ISPRS BENCHMARK on MULTISENSORY INDOOR MAPPING and POSITIONING

ISPRS BENCHMARK ON MULTISENSORY INDOOR MAPPING AND POSITIONING

Cheng Wang[1 *], Yudi Dai[1], Naser Elsheimy[2], Chenglu Wen[1], Guenther Retscher[3], Zhizhong Kang[4], and Andrea Lingua[5]

[1] Fujian Key Laboratory of Sensing and Computing, School of Informatics, Xiamen University, 422 Siming Road South, Xiamen 361005, China - (cwang@xmu.edu.cn, daiyudi@stu.xmu.edu.cn, clwen@xmu.edu.cn)
[2] University of Calgary, Canada - elsheimy@ucalgary.ca
[3] TU Wien - Vienna University of Technology, Austria - guenther.retscher@geo.tuwien.ac.at
[4] China University of Geosciences, Beijing - zzkang@cugb.edu.cn
[5] Polytechnic University of Turin, Italy - andrea.lingua@polito.it

KEY WORDS: Multi-sensor, Indoor, Benchmark Dataset, SLAM, BIM, Indoor Positioning

ABSTRACT:
In this paper, we present a publicly available benchmark dataset on multisensorial indoor mapping and positioning (MiMAP), which is sponsored by ISPRS scientific initiatives. The benchmark dataset includes point clouds captured by an indoor mobile laser scanning system in indoor environments of various complexity. The benchmark aims to stimulate and promote research in the following three fields: (1) LiDAR-based Simultaneous Localization and Mapping (SLAM); (2) automated Building Information Model (BIM) feature extraction; and (3) multisensory indoor positioning. The MiMAP project provides a common framework for the evaluation and comparison of LiDAR-based SLAM, BIM feature extraction, and smartphone-based indoor positioning methods. This paper describes the multisensory setup, data acquisition process, data description, challenges, and evaluation metrics included in the MiMAP project.

1. INTRODUCTION

Indoor environments such as office, classroom, shopping mall, and parking lots are essential to our daily life. Three-dimensional (3D) mapping and positioning technologies for indoor environments have become in high demand in recent years. Online visualization, location-based services (LBS), indoor navigation, elder assistance, and emergency evacuation are just a few examples of the emerging applications that require 3D mapping and positioning of indoor environments. SLAM-based indoor mobile laser scanning systems (IMLS) (Wen et al., 2016) provide a useful tool for indoor applications. During the IMLS procedure, 3D point clouds and high accuracy trajectories with position and orientation are acquired. Many efforts have been made in the last few years to improve the SLAM algorithms (Zhang et al., 2014) and the geometric/semantic information extraction from point clouds and images (Armeni et al., 2016) (Wang et al.,2018). However, both significant opportunities and severe challenges exist in the multisensory data processing of IMLS. First, lack of efficient or real-time 3D point cloud generation methods of as-built 3D indoor environment; second, face difficulties of building information model (BIM) features extraction in the clustered and occluded indoor environment. Also, given the relatively high accuracy, the IMLS trajectory provides a perfect reference for the low-cost indoor positioning solutions. Standard datasets are critical for the research on these topics.

Under the sponsorship of ISPRS Scientific Initiatives 2019, we developed the ISPRS Benchmark on Multisensorial Indoor Mapping and Positioning (MiMAP). MiMAP aims to promote researches in three aspects: (1) LiDAR-based SLAM; (2) automated BIM feature extraction from point clouds, focusing on extraction of building elements, such as floors, walls, ceilings, doors, windows that are important in building management and navigation tasks; and (3) multisensory indoor positioning, focusing on the smartphone platform solution. MiMAP also provides evaluation methods for these three aspects. MiMAP Dataset is open-access via the ISPRS WG I/6 official Website (ISPRS WG I/6, 2020) or the mirror website http://mi3dmap.net/. The rest of this paper describes the multisensory setup, data acquisition, dataset description, challenges and evaluation metrics in the MiMAP project.

2. DATASET

MiMAP project team upgraded the XBeibao system (Wen et al., 2016), a multi-sensory backpack system developed by Xiamen University to build the MiMAP benchmark. The upgraded system (Figure 1. (a)) can synchronously collect data with multi-beam laser scanners, fisheye cameras, and readings from smartphones built-in sensors, such as barometer, magnetometer, MEMS IMU and WiFi. The baseline SLAM 3D point clouds of the indoor test environments were also provided based on the XBeibao processing software. We used Riegl VZ 1000 (Figure 1. (b)) to collect high accuracy point cloud as the ground-truth of indoor mapping.

2.1 Multisensory setup

The involved sensors are listed as follows:

XBeibao system

- 1× Velodyne VLP-Ultra Puck™ rotating 3D laser scanner. 20Hz, 32 beams, 4cm accuracy, collecting 0.6 million points/second, 200m range, with a field of view of 360° horizontal × 40° vertical.
- 1×Velodyne VLP-16L rotating 3D laser scanner. 20Hz, 16 beams, 0.1° ~ 0.4° horizontal angle resolution, 3cm accuracy, collecting 0.3 million points/second, 100m range, with a field of view of 360° horizontal× ±15° vertical.
- 4×Fisheye Camera. 1280*720 @ 30fps video resolution, with a field of view of 4×175°.

Smartphones

- Sensors: gyroscope, accelerometer, barometer, electronic compass, Wi-Fi, magnetometer, GNSS(GPS).

Riegl VZ 1000 laser scanner

- Range from 1.5m up to 1200m, 5mm precision, 8mm accuracy, collecting 0.3 million points/second, with a field of view of 100° vertical ×360° horizontal.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-V-5-2020-117-2020 | © Authors 2020. CC BY 4.0 License.

(a)

(b)

Figure 1. Data acquisition devices.
(a) XBeibao. (b) Riegl VZ-1000.

When collecting the data, we placed one smartphone facing up on the top of the upper LiDAR sensor, the others are held in hand. A laptop is used to control the data collection of cameras and LiDAR sensors. Also, it is used as a hotspot to connect with the smartphone to synchronize the sensors and used to store the incoming LiDAR data streams. A system operator needs to carry the laptop during the collection process. All the collected data will be transferred to the laptop through wire.

2.2 Dataset

2.2.1 Dataset overview:

The MiMAP benchmark includes three datasets:

Indoor LiDAR-based SLAM dataset

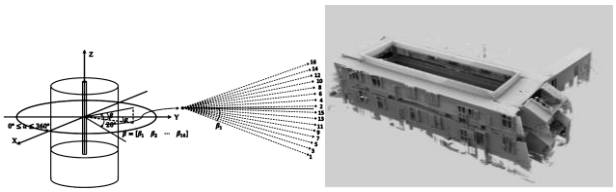
We collected indoor point clouds dataset in three multi-floor buildings with the upgraded XBeibao. This dataset represents the typical indoor building complexity. We provide raw data of one indoor scene with ground truth for users' own evaluation. We also provide raw data of two scenes for evaluation by submitting their results to us. The evaluation criteria encompass the error to the ground truth point cloud acquired with a millimeter-level accuracy terrestrial laser scanner (TLS) (Figure 2(b)).

BIM feature extraction dataset

We provide three data with ground truth for evaluating the BIM feature extraction on indoor 3D point clouds. Ground truth data was manually built, and the examples are presented in Figure 3.

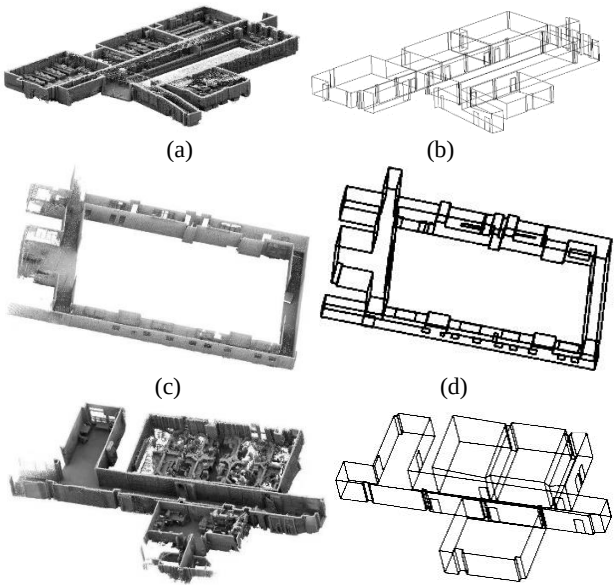
Indoor positioning dataset

We provide two data sequences with ground truth and provide three data sequences without ground truth for evaluation by submitting results. The evaluation criteria encompass the error to the centimeter-level accuracy platform trajectory from the SLAM processing (Figure 4).



(a) (b)

Figure 2. Illustration of indoor LiDAR-based indoor point cloud. (a) multi-beam laser scanning sensor model. (b) the high-accuracy reference TLS point cloud.



(a) (b) (c) (d) (e) (f)

Figure 3. Illustration of BIM feature extraction dataset.
(a,c,e) Point clouds. (b,d,f) Corresponding BIM frame features.

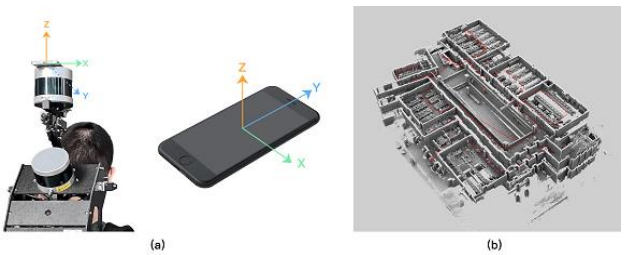


Figure 4. Illustration of the indoor positioning dataset.
(a)Setup of the smartphone with XBeibao. (b)SLAM trajectory as synchronized reference for indoor positioning.

2.2.2 Dataset description:

A sequence of data is compressed into a file with the name format mimap_type_number.zip, where type represents one of the three datasets, and the number indicates the serial number of this type's recording round. The "type" has three values——in_slam, bim and in_pose, representing the indoor LiDAR-based SLAM dataset, the BIM feature extraction dataset, and the indoor positioning dataset, respectively. The dataset's directory structure and detailed description are shown below.

The **indoor LiDAR-based SLAM dataset** consists of three scenes captured by multi-beam laser scanners in indoor environments with various complexity. The original scan frame data from scanners are provided and saved in pcap file. The timestamp of every point from the LiDAR sensor is given in the pcap file.

The *mimap_in_slam_00.zip* and the *mimap_in_slam_01.zip* are acquired by a Velodyne Ultra-pack™, while *mimap_in_slam_02.zip* is acquired by a Velodyne HDL-32e. Only the *mimap_in_slam_00.zip* dataset provides the ground truth point cloud data, which acquired by a Riegl VZ 1000.

We provide the raw videos captured by the four cameras in *mimap_in_slam_02.zip*. The videos are names as *position.avi*, where the *position* is the placeholder of the front, the rear, the left, or the right camera. The time of every frame is saved in *video_frame_timet.txt*. Each line of the file is a relative timestamp(us) to the system boot time, and the line number represents the frame number of the video. The four videos have the same timestamp.

If video data are provided, each camera's intrinsic matrix, extrinsic matrix and distortion coefficients will be saved in *parameter.xml***.** There are four cameras, front, rear, left and right, which respectively refer to the direction of the camera and their positions on the XBeibao system. The extrinsic matrix is used to convert the camera's coordinate system to LiDAR A's coordinate system.

If original pcap files of two Velodyne sensors are provided, the 4×4 calibration matrix converting the LiDAR B's coordinate system to LiDAR A's coordinate system will be saved in *parameter.xml*.

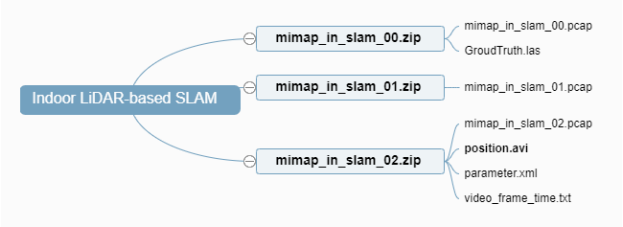


Figure 5. Structure of the indoor LiDAR-based SLAM dataset.

Table 1. Data description of the indoor LiDAR-based SLAM dataset.

Data	File name	Description
00	*mimap_in_slam_00.pcap*	Raw data of a two-floor building scene. Scanned by a Velodyne Ultra pack.
	GroundTruth.las	Scanned by the Riegl VZ 1000.
01	*mimap_in_slam_01.pcap*	Raw data of a five-floor building scene. Scanned by a Velodyne Ultra pack.
02	*mimap_in_slam_02.pcap*	A five-floor building scene. Scanned by a Velodyne HDL-32E.
	front / rear / left / right.avi	Four video files (25fps).
	parameter.xml	Cameras' intrinsic, extrinsic and distortion coefficients parameters
	Video_frame_time.txt	Timestamp(us) of every video frame. The line number is equal to the frame number.

The **BIM feature extraction dataset** contains data from three indoor scenes with various complexity. For each scene, raw data (point cloud in LAS format) and corresponding BIM line framework (in OBJ format) are provided. Users can evaluate their methods using the downloaded reference line frameworks.

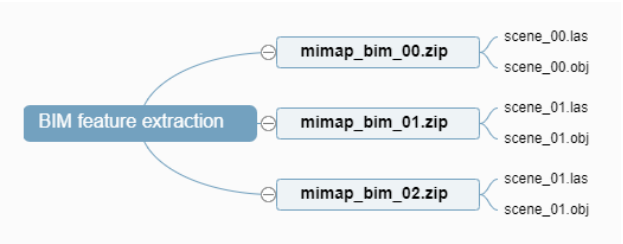


Figure 6. Structure of the BIM feature extraction dataset.

Table 2. Data description of the BIM feature extraction dataset.

Data	File name	Description
00	*Scene_00.las*	A closed-loop corridor scene.
	Scene_00.obj	The line framework of the point cloud scene.
01	*Scene_01.las*	A corridor and multiple rooms scene.
	Scene_01.obj	The line framework of the point cloud scene.
02	*Scene_02.las*	A closed-loop corridor and multiple rooms scene.
	Scene_02.obj	The line framework of the point cloud scene.

The **indoor positioning dataset** consists of five data sequences acquired in indoor environments with various complexity. Data sequences of sensor records from smartphones are provided. Users can test their positioning algorithm on these data. The first two sequences (*mimap_in_pose_00* and *mimap_in_pose 01*) were acquired in one building, and the other three sequences (*mimap_in_pose 02*, *mimap_in_pose 03*, *mimap_in_pose 04*) were acquired in another building. Only *mimap_in_pose_00* and *mimap_in_pose_02* contains ground truth trajectory file(in TXT format). The trajectory is the SLAM result of the LiDAR, containing the position, rotation and timestamp(*us*) of every frame. The detailed format is listed in the file.

Each data sequence contains a *phones* directory folder and *phones_data_description.txt* file. The *phones* folder is the placeholders of the smartphone's name, and usually, there are mutiple *phones* directory folders. In every smartphone's folder, there are *timeOffset.txt* and many *sensor_name.txt* files. *sensor_name* represents the smartphone's sensor abbreviation name, including gyroscope, accelerometer, barometer, electronic compass, Wi-Fi sensor, magnetometer, GPS, *etc.* The *timeOffset.txt* records the time offsets between the phone and the

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-V-5-2020-117-2020 | © Authors 2020. CC BY 4.0 License.

NTP server. The *phones_data_description.txt* details the format of each file in *phones* directory.
The accuracy of the distance between the smartphones and the LiDAR is sufficient for indoor positioning tasks, so we did not provide the calibration files between smartphones and the LiDARs.

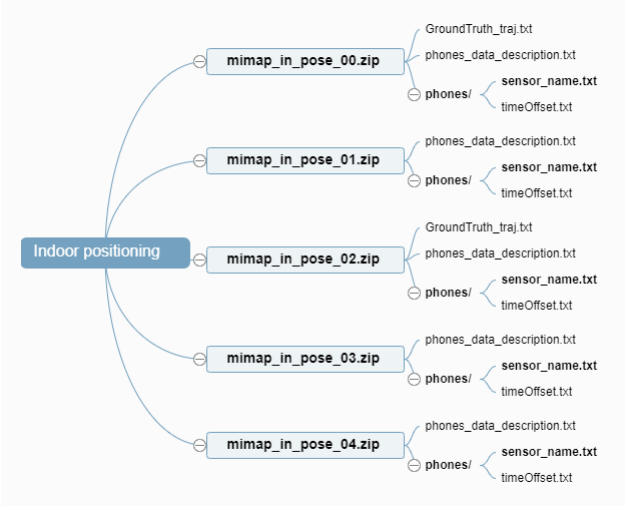


Figure 7. Structure of the indoor positioning dataset.

Table 3. Data description of the indoor positioning dataset.

Data	File name	Description
00	*GroundTruth_traj.txt*	LiDAR's trajectory data containing the positon, rotation and timestamp.
	phones_data_description.txt	A five-floor building scene including data of individual rooms, closed-loop corridors and stairs.
	XIAOMI5S / XIAOMI6	Two directorys of the smartphones data files.
01	*phones_data_description.txt*	A three-floor building scene including data of individual rooms, closed-loop corridors and stairs.
	XIAOMI5S / XIAOMI6	Two directorys of the smartphones data file.
02	*GroundTruth_traj.txt*	LiDAR's trajectory data containing the positon, rotation and timestamp.
	phones_data_description.txt	A six-floor building scene including data of corridors and stairs
	HuaweiP8lite / MI6	Two directorys of the smartphones data files.
03	*phones_data_description.txt*	A single-floor building scene including data of multiple rooms
	MI6 / ALE-L21	Two directorys of the smartphones data file.
04	*phones_data_description.txt*	A single-floor building scene including data of multiple rooms.
	MI5S	The directory of the smartphone data file.

3. CHALLENGES

3.1 Time synchronization

In order to synchronize the smartphone and LiDARs, a laptop is set as a local NTP (Network Time Protocol) server, then the phones are connected to it to synchronize their time. The LiDAR is connected to the laptop through a network cable. The timestamp of every point cloud frame is a relative time to the start recording time. We can view the start Unix-timestamp on the laptop and then add it to all frames' timestamps, the point clouds' timestamp is therefore connected to the NTP server. Thus, the smartphone and LiDAR can synchronize their time now through the laptop as a bridge.
Smartphone's time can synchronize to the local NTP server during the recording, so the Unix-timestamp in every piece of data is relatively accurate. Due to the instability of the Wi-Fi connection, there are time offsets between the smartphones and the NTP server, which range from 20ms to 500ms. We record them before recording the data.
Since all data's timestamps are acquired, we can obtain the position at any time by interpolation and can use the LiDAR's positioning result as the smartphone' positioning ground-truth.

3.2 Multi-Sensors Calibration

In this system, LiDAR sensor A (X_{l1}, Y_{l1}, Z_{l1}) is mounted horizontally; LiDAR sensor B (X_{l2}, Y_{l2}, Z_{l2}) is mounted 45°below the LiDAR sensor A (Figure 1 (b)). Based on our previous work (Gong et al., 2018), point cloud data of LiDAR sensor A, (P_A), and point cloud data of LiDAR sensor B, (P_B), are fused into P_f by the 4×4 transform matrix between the two LiDAR sensors (T_{cal}). (Eq. (1)). Additionally, Terrestrial Laser Scanning (TLS) data is introduced to bridge the calibration between LiDAR sensors and cameras. The calibration process is shown in Figure 8.

$$P_f = P_A + T_{cal} * P_B \quad (1)$$

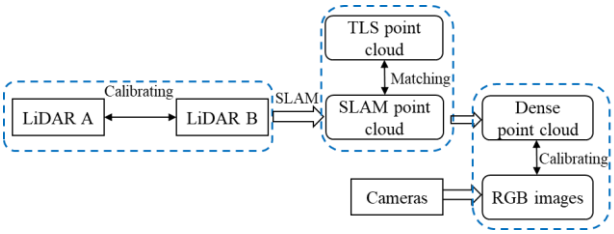


Figure 8. Flowchart of the calibration process (Wen et al., 2019).

3.2.1 LiDAR-to-LiDAR calibration: The calibration of the multi-LIDAR sensor is calculated recursively in the construction of the sub-map and its isomorphism constraint (Gong et al., 2018). Assuming T_A^n is the trajectory of LIDAR sensor A at a time (*0~n*) in the mapping algorithm, P_B^n is the point cloud of LIDAR sensor B at time *n*. T_i is the initial coordinate system transformation between the LIDAR sensors. Calibration is the calculation of the exact calibration matrix T_{cal} by:

$$P_{near}^{n} = NN(M, T_A^n, P_B^n, T_i) \quad (2)$$

$$T_{cal} = arg\min_{T_{cal}} \sum_n \left\| P_B^n * T_{cal} - P_{near}^n \right\|^2 \quad (3)$$

where $NN(\cdot)$ is the nearest neighbour point search algorithm. Using T_A^n and T_i, P_B^n is first transformed to its location at time n in the sub-map M. Then the $NN(\cdot)$ algorithm is used to search the sub-map for the nearest neighbour point set, P_{near}^n. Lastly, an environmental consistency constraint is introduced to obtain T_{cal}.

3.2.2 Camera -to-LiDAR calibration: The camera intrinsic calibration matrix is given by $\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$ and (k_1, k_2, k_3), where (f_x, f_y) is the focal length of the camera, (c_x, c_y) is the position of the camera and (k_1, k_2, k_3) is the factors of radial distortion. Scaramuzza's camera calibration method (Scaramuzza et al., 2006) is used to determine the internal parameters and distortion factors of the camera.

We utilized a TLS (e.g., Riegl VZ 1000) to bridge the calibration between LiDAR sensors and cameras. By manually selected matching points between them, we can acquire the camera's extrinsic transformation $[R, T]$, where R is the 3×3 rotation matrix, and T is the 1×3 translation vector.

Phone-to-LiDAR calibration: We placed the smartphone face up on the LiDAR A (Figure 9), and making the Y-axis parallel to the laser beam scanning direction. Thus, the phone's coordinate system and the LiDAR's coordinate system have the same XYZ-axis direction. We carried more than one smartphone in some scenes, except the one on LiDAR A, other smartphones are held in hand. We did not provide the calibration files, because the accuracy of the distance is sufficient for indoor positioning tasks.

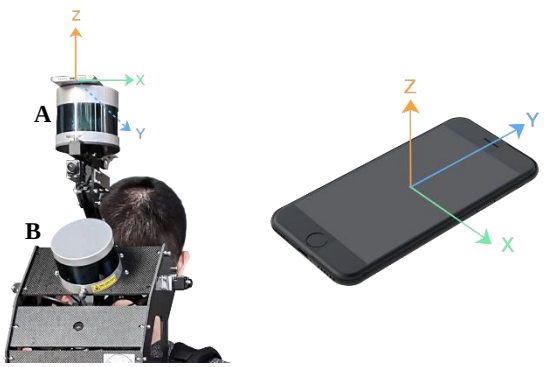


Figure 9. The smart phone's position and coordinate.

3.3 Reference data generation

For benchmark evaluation, we generated reference data from a subset of the raw data and introduced other high accuracy data.

3.3.1 SLAM-based indoor point cloud

The reference data of SLAM-based indoor point cloud is collected by a millimeter-level accuracy terrestrial laser scanner (TLS) (Figure.10). Before scanning, many high-reflection rectangle markers were placed on the wall and ground. Then several sub-maps were generated by scanning the scene in different positions, and overlap was guaranteed between adjacent sub-maps. Finally, these sub-maps were manually registered by picking the same marker and other feature points via RiSCAN PRO.

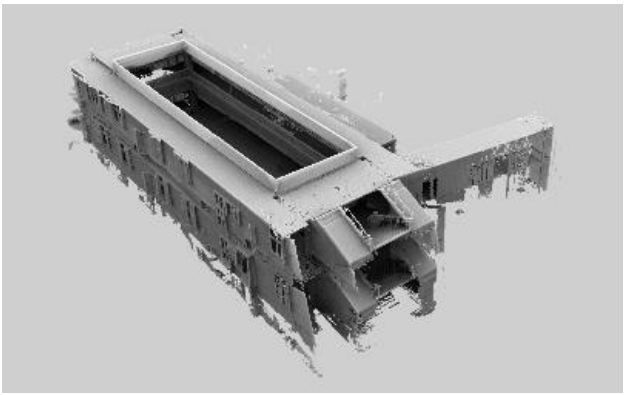


Figure 10. The reference data of SLAM-based indoor point cloud.

3.3.2 BIM feature

We used the building line framework developed by Wang (Wang et al., 2018) and the semantic objects labeled via manually editing. We selected the building lines with their length greater than 0.1 m in structured indoor building and saved their own two endpoints' coordinates. Fig.11 gives an example of BIM features. According to Wang's method, semantically labels the raw point clouds into the walls, ceiling, floor, and other objects firstly. And then, line structures are extracted from the labeled points to achieve an initial description of the building line framework. To optimize the detected line structures caused by occlusion, a conditional Generative Adversarial Nets (cGAN) deep learning model is constructed. The line framework optimization model includes structure completion, extrusion removal, and regularization. Finally, CloudCompare (Girardeau-Montaut, 2011) is used to fine-tune the line framework according to the raw point clouds with human intervention.

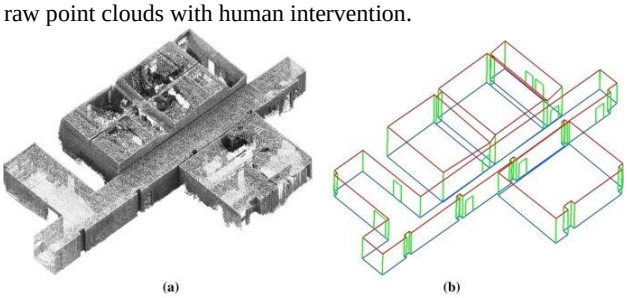


Figure 11. BIM feature examples. (a) Point cloud data. (b) BIM structure model. The green lines are doors and pillars. The red lines are ceilings. The blue lines represent the ground.

3.3.3 Indoor positioning

Firstly, we started to collect the smartphone sensors' data and the LiDAR's data at the same time. Then we applied the SLAM method (Wen et al., 2019) on the LiDAR's pcap data to generate a trajectory file with timestamps. The process of time synchronization was done according to subsection 3.1. The LiDAR's trajectory file is treated as the reference data of indoor positioning. An indoor positioning reference example is shown in Figure 12. The red line is the reference trajectory from SLAM process.

4. EVALUATION METRICS

4.1 SLAM-based indoor point cloud

Kümmerle (K¨ummerle et al., 2009) proposed a metric for measuring the performance of a SLAM algorithm by considering the poses of a robot during data acquisition. However, for indoor environments, it is hard to get the reference of the trajectory poses. We follow the metric for point cloud comparison proposed by

Lehtola (Lehtola, V. V. et al. 2017). To be specific, our evaluation firstly reconstructs the point cloud based on the submitted trajectory. Then, voxel filtering of 3cm is performed to ensure the same resolution of the point cloud. The error of a single point is given by the weighted point-to-point (p2p) absolute distance:

$$\varepsilon(p_i) = w_i \cdot (p_i \ominus q_i) \tag{4}$$

where p_i is a point in the evaluated point cloud, q_i is the corresponding nearest neighbor point in the reference point cloud, $\ominus$ means the Euclidean distance between two points. w_i is calculated as:

$$w_i = \begin{cases} 1, & p_i \ominus q_i < D \\ 0, & others \end{cases} \tag{5}$$

and the error of the whole point cloud is calculated by the mean and stand deviation of each point:

$$\bar{\varepsilon} = \frac{1}{N}\sum_i \varepsilon(\mathrm{p}_i) \tag{6}$$

$$s = \frac{1}{N}\sum_i ||\varepsilon(p_i) \ominus \bar{\varepsilon}||_2 \tag{7}$$

where N is the number of points in the evaluated point cloud which satisfy $w_i = 1$.
The motivation for using absolute distance is that it can be calculated by searching the nearest neighbor instead of manually selecting the corresponding feature points between the two point cloud maps, which will introduces manual errors and unfairness. Since the nearest neighbor search is used for points association, the coordinate system of the point cloud to be evaluated should be the same with the reference one. The point cloud generated by the SLAM algorithm uses the local coordinate system of the first frame as the global coordinate system. To make a fair comparison, we manually registered the first frame of the SLAM point cloud to the reference point cloud to obtain a transformation matrix *T*. By subsequently applied *T* to each evaluated point cloud, this point is aligned to the reference point cloud. The evaluation table will rank methods according to the average of absolute errors.

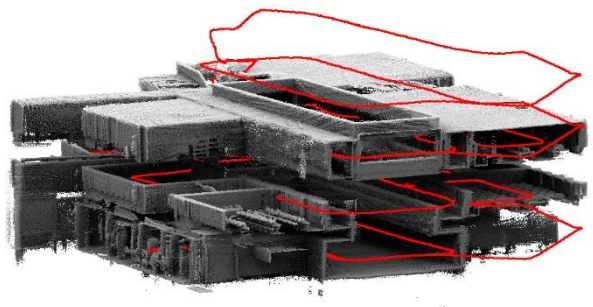


Figure 12. Indoor positioning reference example. The red line is the reference trajectory from SLAM process. The point cloud map generated based on the red trajectory is a five-floors building. (Only part of the building is shown).

4.2 BIM feature

The BIM feature extraction dataset contains data from three indoor scenes with various complexity. For each of the scenes, raw data (point cloud in LAS format) and corresponding BIM line framework (in OBJ format) are provided.
Imitating COCO evaluation criterion (Lin et al., 2014), we adopt the average precision (AP) of the predicted line framework as the primary metric. We use threshold θ to decide whether two lines are coincident, instead of Intersection over Union (IoU) used in COCO.
Given a line $l_t = \overrightarrow{ab}$ in ground truth annotations and a line $l_p = \overrightarrow{a'b'}$ in prediction, if the mean value D of the distance between two pairs of endpoints is less than the threshold θ, the two lines are considered to be coincident.

$$D = \left(\left\|a - a'\right\|_2 + \left\|b - b'\right\|_2\right)/2 \tag{8}$$

Figure 13 shows one example: because the distance between $\overrightarrow{ab}$ and $\overrightarrow{a'b''}$ is $D = 0.3 < 0.5$ and the distance between $\overrightarrow{ab}$ and $\overrightarrow{a''b''}$ is $D = 0.55 > 0.5$, $\overrightarrow{a'b'}$ is considered as true positive while $\overrightarrow{a''b''}$ is considered as false positive.

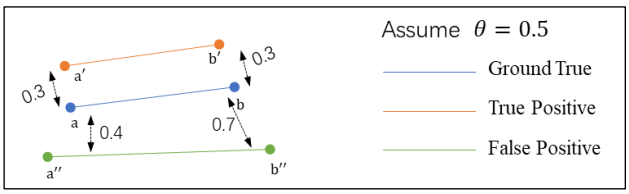


Figure 13. An example of evaluating BIM feature lines. The dots and the lines in the figure represent the vertexes and the lines of the BIM feature. The number is the distance(m) between the line's vertex.

AP is defined as the area under the precision-recall curve, and AP is averaged over multiple threshold θ. Specifically, we set ten thresholds from 1.4cm to 0.5cm at step 0.1.
The proposed metric computes the spatial consistency of the predicted and ground truth line frameworks. If the algorithm fails to the endpoints or capture the correct line direction, the number of true positive will be limited under strict threshold θ and the AP will be small.

4.3 Indoor positioning:

The approach of evaluating indoor positioning is similar to the translation evaluation extended by Geiger (Geiger at al., 2012). Our evaluation firstly locates the corresponding pose information in the submitted trajectory results based on the timestamp of each pose in ground truth files. Then, computes the average of translation errors for all possible sub-sequences of some lengths (5, 10, 25, 50 meters).

$$\mathcal{E}_{trans}(\delta) = \frac{1}{N}\sum_{i,j} trans\left(\delta_{i,j} \ominus \delta^*_{i,j}\right)^2 \tag{9}$$

where N is the number of relative sub-sequences, and $\ominus$ is the inverse of a standard motion composition operator. Let $\delta_{i,j}$ be the relative transformation from pose j to pose i and $\delta^*_{i,j}$ be the reference relative sub-sequence
The indoor positioning dataset provides two data sequences with ground truth for evaluation. Each ground truth trajectories file (in TXT format) contains an Nx9 table, where N is the number of frames of this sequence. The format of each row in the file is: {frame_id p_x p_y p_z q_x q_y q_z q_w timestamp}. Here, frame_id is the index of lidar frame with the current pose, p_x, p_y, and p_z are the translation components of the current pose,

q_x, q_y, q_z, and q_w are the quaternion representations of the rotation component of the current pose.
The dataset also provides three data sequences for submitting results. In the submitted trajectory file, each line in the file formats as: {frame_id p_x p_y p_z timestamp(UTC time(s))}.
The evaluation table will rank methods according to the average of translation errors, where errors are measured in percent.

4.4 Examples of dataset

Fig. 14 shows some examples of this dataset. Fig 14 (a) shows a frame of the Velodyne VLP-16 LiDAR data. Different color represents the intensity of every point; the brighter color means the stronger intensity. Fig 14 (b) shows the high accuracy data from Riegl VZ 1000, which is used as Indoor LiDAR SLAM ground truth. Fig 14 (c) and (d) show two examples of BIM benchmark, and Fig 14 (e) and (d) show two examples of indoor positioning benchmark. The blue dots in (d) are trajectories generated from the LiDAR-based SLAM method, and the yellow dots are trajectories generated by the smartphone sensor data.

5. CONCLUSION

This paper presents the design of the benchmark dataset on multisensory indoor mapping and position (MIMAP). Each scene in the dataset contains the point clouds from the multi-beam laser scanner, the images from fisheye lens cameras, and the records from the attached smartphone sensors. The benchmark dataset can be used to evaluate algorithms on: (1) SLAM-based indoor point cloud generation; (2) automated BIM feature extraction from point clouds; and (3) low-cost multisensory indoor positioning, focusing on the smartphone platform solution.

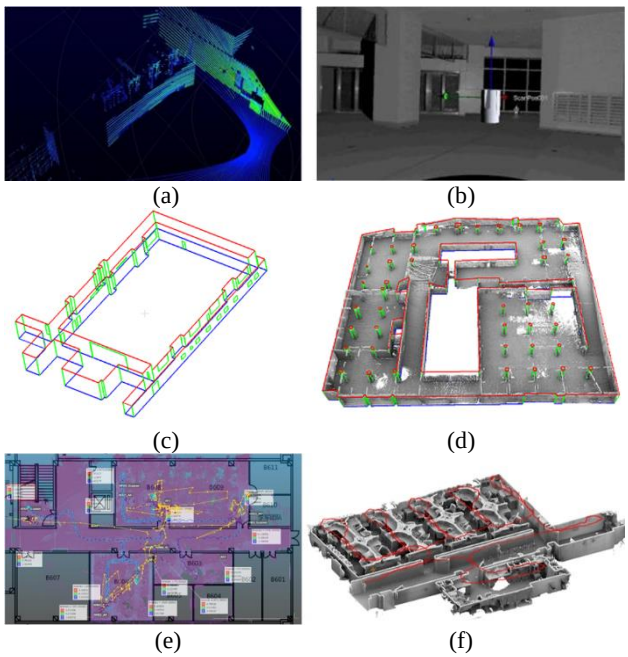


Figure 14. (a)A single frame from the LiDAR stream. (b) An indoor view of Riegl VZ 1000 data. (c) BIM structure model of a circular corridor. (d) BIM structure model with its point cloud. (e) An example of the indoor positioning benchmark. (f) The ground truth trajectory with the corresponding point cloud.

6. ACKNOWLEDGEMENTS

This work was supported by ISPRS Scientific Initiatives 2019 "ISPRS BENCHMARK ON MULTISENSORY INDOOR MAPPING AND POSITIONING", and National Natural Science Foundation of China project U1605254 and 61771413.

7. REFERENCES

Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., & Savarese, S., 2016. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1534-1543).

Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3354-3361).

Girardeau-Montaut, D., 2011. 3D point cloud and mesh processing software. Open Source Project. cloudcompare.org

Gong, Z., Wen, C., Wang, C., Li, J., 2018. A target-free automatic self-calibration approach for multibeam laser scanners. *IEEE Trans. Instrum. Meas*, *67*(1), 238-240.

ISPRS Commission I WG I/6, ISPRS Benchmark on Multisensorial Indoor Mapping and Positioning, 2020. http://www2.isprs.org/commissions/comm1/wg6/isprs-benchmark-on-multisensory-indoor-mapping-and-positioning.html

Kümmerle, R., Steder, B., Dornhege, C., Ruhnke, M., Grisetti, G., Stachniss, C., Kleiner, A., 2009. On measuring the accuracy of SLAM algorithms. *Auton. Robots*, *27*(4), 387.

Lin, T. Y., Maire, M., and et.al., 2014. Microsoft coco: common objects in context. In European Conference on Computer Vision (pp. 740-755). Springer, Cham.

Lehtola, V. V. , Kaartinen, H., Nüchter, A., Kaijaluoto, R., Kukko, A., Litkey, P., Honkavaara, E., Rosnell, T., Vaaja , M. T., Virtanen, J-P., Kurkela, M., El Issaoui, A. , Zhu, L., Jaakkola A & Hyyppä, J., 2017. Comparison of the selected state-of-the-art 3D indoor scanning and point cloud generation methods. *Remote Sens.*, 9 (8), 796.

Scaramuzza, D., Martinelli, A., Siegwart, R., 2006. A toolbox for easily calibrating omnidirectional cameras. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 5695-5701).

Wen, C., Pan, S., Wang, C., Li, J., 2016. An indoor backpack system for 2-D and 3-D mapping of building interiors. *IEEE Geosci. Rem. Sens. Lett.*, *13*(7), 992-996.

Wen, C., Dai, Y., Xia, Y., Lian, Y., Tan, J., Wang, C., Li, J., 2019. Toward efficient 3-D colored mapping in GPS/GNSS-denied environments. *IEEE Geosci. Rem. Sens. Lett.*, 17(1), 147-151.

Wang, C., Hou, S., Wen, C., Gong, Z., Li, Q., Sun, X., Li, J., 2018. Semantic line framework-based indoor building modeling using backpacked laser scanning point cloud. *ISPRS J. Photogramm. Rem. Sens.*, *143*, 150-166.

Zhang, J., Singh, S., 2014. LOAM: Lidar odometry and mapping in real-time. In *Robotics: Science and Systems* (Vol. 2, p. 9).