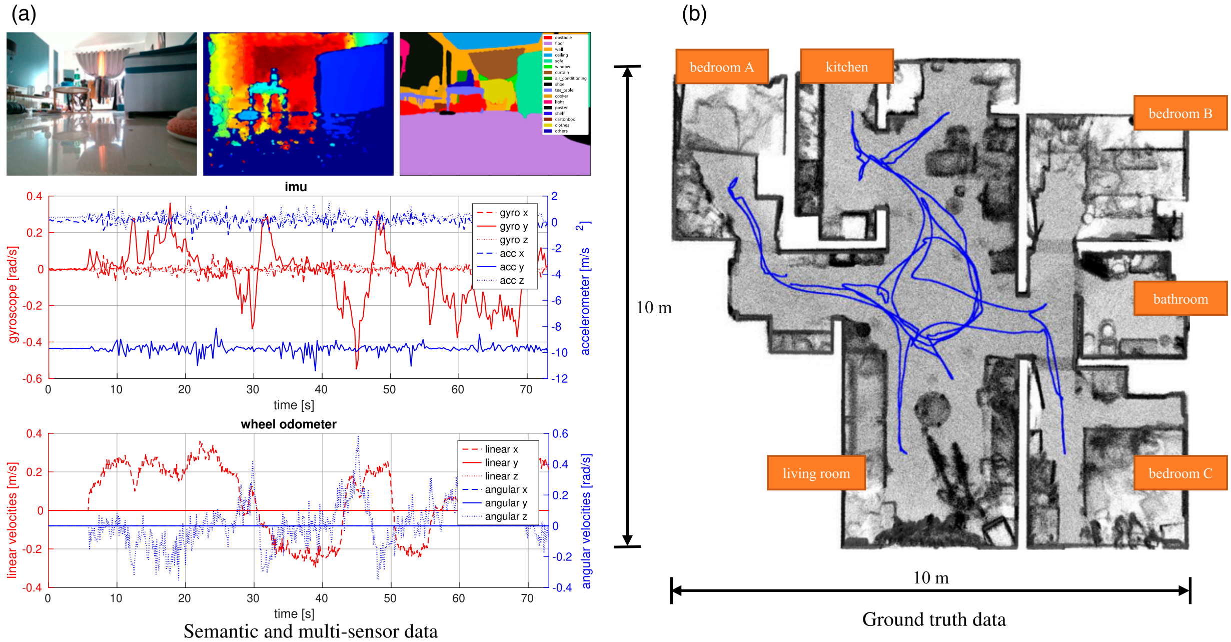
*These authors contributed equally to this work and should be considered co-first authors.

**Corresponding authors:**
Fulin Tang, State Key Laboratory of Multimodal Artificial Intelligence, Institute of Automation, Chinese Academy of Sciences, No. 95 Zhongguancun East Road, Beijing 100190, China.
Email: fulin.tang@nlpr.ia.ac.cn

Yihong Wu, State Key Laboratory of Multimodal Artificial Intelligence, Institute of Automation, Chinese Academy of Sciences, No. 95 Zhongguancun East Road, Beijing 100190, China.
Email: yhwu@nlpr.ia.ac.cn

public datasets for indoor scenes are mainly collected by unmanned aerial vehicles (UAVs) (Burri et al. (2016)) or handheld devices (Handa et al. (2014); Schubert et al. (2018)), which have 6-degree-of-freedom (6-DoF) motions. Existing solutions frequently perform poorly in practice because degenerate 3D motions reduce the observability of SLAM systems, thereby leading to performance degradation (Wu et al. (2017)). Therefore, these 6-DoF datasets are not suitable for evaluating the performances of SLAM and 3D reconstruction algorithms under 3-DoF motions, which are commonplace for ground wheeled robots. Furthermore, many commercial ground wheeled robots[2,3] can only observe environments with a low camera viewpoint. The data from them has more regions on floors and hence has much noise, which increases the challenge for existing algorithms. In this situation, the combination of multi-sensor and multivariate information is important (Zhang et al. (2019); Zhou et al. (2019); Wei et al. (2021b); Liu et al. (2022)). Although wheel odometer data has been shown to be crucial for accurate and robust localization of ground wheeled robots in many publications (Wu et al. (2017); Zhang et al. (2019); Liu et al. (2019); Zhuang (2021)), few indoor datasets provide wheel odometer data. In addition, semantic information is increasingly used to improve the accuracy and robustness of SLAM and 3D reconstruction tasks (Lianos et al. (2018); Rosinol et al. (2020); Fan et al. (2022); Schmid et al. (2022)). However, semantic SLAM and 3D reconstruction researchers struggle with evaluating and comparing their results due to the lack of specific benchmarks with semantic information. They usually need to obtain semantic information through modern deep-learning methods, which greatly increases the workload.

For the above considerations, the existing datasets are not sufficient for thorough evaluations of SLAM and 3D reconstruction algorithms with different inputs, such as the combination of RGBD images and odometer data. In this paper, for the researches of SLAM and 3D reconstruction in complex indoor scenes for ground wheeled robots, we present a novel and complex indoor benchmark that contains multi-sensor data, semantic information, accurate ground truth trajectories, and 3D point clouds. We select two viewpoints to capture images. The low-viewpoint simulates extremely low robots, such as sweeping floor robots.[2,3] The high viewpoint simulates other service robots, such as food delivery robots.[3] The dataset consists of 22 sequences from nine different scenes in office building and apartment environments, covering highly diverse scenes such as office rooms, bedrooms, and long corridors. Some challenging settings are also designed in the dataset, ranging from different visual conditions (e.g., motion blurs, illumination changes, and weak textures) to different motion modes (e.g., fast motions and straight line motions). Figure 1 exhibits the data information of a sequence. Each sequence provides aligned color and depth images with the resolution of 640 × 480 at 30 Hz, 6-axis IMU measurements at 200 Hz and wheel odometer data at 20 Hz. For trajectory evaluation, we provide accurate 6-DoF



Semantic and multi-sensor data

Ground truth data

**Figure 1.** Data information of sequence Apartment1_1. The trajectory length is 66 m and the duration is 300 s. (a) The top row shows a color image, an aligned depth image and the corresponding semantic segmentation result. The middle and the bottom rows show 6-axis IMU data and 6-axis wheel odometer data over time, respectively. For clarity, only the first 70 s are displayed. (b) The ground truth trajectory (blue line) and the 3D point cloud (top view) are depicted in a unified coordinate system.

camera poses at a high frequency of 100 Hz over the whole sequence. For 3D reconstruction evaluation, we provide a dense 3D point cloud for each scene. Furthermore, 2D segmentation results including a total of 45 semantic classes are provided for the images in our dataset. All the sensors are well calibrated and aligned with the ground truth to ensure that the proposed dataset can be used directly for evaluating SLAM and 3D reconstruction algorithms.

The motivation for this work is to move toward a more comprehensive assessment of SLAM and 3D reconstruction approaches with different inputs. Compared with other benchmarks, our dataset provides rich multi-sensor data including RGBD images, IMU measurements, and wheel odometer data in indoor scenes from a ground wheeled robot viewpoint. To generate ground truth trajectories on the entire sequences and 3D point clouds for both SLAM and 3D reconstruction tasks, we choose GeoSLAM,[1] a mobile 3D laser scanner within two-centimeter accuracy that is in small size, flexible to assemble and suitable for various difficult environments. It empowers users to capture accurate scans and intricate details from the surrounding environments and can simultaneously output the device's trajectory. GeoSLAM is widely employed in industrial applications and can achieve accurate, stable and reliable performances (Fei et al. (2019)). We also evaluate it and find the consistent accuracy and reliability. Furthermore, 2D semantic information is available for evaluating semantics-driven methods. To the best of our knowledge, this is the first indoor benchmark that combines multi-sensor data with semantic information and meanwhile provides ground truth of both trajectories and 3D point clouds. The dataset can be downloaded from https://cid-sims.github.io.

The main contributions of this work are summarized as follows:

- Semantic information and rich multi-sensor data including RGBD images, IMU measurements, and wheel odometer data in complex indoor scenes are furnished firstly for a ground wheeled robot.
- Considering accuracy and portability, we employ GeoSLAM to simultaneously obtain the ground truth trajectories and 3D point clouds within two-centimeter accuracy for the first time.
- We evaluate state-of-the-art SLAM approaches, including SVO2 (Forster et al. (2017)), VINS-Mono (Qin et al. (2018)), VINS-RGBD (Shan et al. (2019)), OpenVINS (Geneva et al. (2020)), ORBSLAM3 (Campos et al. (2021)), and VIW-Fusion (Zhuang (2021)), and 3D reconstruction approaches including ElasticFusion (Whelan et al. (2015; 2016)), InfiniTAM (Kähler et al. (2016)), BAD-SLAM (Schops et al. (2019)), Voxblox (Grinvald et al. (2019)), and Kimera (Rosinol et al. (2020)) on our dataset and sufficiently analyze the experimental results.

The rest of the paper is structured as follows. Section 2 summarizes the currently open datasets and compares them with our dataset. Section 3 describes the configuration of the sensor system for acquiring data. Section 4 introduces the calibration process for the sensors equipped on our data acquisition robots. Section 5 explains the structure of the dataset. Section 6 displays the experimental results of state-of-the-art SLAM and 3D reconstruction methods on our dataset. Finally, Section 7 discusses the conclusion and future directions of this study.

## 2. Related work

For indoor scenes, several datasets have been released to facilitate visual SLAM and 3D reconstruction researches. We briefly review the most relevant datasets in Table 1.

### 2.1. Datasets with multiple sensors

The TUM VI dataset (Schubert et al. (2018)) consists of stereo gray images and IMU sequences collected with a handheld device. Similar to our benchmark, a few datasets are targeted at ground wheeled robots. The TUM RGBD dataset (Sturm et al. (2012)) is partly collected by a ground wheeled robot in an office and an industrial hall. It only contains RGBD sequences from a Microsoft Kinect. The VCU RVI benchmark (Zhang et al. (2020)) provides sequences using ground wheeled robots with RGBD images and 6-axis IMU measurements in various indoor environments. Due to the limitation of motion capture equipment, the ground truth camera poses are partially valid for long trajectories. M2DGR (Yin et al. (2021)) is collected by a ground wheeled robot with a rich sensor suite including five kinds of cameras, IMUs, LiDAR, and navigation signals. The ground truth trajectories are obtained with a motion capture system for indoor environments and a laser tracker for outdoor environments. The OpenLORIS-Scene Datasets (Shi et al. (2020)) are collected with commodity sensors (RGBD cameras, a LiDAR, IMUs, and wheel odometers) to facilitate the study of SLAM within long-term constantly changing environments. Currently, there are very few indoor datasets containing wheel odometer data.

### 2.2. Datasets for SLAM and 3D reconstruction

Constrainedly, the above benchmarks are not suitable for comprehensively evaluating 3D reconstruction algorithms since they do not provide any information about the scene geometry. Without the 3D models of the scenes, these datasets can only be used to evaluate the localization of a SLAM algorithm, regardless of the mapping. There are currently a few public benchmarks specifically for evaluating SLAM and 3D reconstruction algorithms meanwhile. The ICL-NUIM dataset (Handa et al. (2014)) is synthetic for benchmarking visual odometry, SLAM, and surface reconstruction with exact ground truth trajectories and surface models. CoRBS (Wasenmüller et al. (2016)) is the first to

**Table 1.** Comparison of Datasets for SLAM and 3D Reconstruction.

| Dataset | Environment | Carriers | Cameras | IMU | Odometer | Trajectory | Scene model | Semantic |
|---|---|---|---|---|---|---|---|---|
| TUM RGBD<br>Sturm et al. (2012) | Indoors | Handheld/ Robot | RGBD | × | × | √ | × | × |
| ICL NUIM<br>Handa et al. (2014) | Synthetic | Handheld | RGBD | × | × | √ | O | × |
| EuRoC MAV<br>Burri et al. (2016) | Indoors | MAV | Stereo gray | √ | × | √ | O | × |
| CoRBS<br>Wasenmüller et al. (2016) | Indoors | Handheld | RGBD | × | × | √ | √ | × |
| ScanNet<br>Dai et al. (2017) | Indoors | Handheld | RGBD | √ | × | √ | √ | √ |
| Robot@Home<br>Ruiz-Sarmiento et al. (2017) | Indoors | Robot | RGBD | × | × | √ | √ | √ |
| TUM VI<br>Schubert et al. (2018) | In-/Outdoors | Handheld | Stereo gray | √ | × | O | × | × |
| ADVIO<br>Reina et al. (2018) | In-/Outdoors | Handheld | RGB | √ | × | √ | √ | × |
| Replica<br>Straub et al. (2019) | Indoors | Handheld | Wide-angle/ RGBD/Grayscale | √ | × | √ | √ | √ |
| VCU RVI<br>Zhang et al. (2020) | Indoors | Handheld/ Robot | RGBD | √ | × | O | × | × |
| OpenLORIS<br>Shi et al. (2020) | Indoors | Robot | Stereo fisheye/ RGBD | √ | √ | √ | × | × |
| M2DGR<br>Yin et al. (2021) | In-/Outdoors | Robot | Fisheye/Sky-point/ Pinhole/Event/ Infrad | √ | × | √ | × | × |
| Hilti-Oxford<br>Zhang et al. (2023) | In-/Outdoors | Handheld | Fisheye cameras | √ | × | √ | √ | × |
| OURS | Indoors | Robot/ Handheld | RGBD | √ | √ | √ | √ | √ |

Scene model refers to any form of the 3D geometry of the scene, such as 3D point cloud, and mesh. O means that the item is partially available.

provide the combination of real depth and color data with ground truth camera trajectories and 3D models. The EuRoC MAV dataset (Burri et al. (2016)) consists of 11 indoor sequences recorded with a VI (Visual-Inertial) sensor from a Micro Aerial Vehicle (MAV), providing ground truth poses and 3D point clouds from a motion capture system and a laser tracker. ADVIO (Reina et al. (2018)) and Hilti-Oxford (Zhang et al. (2023)) provide both ground truth camera trajectories and scene geometries with handheld devices. However, sequences provided in these datasets are short and limited to small rooms.

## 2.3. Datasets with semantic information

Indoor semantic segmentation has enabled breakthroughs for several tasks such as objection segmentation and scene understanding. The NYU v2 dataset (Nathan Silberman and Fergus (2012)) offers RGBD images capturing 464 diverse indoor scenes with detailed annotations. ScanNet (Dai et al. (2017)) is annotated with 3D camera poses, surface reconstructions, and semantic segmentations in the context of RGBD scene understanding. The Replica Dataset (Straub et al. (2019)) collects time-aligned raw IMU, RGB, depth, and wide-angle grayscale sensor data. It can serve as a generative model for benchmarking SLAM and dense reconstruction systems. In addition, Robot@Home (Ruiz-Sarmiento et al. (2017)) provides observations from a mobile robot with four RGBD cameras and a 2D laser scanner, including 3D reconstructions and 2D geometric maps of the inspected rooms, which are both annotated with the ground truth categories of the surveyed rooms and objects. We provide 2D semantic annotations concentrating on SLAM and 3D reconstruction tasks for ground wheeled robots in real indoor scenes.

In summary, the aforementioned datasets are not complete because of lacking rich sensory sources, missing entire ground truth trajectories or 3D point clouds, no semantic information, and insufficient challenges for 3-DoF motions. To address all these issues, we release a novel benchmark for indoor ground wheeled robot navigation, containing sufficient data sequences with semantic information and multi-sensor data. More importantly, we provide entire trajectories and 3D point clouds as ground truths.

## 3. Sensor setup

In this section, we introduce the data acquisition equipment. We design a ground wheeled robot equipped with an RGBD-IW (an RGBD camera, an IMU, and a wheel odometer) sensor suite for data acquisition. In order to obtain ground truth trajectories and 3D point clouds, a high-precision 3D laser scanner called GeoSLAM is fixed on the top of the robot. In addition, to enrich the dataset, two different viewpoints (high and low) are selected for capturing images. Figure 2 illustrates the robot setup, the coordinate systems of all sensors, and two different viewpoints for the RGBD camera.

### 3.1. RGBD camera

The Intel RealSense D455, a global shutter RGBD camera, is chosen to capture aligned RGB images and depth maps with the resolution of 640 × 480 pixels at 30 fps. The ideal depth range is 0.6 m to 6 m from the image plane, and the error is less than 2% within 4 m.[5]

### 3.2. IMU

The RealSense D455 integrates an IMU (Bosch BMI055) to provide 3-axis accelerometer and 3-axis gyroscope measurements at 200 Hz. The resolutions of the accelerometer and the gyroscope are 0.98 $mg$ and $0.004°/s$ and the noise densities are 150 $\mu g/\sqrt{Hz}$ and 0.014 $°/s/\sqrt{Hz}$, respectively.

### 3.3. Wheel odometer

The WHEELTEC R550 (DIFF) PLUS supports the movement of our data acquisition equipment. Two differential wheels are mounted on a common axis (baseline) and each of them is equipped with a giant magnetoresistive (GMR) high-precision encoder, which provides local angular rate readings. Both wheel encoders have the resolution of 500 pulses per round and support a maximum speed of 1.2 m/s and a maximum load capacity of 4 kg.[6]

Using the angular rate readings, we can solve for the linear velocities $v$ and angular velocities $\omega$ at the center of the baseline by the following equations:

$$v = \frac{r_l\omega_l + r_r\omega_r}{2}, \omega = \frac{r_r\omega_r - r_l\omega_l}{a}, \tag{1}$$

where $\omega_l$, $\omega_r$ are angular rate readings of the wheels, $r_l$, $r_r$ are the corresponding radii, and $a$ denotes the baseline length. The wheel odometer measurements are logged at 20 Hz.

### 3.4. 3D laser scanner

We are the first to use GeoSLAM ZEB Horizon,[1] a handheld 3D LiDAR scanner with powerful offline SLAM technology, to generate precise ground truth trajectories and dense point clouds for large-scale indoor environments. GeoSLAM can collect 300,000 points per second. It provides trajectories at 100 Hz, which are in the same coordinate system as the corresponding 3D point clouds. With an effective range of 100 m, the GeoSLAM ZEB Horizon achieves a 1-sigma accuracy of 6 mm and 19 mm at

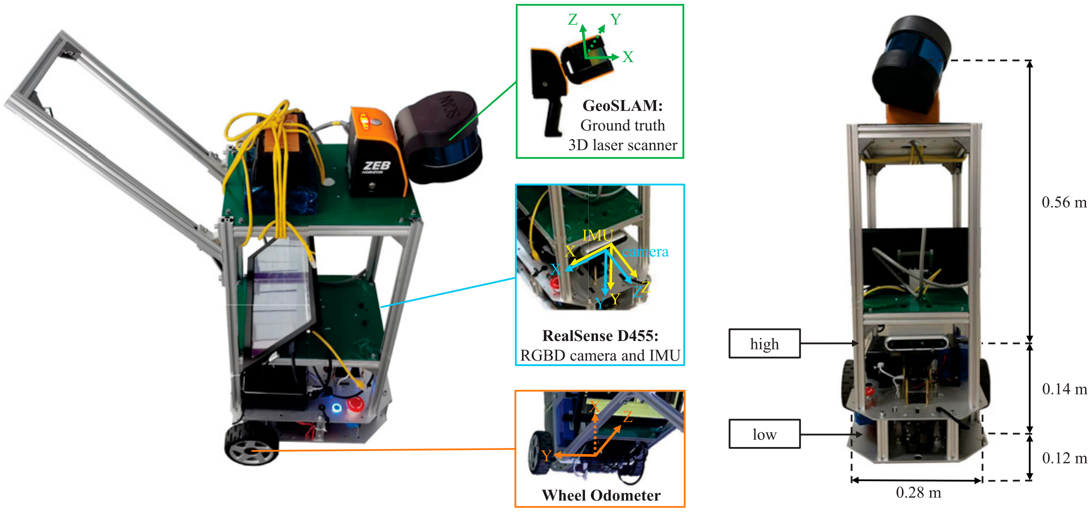


**Figure 2.** Data acquisition robot. The RealSense D455 is responsible for collecting aligned color and depth images, while the built-in IMU collects 3-axis acceleration and 3-axis gyroscope data. The wheel odometer provides 3-axis linear velocity and 3-axis angular velocity at the midpoint of the baseline of the two wheels. The 3D laser scanner (GeoSLAM) acquires ground truth trajectories and 3D point clouds.

2-sigma. Figure 3 shows an example of the $z$-axis coordinates changing over time in a long trajectory. Taking advantage of the fact that the robot moves on the ground, the $z$-axis coordinate varying within 0.008 m on average reflects the validity and accuracy of the ground truth in our dataset. A detailed analysis of the GeoSLAM's accuracy can be found in their accuracy report.[7]

## 4. Calibration

Our benchmark provides accurate intrinsic and extrinsic calibration parameters and time synchronization among different sensors and GeoSLAM. In this section, we describe the calibration process. Table 2 summarizes the parameters and the calibration tools. All the parameters are reported in *calibration.yaml*.

### 4.1. Camera intrinsic calibration

For camera intrinsic calibration (Liebowitz and Zisserman (1998); Zhang (2000); Furgale et al. (2013)), we record image sequences with 6-DoF motions in front of an ArUco marker board (Romero-Ramirez et al. (2018)). Using the Kalibr toolbox (Furgale et al. (2013)), the intrinsic parameters of the camera, including focal length, optical center, and distortion parameters, are calibrated based on the pinhole camera model and the radial-tangential distortion model.

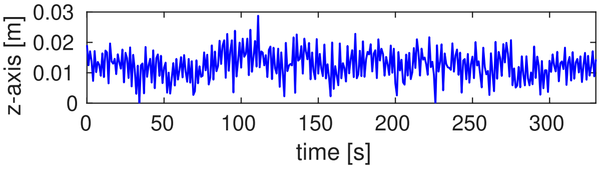


**Figure 3.** $Z$-axis coordinate of a GeoSLAM trajectory when the robot is moving on the ground plane.

### 4.2. IMU intrinsic calibration

For IMU intrinsic calibration, we record static IMU sequences at 200 Hz for more than 3 h. By means of the Kalibr (Furgale et al. (2013)) toolbox, the intrinsic parameters including noise densities and random walk biases are calibrated based on the assumption that the IMU measurements are perturbed by a rapidly fluctuating additive Gaussian noise (white noise) and a slowly varying random walk noise (bias).

### 4.3. Camera-IMU extrinsic calibration

The RGBD camera and the IMU are time-synchronized under the same clock system. To make the calibration fully observable, we ensure sufficient rotational and accelerated motions in camera-IMU sequences in front of an ArUco marker board. The transformation matrix $T_c^b$ (from the RGBD camera coordinate system to the IMU coordinate system) is estimated by Furgale et al. (2013).

### 4.4. Camera-wheel odometer extrinsic calibration

*4.4.1. Time synchronization.* Motivated by Usenko et al. (2020), we compare gyroscope readings from the IMU sensor with angular readings from the wheel odometer to obtain the optimal time offset since the gyroscope $y$-axis data and the wheel odometer $z$-axis data tend to change in opposite directions during the motion. In particular, we iterate over a time interval to find the initially optimal time compensation corresponding to the minimum error, which is defined as the difference between two matching readings.

**Table 2.** Calibration Parameters and Tools.

| Sensors | Parameters | Tools |
|---|---|---|
| Camera | Focal length ($f_x$, $f_y$)<br>Optical center ($c_x$, $c_y$)<br>Distortion parameters$k_1$, $k_2$, $d_1$, $d_2$ | Furgale et al. (2013) |
| IMU | Noise density $\sigma_g$, $\sigma_a$<br>Random walk bias $b_g$, $b_a$ | Furgale et al. (2013) |
| Camera-IMU | Time offset<br>Transformation matrix $T_c^b$ | Furgale et al. (2013) |
| Wheel odometer | Wheel radius $r_l$, $r_r$<br>Baseline length $a$ | Dongfu Zhu (2019) |
| Camera-Odometer | Time offset<br>Transformation matrix $T_c^o$ | Usenko et al. (2020)<br>Dongfu Zhu (2019) |
| Camera-GeoSLAM | Time offset<br>Transformation matrix $T_c^g$ | Usenko et al. (2020)<br>Tsai et al. (1989) |

Afterward, by fitting a quadratic curve with 20 error values in the neighborhood around the minimum error, the final time offset corresponding to the lowest point of the curve is obtained at the level of $10^{-3}$ seconds. We have corrected the time delay already for each sequence in our dataset, so no further action is required by users. Figure 4(a) and (b) show an example of the two-step synchronization result.

*4.4.2. Spacial calibration.* We record camera-IMU-wheel odometer sequences by moving the acquisition device on the ground and making the camera orient to a fixed checkerboard with a grid size of 10 × 10. After time synchronization, the intrinsic and extrinsic parameters of the wheel odometer are estimated together using the method proposed in Dongfu Zhu (2021).

### *4.5. Camera-3D laser scanner extrinsic calibration*

*4.5.1. Time synchronization.* For each sequence, we perform a coarse-to-fine time alignment between the IMU and GeoSLAM. Since no Network Time Protocol (NTP) exists between GeoSLAM and our control computer, a coarse adjustment is conducted by aligning the start frame of the IMU measurements and the ground truth poses. For fine synchronization, we compute the optimal time using Usenko et al. (2020). In particular, we calculate the angular velocities using the central differences in the orientation from the ground truth poses, and then iterate over a time interval to find the optimal time compensation corresponding to the minimum error, which is computed based on the difference of the two matched angular velocities. Besides, a sub-step to fit the quadratic curve is carried out to achieve a millisecond-level time synchronization. An example of the aligned results is shown in Figure 4(c) and (d).

*4.5.2. Spacial calibration.* As GeoSLAM already calibrates the LiDAR's intrinsic parameters and provides a point cloud for each sequence without raw data, no additional intrinsic calibration (Pandey et al. (2010); Huang et al. (2020); Glennie and Lichti (2010)) is performed. In order to evaluate the results of SLAM approaches with ground truth trajectories, the extrinsic transformation matrix $T_c^g$ from the camera coordinate system to the GeoSLAM coordinate system is required. Different from methods using raw LiDAR data (Huang and Grizzle (2020); Jeong et al. (2019); Jiang et al. (2018)), our extrinsic calibration is based on GeoSLAM's output trajectories using a hand-eye calibration approach (Tsai et al. (1989)). We record image sequences and ground truth poses with 6-DoF motions in front of a checkerboard. As depicted in Figure 5, the checkerboard is fixed and its upper left corner is the origin of the world coordinate system. Knowing the size of the checkerboard, the 3D coordinates of each checkerboard corner can be calculated. After extracting the checkerboard corners in the images and solving the corresponding camera poses $P_w^c$ with the Perspective-n-Point (PnP) algorithm, $T_c^g$ is estimated through a hand-eye calibration approach proposed in Tsai et al. (1989). To be specific, knowing a group of relative camera poses $P_{c_i}^{c_j}$ and relative ground truth poses $P_{g_i}^{g_j}$, $T_c^g$ can be estimated from

$$T_c^g * P_{c_i}^{c_j} = P_{g_i}^{g_j} * T_c^g. \tag{2}$$

We select 20 sets of positions with sufficient motion for the calculation and the translation result is close to what we manually measured. Figure 6 displays an example of coloring the ground truth point cloud with RGB images according to the calibration results, verifying the accuracy of the calibration.

## 5. Dataset

Our dataset consists of 22 sequences captured from nine different environments using a ground wheeled robot. All the data is recorded in real environments under normal walking conditions, where specular reflections, sunlight, shadows, and dynamic objects are commonly observed. Figure 7 shows the data collection environments. Each sequence provides aligned color images and depth maps, semantically annotated images, 6-axis inertial measurements, and wheel odometer data. The entire ground truth poses and the 3D point clouds for each scene are supported

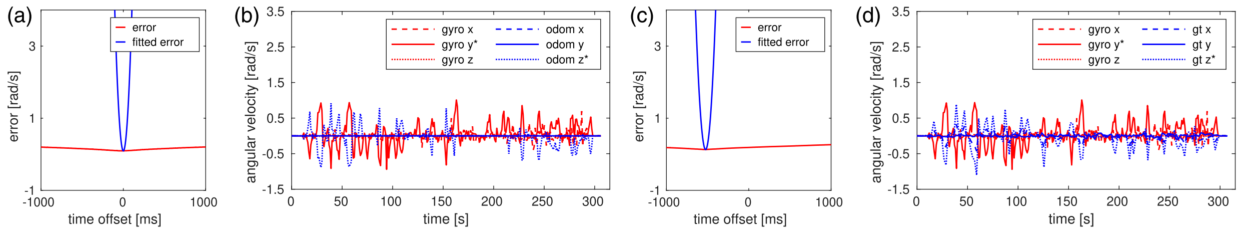


**Figure 4.** Time synchronization. (a) and (b) indicate time synchronization between the IMU and the wheel odometer and (c) and (d) indicate time synchronization between the IMU and GeoSLAM in sequence Floor14_1. (a) and (c) show the time offsets and errors, where the red line represents each candidate time offset and its error calculated by the iterative process and the blue line is the fitted quadratic curve near minimum error. (b) and (d) show the synchronization results, achieving time alignment with the error of 0.0942 rad/s and 0.1314 rad/s, respectively. According to the frames in Figure 2, the *y*-axis data of the gyroscope and the *z*-axis of GeoSLAM/wheel odometer (marked with asterisks) have the opposite trend of change.

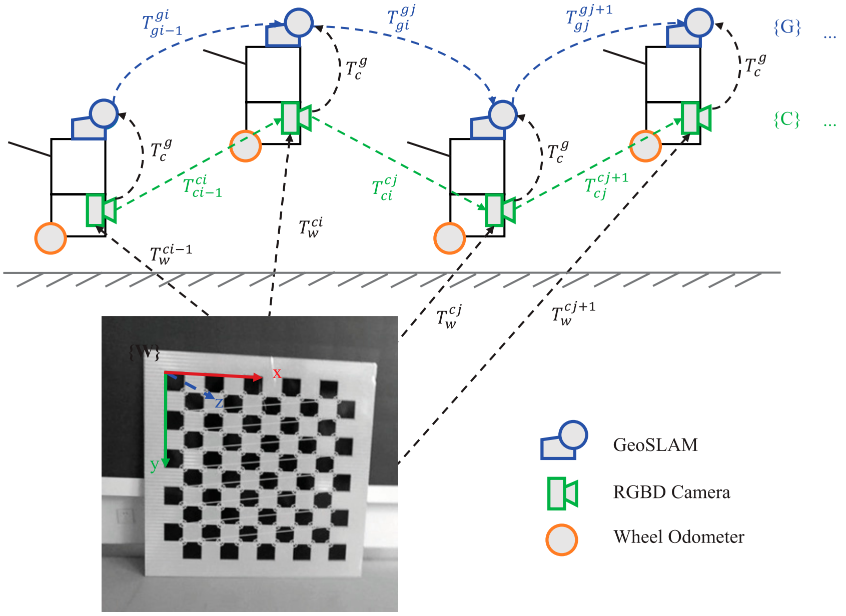


**Figure 5.** Diagram of the camera-GeoSLAM extrinsic parameters calibration process.

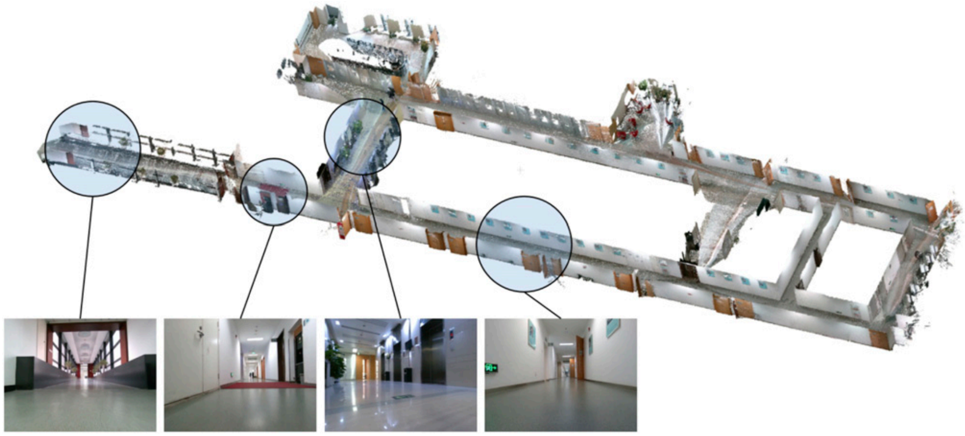


**Figure 6.** Colored 3D point cloud of Floor3 using the camera-GeoSLAM calibration results.

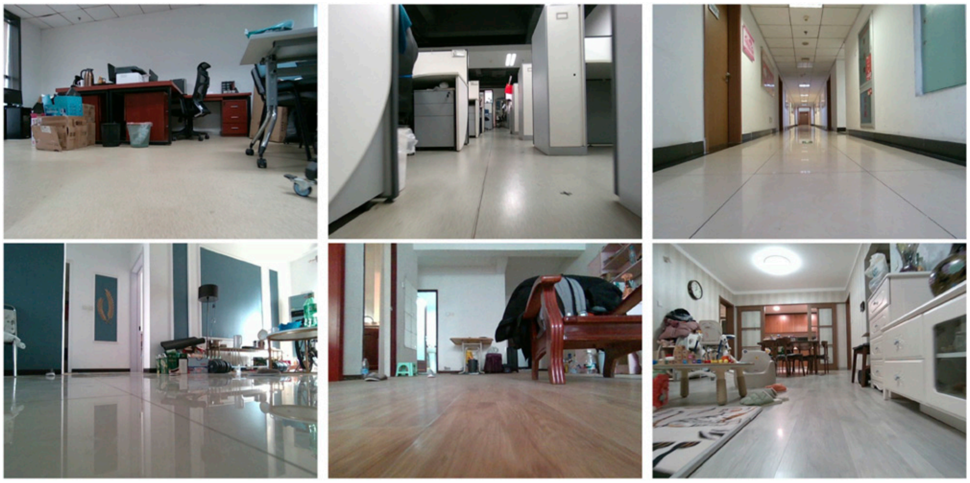


**Figure 7.** Data collection environment. The top row shows the office building environments and the bottom row shows the apartment environments.

by GeoSLAM. With precise calibration of intrinsic and extrinsic parameters among the sensors, our dataset can be used directly for evaluating SLAM and 3D reconstruction approaches. The statistical information of our dataset is given in Table 3. Trajectories in different scenes are captured under different visual or motion conditions. In the following, we will give a detailed description.

## *5.1. Sequence description*

Sequences in this dataset are mainly from a ground wheeled robot with two camera views (approximately 0.26 m (high) and 0.12 m (low) above the ground), except for two sequences that contain handheld situations when going downstairs. The dataset covers a large variety of scenes and motion modes, as shown in Table 3. There are some challenging scenarios such as motion blurs, illumination changes, reflective objects, and weak textures, as shown in Figure 8. To support different initialization modes (stationary and motion initialization), each sequence starts with a stationary time for about 5 s. Furthermore, each sequence contains several loops. The sequences can be divided into the following two categories.

*5.1.1. Office building.* We collect 13 sequences in a typical office building, which covers six different scenes. There is obvious sunlight at the window, causing parts of the images to be overexposed. Sequences in floor scenes are challenging because of insufficient rotation and weak textures.

- Office: Three sequences are captured inside a small office room (about 6 m × 7 m), where all the objects are static.
- Floor14: Three sequences are captured in the open office area on the 14th floor of a building, including dark images and dynamic pedestrians.
- Downstairs: A sequence is collected in the stairwell by manually carrying the robot going downstairs. The sequence is under 6-DoF motion and the wheel odometer data is missing.
- Two Floors: A long sequence is recorded by firstly moving on the 14th floor, then going downstairs, next moving on the 13th floor, and finally going back to the 14th floor. When passing through the stairs, the wheel odometer data is unavailable.
- Floor3: Three sequences are captured along corridors (about 70 m) with straight line motion, dynamic pedestrians, reflective objects, and an uphill movement.
- Floor13: Two long sequences are recorded in a long corridor (about 65 m).

*5.1.2. Apartment.* We record nine sequences in real living environments, which cover three different apartments. The apartments are about 10 m × 10 m, including a living room, two or three bedrooms, a bathroom, and a kitchen. These rooms are cluttered with small and unstructured obstacles. Sequences in apartment scenes contain rich loops and rotational motions. There are also challenging situations such as motion blurs, weak textures, and reflective objects.

- Apartment1: Three sequences are captured in an apartment during the day, where the tiled floor is reflective. There exists significant sunlight at the windows.
- Apartment2: Three sequences are collected in an apartment during the day. The apartment is messy, in which dust and litter cause uneven floors. There is strong sunlight near the windows.
- Apartment3: Three sequences are acquired in an apartment at night with the lights turned on.

## *5.2. Semantic annotation*

The dataset is annotated with 45 label classes. We provide per-pixel semantic labels of the images and represent the results in grayscale images where the pixel value corresponds to the class identity (from 0 to 44). Some segmentation results are visualized in Figure 9. In the following, we describe our semantic annotation details.

*5.2.1. Manual annotation.* Since sequences in the same scene cover the same objects, we select the longest sequence in each scene to extract keyframes. According to the degree of the overlap, a keyframe is selected every 30 frames in the office building environments and every 60 frames in the apartment environments. In total, approximately 2% of the images are manually segmented in great details as training data using a self-developed 2D masking tool.

*5.2.2. Network and training.* In our dataset, all images, including those manually annotated, are labeled using a semi-supervised semantic segmentation algorithm called ST++ (Yang et al. (2022)), which adopts an image-level strategy for holistic contextual information. The algorithm performs selective re-training via prioritizing reliable unlabeled images based on holistic prediction-level stability. Through selection and sorting, reliable pseudo-labels are selected for re-training to mitigate the impact of false pseudo-labels. Choosing *deeplabv3 plus with resnet50* (Chen et al. (2018)) as the core structure of ST++, we train a network for 200 epochs for each scene with default parameters. The means of the Mean Intersection over Union (mIOU) for the test images are detailed in Table 4. Our segmentation results demonstrate accuracy equivalent to the ground truth, as shown in Figure 10. In scenes such as Floor14 and Two Floors, the presence of a large number of small structures (e.g., chair legs) contributes to the low mIOUs. It's worth noting that bad pixel labels may occur around the edges of the objects and the small structures, and thus users should exercise caution when using it.

**Table 3.** Statistics on the sequences. (O Means Partially Available.)

| Scene | | Sequence | Viewpoint | Length (m) | Duration (s) | # of frames | Wheel | Complexity | Description |
|---|---|---|---|---|---|---|---|---|---|
| Office building | Office | Office_1 | High | 42.01 | 141.25 | 4250 | √ | ★☆☆☆☆ | Static, small room, simple track |
| | | Office_2 | High | 90.31 | 247.00 | 7396 | √ | ★★☆☆☆ | Static, small room, medium track |
| | | Office_3 | High | 95.25 | 238.99 | 7132 | √ | ★★★☆☆ | Static, small room, complex track |
| | Floor14 | Floor14_1 | High | 103.70 | 184.30 | 5530 | √ | ★★★★☆ | Short straight line motion, dynamic objects |
| | | Floor14_2 | High | 106.40 | 183.00 | 5459 | √ | ★★★★★ | Long straight line motion, dynamic objects, dark images |
| | | Floor14_3 | High | 180.43 | 299.00 | 8918 | √ | ★★★★★ | Long straight line motion, dynamic objects |
| | Two Floors | 14-13-14 | High | 249.96 | 379.72 | 11,379 | O | ★★★★★ | Partial handheld, two floors |
| | Downstairs | 14-13-12 | High | 21.89 | 70.40 | 2138 | × | ★★★★☆ | Handheld, downstairs |
| | Floor3 | Floor3_1 | High | 85.61 | 128.21 | 3834 | √ | ★★★★☆ | Short straight line motion, dynamic objects, reflective objects |
| | | Floor3_2 | High | 150.55 | 187.06 | 5649 | √ | ★★★★★ | Long straight line motion, dynamic objects, reflective objects |
| | | Floor3_3 | High | 196.46 | 243.74 | 7299 | √ | ★★★★★ | Long straight line motion, dynamic objects, reflective objects, upslope |
| | Floor13 | Floor13_1 | High | 130.43 | 177.71 | 5336 | √ | ★★★★☆ | Long straight line motion, dynamic objects |
| | | Floor13_2 | High | 135.10 | 194.17 | 5820 | √ | ★★★★☆ | Long straight line motion, dynamic objects |
| Apartment | Apartment1 | Apartment1_1 | Low | 66.01 | 305.99 | 9159 | √ | ★★★★☆ | Tiled floor, daylight, reflective objects, simple track |
| | | Apartment1_2 | High | 77.18 | 206.99 | 6179 | √ | ★★★★★ | Tiled floor, daylight, reflective objects, medium track, fast motion |
| | | Apartment1_3 | High | 154.00 | 376.99 | 11,280 | √ | ★★★★★ | Tiled floor, daylight, reflective objects, complex track |
| | Apartment2 | Apartment2_1 | Low | 68.50 | 239.99 | 7160 | √ | ★★★★☆ | Wood floor, daylight, messy rooms, simple track |
| | | Apartment2_2 | High | 85.88 | 334.65 | 10,076 | √ | ★★★★★ | Wood floor, daylight, messy rooms, medium track |
| | | Apartment2_3 | Low | 100.04 | 387.99 | 11,596 | √ | ★★★★★ | Wood floor, daylight, messy rooms, complex track |
| | Apartment3 | Apartment3_1 | Low | 73.22 | 382.99 | 11,437 | √ | ★★★★☆ | Wood floor, night, dynamic objects, simple track |
| | | Apartment3_2 | Low | 84.42 | 259.21 | 9954 | √ | ★★★★★ | Wood floor, night, dynamic objects, medium track, fast motion |
| | | Apartment3_3 | High | 147.96 | 361.00 | 10,800 | ✓ | ★★★★★ | Wood floor, night, dynamic objects, complex track |

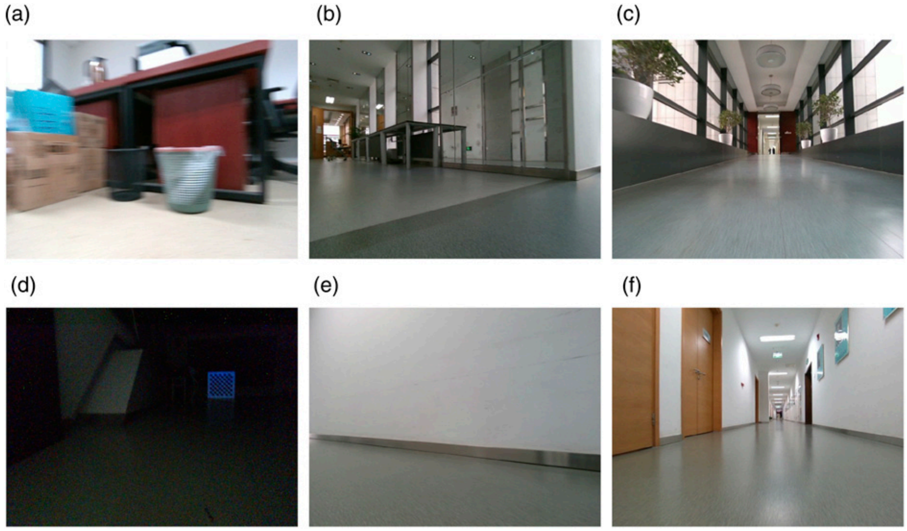


**Figure 8.** Examples of the challenging scenarios in our dataset. (a) Motion blur, (b) Mirror, (c) Slope, (d) Illumination change, (e) Weak texture, and (f) Long corridor

## 5.3. Ground truth

The ground truth 3D point clouds are acquired simultaneously with the corresponding trajectories using GeoSLAM. In the same scenes, for different trajectories, the captured point clouds are in different coordinate systems. Using CloudCompare,[8] We merge the 3D point clouds in the same scenes into a complete point cloud. Initially, we conduct a coarse alignment ($T_c$) by manually selecting point correspondences, followed by a densely sampled fine alignment ($T_f$) using the Iterative Closest Point (ICP) method. The average error of the ICP alignment is 0.03 m. Based on the two steps, a global calibration for each scene is established so that the 3D point clouds are unified into a global coordinate system. After manually eliminating moving objects, we provide a complete point cloud in the standard *.las* file format for each scene. Similarly, the transformation of trajectories from the original world coordinate system to the global coordinate system is achieved by applying the following rigid transformation:

$$T_{cam\_global} = T_f * T_c * T_{cam\_origin}, \tag{3}$$

in which, $T_{cam\_origin}$ denotes the camera poses from the camera coordinate system to the original world coordinate system and $T_{cam\_global}$ denotes the camera poses from the camera coordinate system to the new global coordinate system.

## 5.4. File format

We provide spatiotemporally aligned data and ground truth poses for every sequence as well as a precise 3D point cloud for each scene. Additionally, extrinsic and intrinsic calibration parameters are provided in a *.yaml* file. The data is organized as:

```
/
├─calibration.yaml
├─annotations.txt
└─office_building/apartment
   ├─<scene_name>
   │  ├─pointcloud.las
   │  ├─<sequence_name>.zip
   │  │  ├─color
   │  │  │  ├─1673420102.156301.png
   │  │  │  ├─1673420102.189610.png
   │  │  │  └─...
   │  │  ├─depth
   │  │  │  ├─1673420102.156301.png
   │  │  │  ├─1673420102.189610.png
   │  │  │  └─...
   │  │  ├─mask
   │  │  │  ├─1673420102.156301.png
   │  │  │  ├─1673420102.189610.png
   │  │  │  └─...
   │  │  ├─associations.txt
   │  │  ├─imu.txt
   │  │  ├─odom.txt
   │  │  └─groundtruth.txt
   │  └─...
   └─...
```

Specifically, the folders and files contain:

- color/: color images, named with timestamps, image format: png, three channels, 8 bit per channel.
- depth/: depth images, named with timestamps, image format: png, one channel, 16 bit per channel, values in meters scaled by factor 1000. In other words, the unit of the depth measurement is the millimeter.
- mask/: semantic images, named with timestamps, image format: png, three channel, 8 bit per channel, values of any channel corresponding to the class labels.

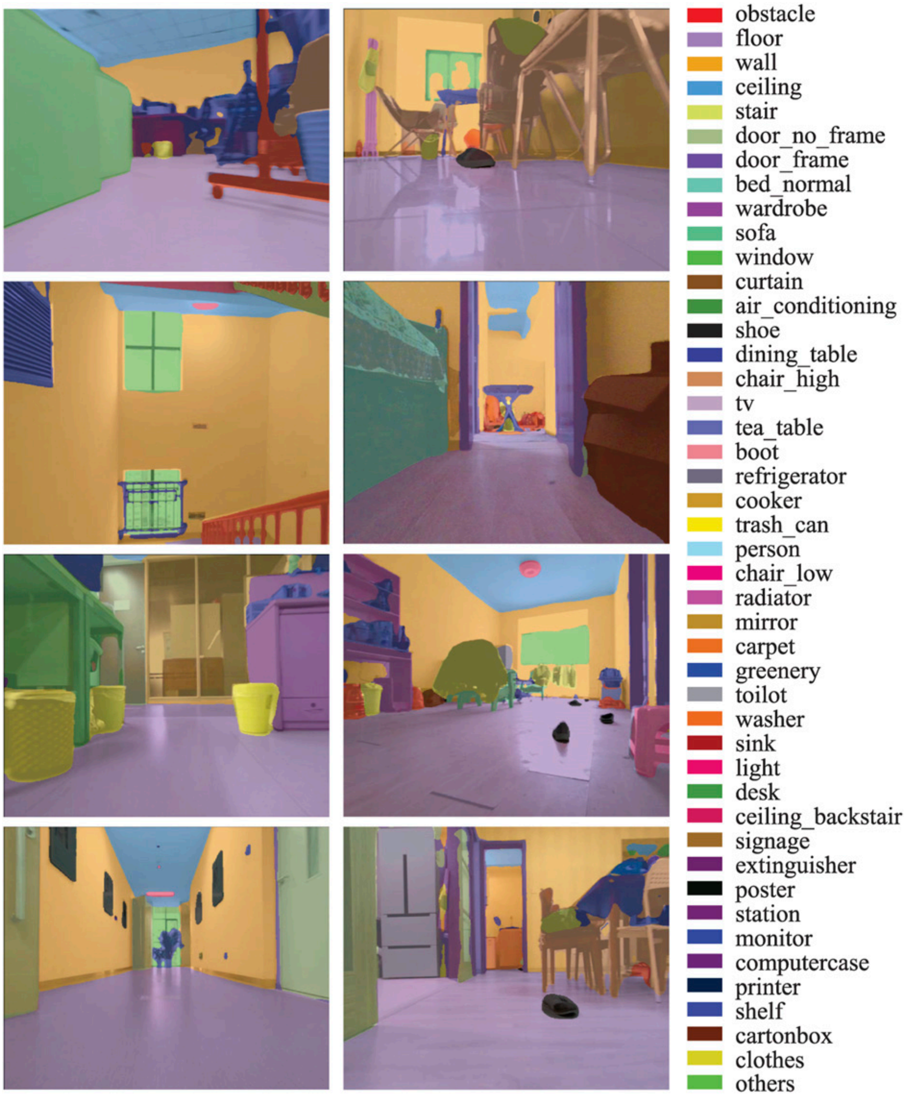


**Figure 9.** Example images with annotations over the 45 semantic classes contained in our dataset. The labels are encoded in different colors.

**Table 4.** Mean Test mIOU (%) for each Scene.

| Scene | Office | Floor14 | Two Floors | Downstairs | Floor3 | Floor13 | Apartment1 | Apartment2 | Apartment3 |
|---|---|---|---|---|---|---|---|---|---|
| Mean test mIOU | 72.64 | 67.67 | 69.01 | 77.53 | 78.99 | 81.40 | 75.71 | 76.31 | 78.19 |

- calibration.yaml: extrinsic and intrinsic parameters for the sensors.
- annotations.txt: correspondences between the labels (pixel values) and the categories with description sentences.
- associations.txt: associated timestamps for the color images and the corresponding depth images, format: *color_timestamp*[*s*] *depth_timestamp*[*s*].
- imu.txt: timestamped gyroscope and accelerometer data from the IMU, format: *timestamp*[*s*] $g_x[rad/s]$ $g_y[rad/s]$ $g_z[rad/s]$ $a_x[m/s^2]$ $a_y[m/s^2]$ $a_z[m/s^2]$.
- odom.txt: timestamped pose positions, orientations and twist linear/angular velocities from the wheel odometer, format: *timestamp*[*s*] $x[m]$ $y[m]$ $z[m]$ $q_x$ $q_y$ $q_z$ $q_w$ $v_x[m/s]$ $v_y[m/s]$ $v_z[m/s]$ $\omega_x[rad/s]$ $\omega_y[rad/s]$ $\omega_z[rad/s]$.
- groundtruth.txt: timestamped translation vectors and unit quaternions (in the form of Hamilton), format: *timestamp* [*s*] $t_x[m]$ $t_y[m]$ $t_z[m]$ $q_x$ $q_y$ $q_z$ $q_w$, where $q_w$ is the real part.
- pointcloud.las: unclolored 3D point cloud, each point has its position $x[m]$, $y[m]$, $z[m]$ and the intensity.

## 6. Evaluation

With precise ground truth trajectories and 3D point clouds, our dataset offers practical and challenging sequences for

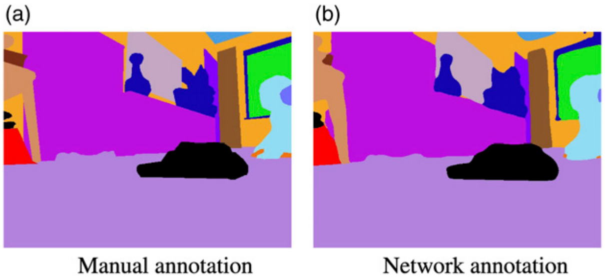


**Figure 10.** Camparision between (a) manual annotation and (b) network annotation. The network training results accurately reflect the manual annotated 2D semantic segmentation.

assessing both SLAM and 3D reconstruction tasks with multi-information input. In this section, we compare state-of-the-art vision-based SLAM and 3D reconstruction algorithms to validate the effectiveness of our dataset.

## 6.1. SLAM evaluation

We evaluate the overall performance of a SLAM system considering pose estimation accuracy and employ the Root Mean Square Error (RMSE) proposed in Sturm et al. (2012) as the evaluation criteria. Assume that $\mathbf{p}_{1:n}$, $\mathbf{q}_{1:n} \in R^3$ are time-synchronized positions (from an arbitrary reference frame) of the estimated trajectory and the ground truth trajectory, respectively. We calculate an alignment matrix $\boldsymbol{S}$ between the estimated poses and the ground truth poses using the ICP method. The RMSE of the two trajectories is defined as

$$RMSE: = \left(\frac{1}{n}\sum_{i=1}^{n}\|\mathbf{S}\mathbf{p}_i - \mathbf{q}_i\|^2\right)^{\frac{1}{2}}, \tag{4}$$

where $\boldsymbol{S}$ represents the transformation matrix mapping the estimated trajectory onto the ground truth trajectory. Thus, the RMSE measures the translational components of the absolute trajectory error (ATE) over all time.

We test several state-of-the-art SLAM methods on the dataset. SVO2 (Forster et al. (2017)) is a semi-direct vision odometry method that supports monocular and IMU data as input with a loop detection module. OpenVINS (Geneva et al. (2020)) is a filter-based system that utilizes RGB images and IMU data as input. To compare performances on different input sensors, we extend OpenVINS to OpenVINS-Depth by incorporating depth information. When the anchor frame has a reliable depth, we use the depth directly instead of the original triangulation operation to further improve accuracy and efficiency. VINS-Mono (Qin et al. (2018)) is an optimization-based approach with loop detection and global map module. VINS-RGBD (Shan et al. (2019)) extends VINS-Mono by adding depth information in the process of motion initialization and feature triangulation. VIW-Fusion (Zhuang (2021)), a visual-inertial-wheel fusion odometry system, is developed based on VINS-Fusion (Qin et al. (2019b); Qin and Shen (2018); Qin et al. (2018, 2019a)) by adding wheel odometer information and plane constrains. ORBSLAM3 (Campos et al. (2021)) is a versatile SLAM system, which can perform visual, visual-inertial, and multi-map SLAM with monocular, stereo, and RGBD cameras.

Considering that there is a certain randomness of estimation, we run 5 times for each system on Intel i9-13900KS CPU and the median results are summarized in Table 5. The results are from the EVO toolbox (Grupp (2017)) and all estimated poses are used to calculate the alignment matrix $\boldsymbol{S}$. Some trajectories estimated by these methods and the corresponding ground truths are presented in Figure 11. We can infer from the results that SVO2 (with loop) is easily drifting and suffers from large-scale errors caused by insufficiently robust initialization. OpenVINS and VINS-Mono (with loop) fail on some sequences due to the significant drift in the middles of the track. VINS-RGBD (with loop) fails in all apartment sequences due to the large noise of depth on the ground. VIW-Fusion also struggles with some long corridor sequences. Although ORBSLAM3 (offline) fails in some long trajectories because of tracking lost in some visually challenging sequences (e.g., textureless region), it outperforms other methods in most sequences benefiting from the global map offline optimization and the loop detection module. OpenVINS-Depth improves robustness by adding depth information and achieves the best performance among online algorithms. Compared to sequences in small scenes, such as offices and apartments, all the methods show degraded performances in building floor sequences due to degradation in long straight line 2D motions. Moving along straight lines makes the systems fail to observe the correct scale information. The experimental results demonstrate that our dataset has several open challenges for such approaches.

## 6.2. 3D reconstruction evaluation

In our case, the accuracy is employed to quantify a reconstructed 3D model. For each point $p$ in the reconstruction, the Euclidean distance is computed after the closest point $p$* in the ground truth model is located between $p$ and $p$*. The accuracy measures the RMSE distance between the reconstructed 3D model and the ground truth model, which is defined as

$$Acc: = mean_{p\in P}\left(\min_{p^*\in P^*}\|p - p^*\|\right), \tag{5}$$

where $P$ and $P$* are the predicted and the ground truth point clouds, respectively.

Using ground truth trajectories and RGBD images as input, we have implemented several state-of-the-art 3D reconstruction methods on our dataset with Intel i9-13900KS CPU and NVIDIA GeForce RTX 4070 Ti. ElasticFusion (Whelan et al. (2015; 2016)) is a real-time dense visual SLAM system capable of capturing

**Table 5.** RMSE ATE (↓) in Meters of the Evaluated SLAM Methods.

| Method | SVO2<br>Online | VIW-Fusion<br>With loop | ORB-SLAM3<br>Offline | VINS-mono<br>With loop | VINS-RGBD<br>With loop | OpenVINS<br>Online | OpenVINS-<br>Depth<br>Online |
|---|---|---|---|---|---|---|---|
| Sensors | RGB + IMU | RGBD + IMU + Wheel | RGBD + IMU | RGB + IMU | RGBD + IMU | RGB + IMU | RGBD + IMU |
| Office_1 | 0.266 | 0.108 | **0.035** | 0.075 | 0.053 | 0.132 | 0.069 |
| Office_2 | 0.474 | 1.346 | **0.028** | 0.041 | 0.038 | 0.107 | 0.095 |
| Office_3 | 0.527 | 0.397 | **0.043** | 0.052 | 0.042 | 0.077 | 0.074 |
| Floor14_1 | 0.678 | - | **0.066** | 0.360 | 1.038 | 0.382 | 0.199 |
| Floor14_2 | 2.412 | - | **0.224** | 0.314 | 0.266 | 0.253 | 0.256 |
| Floor14_3 | - | 14.417 | 0.723 | 0.814 | 0.732 | 0.434 | **0.346** |
| 14-13-14 | - | 6.153 | **0.224** | 0.565 | 0.617 | 0.668 | 0.354 |
| 14-13-12 | - | 0.110 | 1.819 | 0.433 | 1.691 | **0.081** | 0.124 |
| Floor3_1 | - | - | **0.072** | - | 0.313 | 0.498 | 0.320 |
| Floor3_2 | - | 5.808 | 0.520 | - | 0.417 | 1.480 | **0.384** |
| Floor3_3 | - | - | **0.318** | 0.877 | 0.622 | 1.723 | 0.438 |
| Floor13_1 | 1.693 | 19.835 | **0.496** | 0.699 | 0.653 | 0.613 | 0.641 |
| Floor13_2 | 2.221 | 16.714 | **0.224** | 0.779 | 0.425 | 1.221 | 0.758 |
| Apartment1_1 | - | 0.166 | **0.033** | 0.111 | - | - | 0.131 |
| Apartment1_2 | - | 0.184 | - | - | - | 0.169 | **0.124** |
| Apartment1_3 | - | 0.619 | 0.574 | 0.186 | - | - | **0.135** |
| Apartment2_1 | - | 0.123 | - | 0.169 | - | 0.152 | **0.111** |
| Apartment2_2 | - | 0.121 | **0.106** | 0.111 | - | - | 0.145 |
| Apartment2_3 | - | 0.215 | **0.048** | 0.108 | - | - | 0.135 |
| Apartment3_1 | - | 0.131 | 1.547 | 0.157 | - | **0.087** | 0.116 |
| Apartment3_2 | - | 0.151 | **0.050** | 0.114 | - | 0.114 | 0.086 |
| Apartment3_3 | - | - | **0.099** | - | - | 0.140 | 0.108 |

Bold and underlined indicate the best and second-best performances among all methods on the same sequence. — Indicates that the method fails in the sequence when the estimated trajectory is incomplete (more than 100 frames lost after initialization) or drifts significantly (RMSE > 500 m).

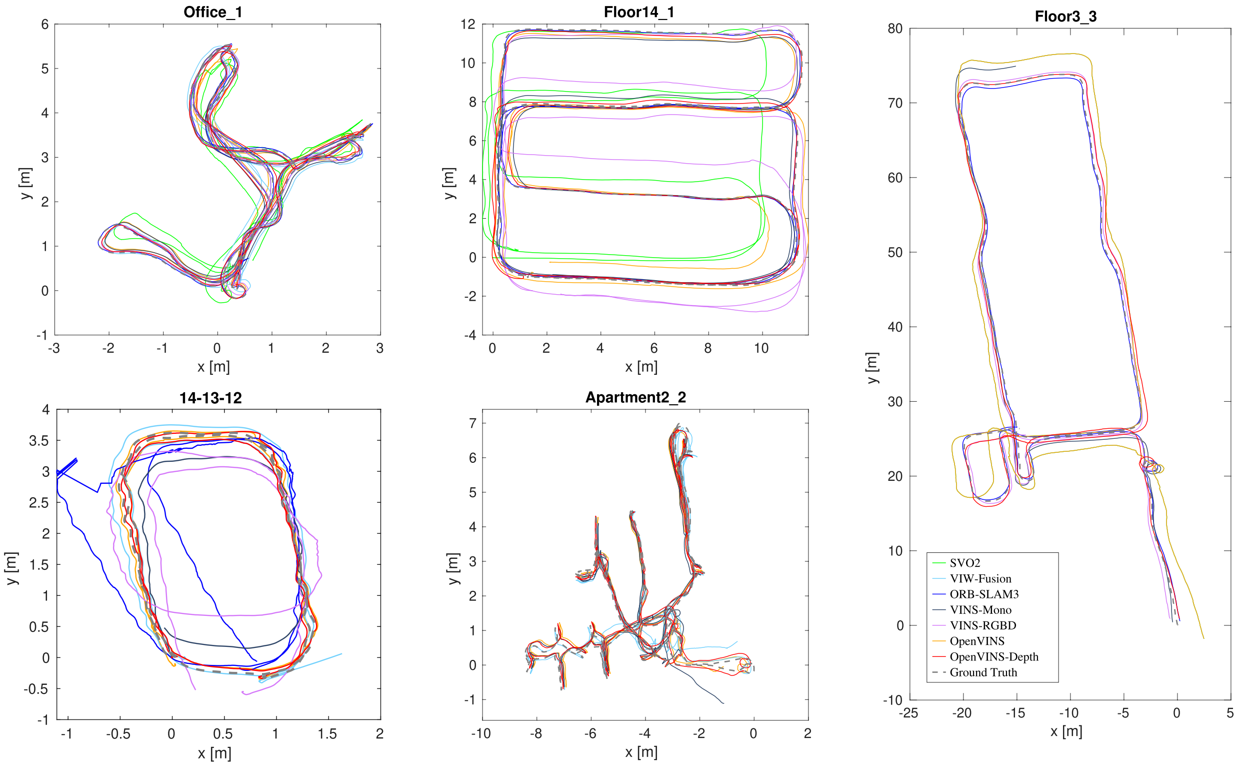


**Figure 11.** Exemplary trajectories estimated by the evaluated methods with the corresponding ground truth trajectories.

comprehensive dense globally consistent surfel-based maps. InfiniTAM (Kähler et al. (2016)) is suitable for large-scale 3D reconstruction with truncated signed distance field (TSDF) and surfel representation. BAD-SLAM (Schops et al. (2019)) exploits the joint optimization of the estimated 3D map and the camera trajectory for real-time dense RGBD SLAM based on surfel representation. Voxblox (Grinvald et al. (2019)) incrementally builds Euclidean Signed Distance Fields (ESDFs) out of TSDFs in dynamically growing maps, using different weight factors for the

**Table 6.** Accuracy (↓) in Centimeters of the Evaluated 3D Reconstruction Methods.

| Method | ElasticFusion | BAD-SLAM | InfiniTAM (TSDF) | InfiniTAM (Surfel) | Voxblox | Kimera (Semantic) |
|---|---|---|---|---|---|---|
| Processor | GPU | GPU | CPU | CPU | CPU | CPU |
| Office_1 | 2.97 | 2.35 | 3.85 | **2.12** | 4.69 | 4.54 |
| Office_2 | 3.03 | 2.76 | 5.20 | **2.47** | 6.77 | 7.08 |
| Office_3 | 2.58 | 2.69 | 4.70 | **2.30** | 6.23 | 5.66 |
| Floor14_1 | 4.99 | **3.84** | 4.93 | 4.85 | 5.42 | 7.02 |
| Floor14_2 | 5.96 | **4.41** | 5.51 | 4.63 | 6.03 | 4.77 |
| Floor14_3 | 5.16 | 4.35 | 5.02 | **4.06** | 5.35 | 6.18 |
| 14-13-14 | 4.05 | **3.78** | 8.94 | 3.80 | 6.11 | 5.86 |
| 14-13-12 | **1.93** | 3.10 | 5.56 | 2.19 | 3.76 | 2.62 |
| Floor3_1 | 15.37 | 12.11 | 9.36 | **6.94** | 9.46 | 9.22 |
| Floor3_2 | 10.03 | 9.16 | 6.60 | **6.16** | 6.70 | 6.59 |
| Floor3_3 | 8.91 | 5.92 | 5.63 | **3.98** | 5.96 | 5.35 |
| Floor13_1 | 3.84 | 4.28 | 4.75 | **3.65** | 5.06 | 4.00 |
| Floor13_2 | 3.58 | 3.43 | 3.90 | **3.35** | 4.27 | 4.96 |
| Apartment1_1 | **2.75** | 3.31 | 18.89 | 4.19 | 22.08 | 20.00 |
| Apartment1_2 | 3.03 | **2.25** | 5.67 | 2.80 | 5.70 | 6.03 |
| Apartment1_3 | 3.24 | **2.33** | 7.41 | 4.34 | 6.92 | 8.04 |
| Apartment2_1 | 2.10 | **1.70** | 2.70 | 2.20 | 2.83 | 14.16 |
| Apartment2_2 | 2.96 | **2.18** | 7.77 | 2.77 | 13.77 | 4.75 |
| Apartment2_3 | 2.10 | **2.02** | 12.18 | 2.32 | 18.76 | 18.31 |
| Apartment3_1 | 2.59 | **2.20** | 7.66 | 2.88 | 14.40 | 20.00 |
| Apartment3_2 | 2.31 | **2.23** | 13.92 | 2.43 | 12.35 | 14.99 |
| Apartment3_3 | 2.55 | **2.10** | 3.77 | 2.73 | 4.53 | 5.16 |

Bold and underlined indicate the best and second-best performances among all methods on the same sequence.

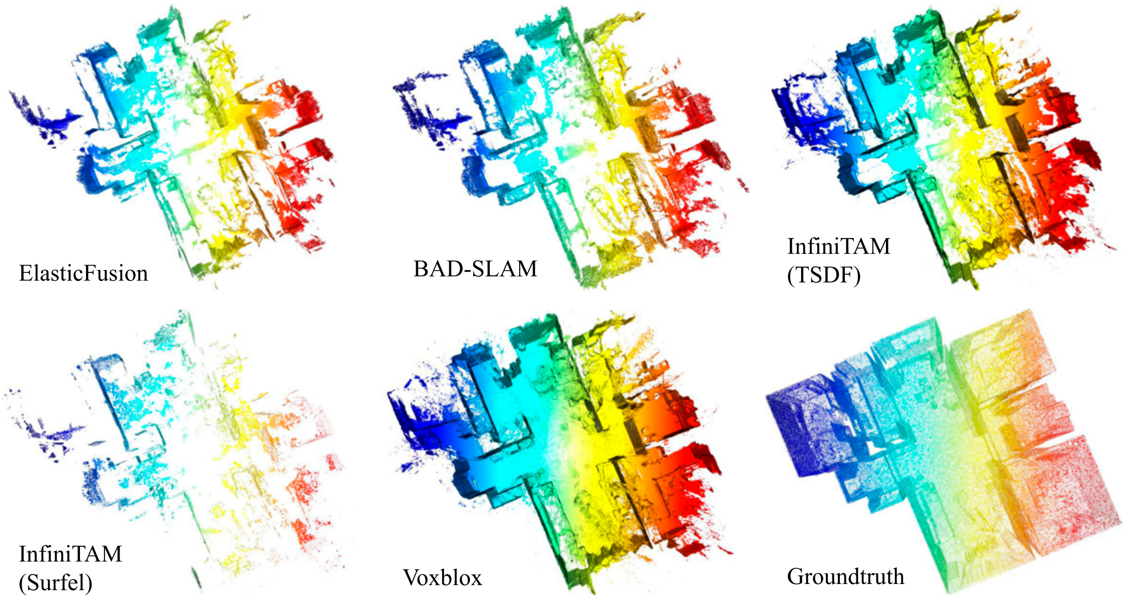


**Figure 12.** Reconstruction results on Apartment1. The three sequences are reconstructed separately and then combined in the global coordinate system.

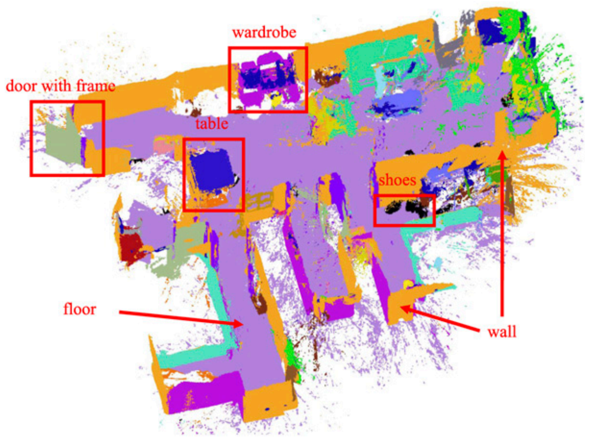


**Figure 13.** Reconstruction results from Kimera with semantic annotations on Apartment2.

surface sample points. Kimera (Rosinol et al. (2020)) is built based on Voxblox by adding 2D semantic information to generate a global semantically annotated 3D mesh.

We uniformly sample 200,000 points from the constructed model for evaluation, and the qualitative results are listed in Table 6. According to the results, surfel-based methods (BAD-SLAM, ElasticFusion and InfiniTAM (Surfel)) outperform others in most sequences. We can observe that InfiniTAM (TSDF), Voxblox, and Kimera (Semantic) exhibit significant drops in performance on low-view sequences. This could be attributed to the fact that low-viewpoint images contain more small structures (e.g., shoes and chair legs) and noise. Figure 12 shows the reconstruction results on Apartment1 as an example. At the cost of including large noise, Voxblox and InfiniTAM (TSDF) reconstruct more complete structures, while BAD-SLAM, ElasticFusion, and InfiniTAM (Surfel) obtain better reconstruction quality by removing outliers. BAD-SLAM achieves good compromise by combining surfel points and poses for global optimization. Figure 13 presents the semantically annotated reconstruction result from Kimera on Apartment2. Although the reconstruction is based on the high-precision poses, our dataset is challenging due to the noise in depth from a ground wheeled robot viewpoint in realistic environments, particularly in areas on the floor.

## Conclusion

This paper releases CID-SIMS, a challenging new dataset for indoor SLAM and 3D reconstruction researches. Compared to existing datasets, our dataset comprises sequences captured by a ground wheeled robot equipped with an RGBD-IW sensor suite, which includes an RGBD camera, an IMU, and a wheel odometer. To provide ground truth camera trajectories and 3D point clouds, we are the first to use an accurate 3D laser scanner called GeoSLAM. Furthermore, 2D semantic information is provided to support localization and reconstruction from semantically labeled images. Consequently, our dataset enables a thorough evaluation of both SLAM and 3D reconstruction algorithms that relies on visual, IMU, wheel odometer, and semantic information. All the sensors are well calibrated and synchronized so that no further alignment is necessary for evaluation with our benchmark. We conduct experiments on state-of-the-art approaches, demonstrating that our dataset is well suited for the evaluation of SLAM and 3D reconstruction using different types of inputs from a ground wheeled robot viewpoint. All the data is released to the public. We believe that our dataset will facilitate the development and benchmarking of new algorithms for indoor SLAM and 3D reconstruction under 3-DoF plane motions. In the future, we plan to assign 3D semantic labels to the point clouds for further use.

## Acknowledgements

This work was supported by the National Key R&D Program of China under Grant No. 2022YFB3303202 and National Natural Science Foundation of China under Grant Nos. 61836015 and 62202468.

## Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

## Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Key R&D Program of China under Grant No. 2022YFB3303202 and National Natural Science Foundation of China under Grant Nos. 61836015 and 62202468.

## ORCID iDs

Yidi Zhang https://orcid.org/0009-0002-2076-2531
Shuo Wang https://orcid.org/0000-0003-1269-7506

## Notes

1. Geoslam horizon scanner. Website: https://geoslam.com/solutions/zeb-horizon
2. ECOVACS. Website: https://www.ecovacs.com/global
3. DREAME. Website: https://global.dreametech.com/
4. KEENON. Website: https://www.keenonrobot.com/en/
5. Intel RealSenseTM Camera 400 Series Product Family Datasheet. Website: https://www.intelrealsense.com
6. WHEELTEC. Website: https://wheeltec.net/product/html/?173.html
7. GeoSLAM Accuracy Report. Website: https://geoslam.com/wp-content/uploads/2021/08/GeoSLAM_Accuracy_Report.pdf
8. CloudCompare. Website: https://cloudcompare.org

## References

Appleton E and Williams DJ (2012) *Industrial Robot Applications*. Berlin, Germany: Springer Science and Business Media.

Burri M, Nikolic J, Gohl P, et al. (2016) The euroc micro aerial vehicle datasets. *The International Journal of Robotics Research* 35(10): 1157–1163.

Campos C, Elvira R, Rodríguez JJG, et al. (2021) Orb-slam3: an accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Transactions on Robotics* 37(6): 1874–1890.

Chen L, Zhu Y, Papandreou G, et al. (2018) Encoder-decoder with atrous separable convolution for semantic image segmentation. CoRR Abs/1802.02611 URL. https://arxiv.org/abs/1802.02611

Chen D, Wang S, Xie W, et al. (2022) Vip-slam: an efficient tightly-coupled rgb-d visual inertial planar slam. In: *2022 International Conference on Robotics and Automation (ICRA)*. Philadelphia: IEEE, pp. 5615–5621.

Dai A, Chang AX, Savva M, et al. (2017) Scannet: richly-annotated 3d reconstructions of indoor scenes. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. San Juan, PR, USA: IEEE, pp. 5828–5839.

Davison AJ, Reid ID, Molton ND, et al. (2007) Monoslam: real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29(6): 1052–1067.

Dongfu Zhu ZC (2019) Calibrate the internel parameters and extrinsec parameters between camera and odometer. https://github.com/MegviiRobot/CamOdomCalibraTool

Fan Y, Zhang Q, Tang Y, et al. (2022) Blitz-slam: a semantic slam in dynamic environments. *Pattern Recognition* 121: 108225.

Fei L, Jaboyedoff M, Pullarello J, et al. (2019) Qualitative comparison of point clouds acquired by lidar, sfm, geoslam and sense 3d for the erosion quantification of a rock wall *Geophysical Research Abstracts* 21: 15938.

Forster C, Pizzoli M and Scaramuzza D (2014) Svo: fast semi-direct monocular visual odometry. In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*. Hong Kong, China: IEEE, pp. 15–22.

Forster C, Zhang Z, Gassner M, et al. (2017) SVO: semidirect visual odometry for monocular and multicamera systems. *IEEE Transactions on Robotics* 33(2): 249–265. DOI: 10.1109/TRO.2016.2623335.

Furgale P, Rehder J and Siegwart R (2013) Unified temporal and spatial calibration for multi-sensor systems. In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Tokyo, Japan: IEEE, pp. 1280–1286. DOI: 10.1109/IROS.2013.6696514.

Geiger A, Lenz P, Stiller C, et al. (2013) Vision meets robotics: the kitti dataset. *The International Journal of Robotics Research* 32(11): 1231–1237.

Geneva P, Eckenhoff K, Lee W, et al. (2020) Openvins: a research platform for visual-inertial estimation. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Paris, France: IEEE, pp. 4666–4672.

Glennie C and Lichti DD (2010) Static calibration and analysis of the velodyne hdl-64e s2 for high accuracy mobile scanning. *Remote Sensing* 2(6): 1610–1624.

Grinvald M, Furrer F, Novkovic T, et al. (2019) Volumetric instance-aware semantic mapping and 3D object discovery. *IEEE Robotics and Automation Letters* 4(3): 3037–3044. DOI: 10.1109/LRA.2019.2923960.

Grupp M (2017) Evo: python package for the evaluation of odometry and slam. https://github.com/MichaelGrupp/evo

Hägele M, Nilsson K, Pires JN, et al. (2016) *Industrial Robotics*. Berlin, Germany: Springer handbook of robotics, pp. 1385–1422.

Handa A, Whelan T, McDonald J, et al. (2014) A benchmark for rgb-d visual odometry, 3d reconstruction and slam. In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*. Hong Kong, China: IEEE, pp. 1524–1531.

Huang JK and Grizzle JW (2020) Improvements to target-based 3d lidar to camera calibration. *IEEE Access* 8: 134101–134110.

Huang JK, Feng C, Achar M, et al. (2020) Global unifying intrinsic calibration for spinning and solid-state lidars. *arXiv preprint arXiv:2012.03321*.

Jeong J, Cho Y and Kim A (2019) The road is enough! extrinsic calibration of non-overlapping stereo camera and lidar using road information. *IEEE Robotics and Automation Letters* 4(3): 2831–2838.

Jiang J, Xue P, Chen S, et al. (2018) Line feature based extrinsic calibration of lidar and camera. In: *2018 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*. Madrid, Spain: IEEE, pp. 1–6.

Kähler O, Prisacariu VA and Murray DW (2016) Real-time large-scale dense 3d reconstruction with loop closure. In: *Computer Vision - ECCV 2016 - 14th European Conference*. Amsterdam, The Netherlands: Springer Link, pp. 500–516.

Li J, Gao W, Wu Y, et al. (2022) High-quality indoor scene 3d reconstruction with rgb-d cameras: a brief review. *Computational Visual Media* 8(3): 369–393.

Lianos KN, Schonberger JL, Pollefeys M, et al. (2018) Vso: visual semantic odometry. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 234–250.

Liebowitz D and Zisserman A (1998) Metric rectification for perspective images of planes. In: *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No. 98CB36231)*. Santa Barbara, CA, USA: IEEE, pp. 482–488.

Liu J, Gao W and Hu Z (2019) Visual-inertial odometry tightly coupled with wheel encoder adopting robust initialization and online extrinsic calibration. In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Macau, China: IEEE, pp. 5391–5397.

Liu Y, Zhao C and Ren M (2022) An enhanced hybrid visual–inertial odometry system for indoor mobile robot. *Sensors* 22(8): 2930.

Moulon P, Monasse P, Perrot R, et al. (2016) OpenMVG: open multiple view geometry. In: *International Workshop on Reproducible Research in Pattern Recognition*. Berlin, Germany: Springer, pp. 60–74.

Mourikis AI and Roumeliotis SI (2007) A multi-state constraint kalman filter for vision-aided inertial navigation. In: *Proceedings 2007 IEEE International Conference on Robotics and Automation*. Rome, Italy: IEEE, pp. 3565–3572.

Mur-Artal R and Tardós JD (2017) Orb-slam2: an open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics* 33(5): 1255–1262.

Mur-Artal R, Montiel JMM and Tardos JD (2015) Orb-slam: a versatile and accurate monocular slam system. *IEEE Transactions on Robotics* 31(5): 1147–1163.

NathanPK Derek Hoiem and Fergus SilbermanR (2012) *Indoor Segmentation and Support Inference from Rgbd Images*. Dubai, United Arab Emirates: ECCV.

Oleynikova H, Lanegger C, Taylor Z, et al. (2020) An open-source system for vision-based micro-aerial vehicle mapping, planning, and flight in cluttered environments. *Journal of Field Robotics* 37(4): 642–666.

Pandey G, McBride J, Savarese S, et al. (2010) Extrinsic calibration of a 3d laser scanner and an omnidirectional camera. *IFAC Proceedings Volumes* 43(16): 336–341.

Qin T and Shen S (2018) Online temporal calibration for monocular visual-inertial systems. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Madrid, Spain: IEEE, pp. 3662–3669.

Qin T, Li P and Shen S (2018) Vins-mono: a robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics* 34(4): 1004–1020.

Qin T, Cao S, Pan J, et al. (2019a) A general optimization-based framework for global pose estimation with multiple sensors. arXiv:1901.03642

Qin T, Pan J, Cao S, et al. (2019b) A general optimization-based framework for local odometry estimation with multiple sensors. arXiv:1901.03638

Reina SC, Solin A, Rahtu E, et al. (2018) *ADVIO: An Authentic Dataset for Visual-Inertial Odometry. CoRR* abs/1807 09828 URL. https://arxiv.org/abs/1807.09828.

Romero-Ramirez FJ, Muñoz-Salinas R and Medina-Carnicer R (2018) Speeded up detection of squared fiducial markers. *Image and Vision Computing* 76: 38–47. DOI: 10.1016/j.imavis.2018.05.004. URL https://www.sciencedirect.com/science/article/pii/S0262885618300799

Rosinol A, Abate M, Chang Y, et al. (2020) Kimera: an open-source library for real-time metric-semantic localization and mapping. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Paris, France: IEEE, pp. 1689–1696.

Ruiz-Sarmiento JR, Galindo C and González-Jiménez J (2017) Robot@ home, a robotic dataset for semantic mapping of home environments. *The International Journal of Robotics Research* 36(2): 131–141.

Schmid L, Delmerico J, Schönberger J, et al. (2022) Panoptic multi-tsdfs: a flexible representation for online multi-resolution volumetric mapping and long-term dynamic scene consistency. In: *2022 IEEE International Conference on Robotics and Automation (ICRA)*. Philadelphia: ICRA, pp. 8018–8024. DOI: 10.1109/ICRA46639.2022.9811877.

Schonberger JL and Frahm JM (2016) Structure-from-motion revisited. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. San Juan, PR, USA: IEEE, pp. 4104–4113.

Schops T, Sattler T and Pollefeys M (2019) Bad slam: bundle adjusted direct rgb-d slam. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. New York, NY, USA: IEEE, pp. 134–144.

Schubert D, Goll T, Demmel N, et al. (2018) The tum vi benchmark for evaluating visual-inertial odometry. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Madrid, Spain: IEEE, pp. 1680–1687.

Shan Z, Li R and Schwertfeger S (2019) Rgbd-inertial trajectory estimation and mapping for ground robots. *Sensors* 19(10): 2251.

Shi X, Li D, Zhao P, et al. (2020) Are we ready for service robots? the openloris-scene datasets for lifelong slam. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Paris, France: IEEE, pp. 3139–3145.

Shi C, Tang F, Wu Y, et al. (2023) *Accurate Implicit Neural Mapping With More Compact Representation in Large-Scale Scenes Using Ranging Data*. Piscataway, NY, USA: IEEE Robotics and Automation Letters.

Straub J, Whelan T, Ma L, et al. (2019) The replica dataset: a digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*.

Sturm J, Engelhard N, Endres F, et al. (2012) A benchmark for the evaluation of rgb-d slam systems. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Vilamoura-Algarve, Portugal: IEEE, pp. 573–580.

Tsai RY and Lenz RK (1989) A new technique for fully autonomous and efficient 3 d robotics hand/eye calibration. *IEEE Transactions on Robotics and Automation* 5(3): 345–358.

Usenko V, Demmel N, Schubert D, et al. (2020) Visual-inertial mapping with non-linear factor recovery. *IEEE Robotics and Automation Letters (RA-L) and Int. Conference on Intelligent Robotics and Automation (ICRA)* 5(2): 422–429. DOI: 10.1109/LRA.2019.2961227.

Wasenmüller O, Meyer M and Stricker D (2016) Corbs: comprehensive rgb-d benchmark for slam using kinect v2. In: *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*. Lake Placid, NY, USA: IEEE, pp. 1–7.

Wei H, Tang F, Xu Z, et al. (2021a) A point-line vio system with novel feature hybrids and with novel line predicting-matching. *IEEE Robotics and Automation Letters* 6(4): 8681–8688.

Wei H, Tang F, Zhang C, et al. (2021b) Highly efficient line segment tracking with an imu-klt prediction and a convex geometric distance minimization. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. Xi'an, China: IEEE, pp. 3999–4005.

Wei H, Tang F, Xu Z, et al. (2022) Structural regularity aided visual-inertial odometry with novel coordinate alignment and line triangulation. *IEEE Robotics and Automation Letters* 7(4): 10613–10620.

Whelan T, Leutenegger S, Salas-Moreno R, et al. (2015) Elasticfusion: dense slam without a pose graph. In: *Robotics: Science and Systems*. Rome, Italy: Schloss Dagstuhl.

Whelan T, Salas-Moreno RF, Glocker B, et al. (2016) Elasticfusion: real-time dense slam and light source estimation. *The International Journal of Robotics Research* 35(14): 1697–1716.

Wu KJ, Guo CX, Georgiou G, et al. (2017) Vins on wheels. In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. Singapore: IEEE, pp. 5155–5162.

Wu Y, Tang F and Li H (2018) Image-based camera localization: an overview. *Visual Computing for Industry, Biomedicine, and Art* 1(1): 1–13.

Xu Z, Wei H, Tang F, et al. (2023) Plpl-vio: a novel probabilistic line measurement model for point-line- based visual-inertial odometry. In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Detroit, Michigan, USA: IEEE.

Yang L, Zhuo W, Qi L, et al. (2022) St++: make self-training work better for semi-supervised semantic segmentation.

Yin J, Li A, Li T, et al. (2021) M2dgr: a multi-sensor and multi-scenario slam dataset for ground robots. *IEEE Robotics and Automation Letters* 7(2): 2266–2273.

Zhang Z (2000) A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(11): 1330–1334.

Zhang M, Chen Y and Li M (2019) Vision-aided localization for ground robots. In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Macau, China: IEEE, pp. 2455–2461.

Zhang H, Jin L and Ye C (2020) The vcu-rvi benchmark: evaluating visual inertial odometry for indoor navigation applications with an rgb-d camera. In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Las Vegas, NV, USA: IEEE, pp. 6209–6214.

Zhang L, Helmberger M, Fu LFT, et al. (2023) Hilti-oxford dataset: a millimeter-accurate benchmark for simultaneous localization and mapping. *IEEE Robotics and Automation Letters* 8(1): 408–415. DOI: 10.1109/LRA.2022.3226077.

Zhou D, Dai Y and Li H (2019) Ground-plane-based absolute scale estimation for monocular visual odometry. *IEEE Transactions on Intelligent Transportation Systems* 21(2): 791–802.

Zhuang T (2021) Viw-fusion: visual-inertial-wheel fusion odometry. https://github.com/TouchDeeper/VIW-Fusion.

Zimmerman N, Guadagnino T, Chen X, et al. (2022) Long-term localization using semantic cues in floor plan maps. *IEEE Robotics and Automation Letters* 8(1): 176–183.

Zimmerman N, Sodano M, Marks E, et al. (2023) Constructing metric-semantic maps using floor plan priors for long-term indoor localization. In: *IEEE/RSJ Intl. Conf. On Intelligent Robots and Systems (IROS)*. Detroit, MI, USA: IEEE.