

ORIGINAL ARTICLE

Hong Kong UrbanNav: An Open-Source Multisensory Dataset for Benchmarking Urban Navigation Algorithms

Li-Ta Hsu* | Feng Huang | Hoi-Fung Ng | Guohao Zhang | Yihan Zhong |
Xiwei Bai | Weisong Wen

Department of Aeronautical and Aviation Engineering, Hong Kong Polytechnic University, Hong Kong

Correspondence

Li-Ta Hsu, Department of Aeronautical and Aviation Engineering, Hong Kong Polytechnic University, Hong Kong
Email: lt.hsu@polyu.edu.hk

Abstract

Accurate positioning in urban canyons remains a challenging problem. To facilitate the research and development of reliable and precise positioning methods using multiple sensors in urban canyons, we built a multisensory dataset, *UrbanNav*, collected in diverse, challenging urban scenarios in Hong Kong. The dataset provides multi-sensor data, including data from multi-frequency global navigation satellite system (GNSS) receivers, an inertial measurement unit (IMU), multiple light detection and ranging (lidar) units, and cameras. Meanwhile, the ground truth of the positioning (with centimeter-level accuracy) is postprocessed by commercial software from NovAtel using an integrated GNSS real-time kinematic and fiber optics gyroscope inertial system. In this paper, the sensor systems, spatial and temporal calibration, data formats, and scenario descriptions are presented in detail. Meanwhile, the benchmark performance of several existing positioning methods is provided as a baseline. Based on the evaluations, we conclude that GNSS can provide satisfactory results in a middle-class urban canyon if an appropriate receiver and algorithms are applied. Both visual and lidar odometry are satisfactory in deep urban canyons, whereas tunnels are still a major challenge. Multisensory integration with the aid of an IMU is a promising solution for achieving seamless positioning in cities. The dataset in its entirety can be found on GitHub at <https://github.com/IPNL-POLYU/UrbanNavDataset>.

Keywords

dynamic objects, gnss raw data, gnss/ins/lidar/camera, ground truth, localization, positioning, urban canyons

1 | INTRODUCTION

Accurate, globally referenced, cost-effective positioning service is a key component for the full autonomy of unmanned autonomous systems. Multi-sensor integration is believed to be a promising solution to achieve this goal based on typical sensors, including global navigation satellite systems (GNSSs), inertial measurement units (IMUs), light detection and ranging (lidar) units, and cameras. Satisfactory performance can be achieved in constrained areas (Wan et al., 2018)

or open areas (Levinson et al., 2011) with good sky visibility for GNSS positioning and abundant environment features for lidar units and cameras. Unfortunately, the performance of existing positioning technologies is significantly degraded in the urban scenarios of megacities, such as Hong Kong, Tokyo, and New York City, due to complex building structures, unstable illumination conditions, and excessive expected dynamic objects.

Poor GNSS positioning in urban canyons: GNSS positioning is significantly degraded in urban canyons because of signal reflection and blockage, leading to the notorious multipath and non-line-of-sight (NLOS) phenomenon (Groves et al., 2013; Hsu, 2018). Numerous existing methods have been investigated to mitigate the impacts of multipath and NLOS reception, such as three-dimensional (3D) mapping-aided (3DMA) GNSS (Zhong & Groves, 2022), camera-aided GNSS NLOS detection (Kato et al., 2016; Meguro et al., 2009; Wen, Bai, et al., 2019), and 3D lidar-aided GNSS NLOS detection (Wen, Zhang et al., 2018) or correction (Wen, Zhang, et al., 2019). Unfortunately, the GNSS positioning achieved thus far is still far from sufficient for fully autonomous systems requiring decimeter-level accuracy and stringent integrity requirements. Further work is needed to effectively solve the GNSS positioning problem in dense urban canyons.

Unreliable odometry estimation in highly dynamic scenarios: Visual/inertial integrated systems (VINSs) (Qin et al., 2018) can provide low-cost, locally accurate odometry estimation in environments with sufficient features using camera and IMU measurements. VINSs are characterized by advantages in size, power consumption, weight, and availability. Many state-of-the-art VINS pipelines with superior performance have been developed in the past several decades, such as filtering-based methods including multi-state constraint Kalman filter (MSCKF) (Mourikis & Roumeliotis, 2007), robust visual inertial odometry (ROVIO) (Bloesch et al., 2015), and OpenVINS (Geneva et al., 2020). Another research stream focuses on optimization-based VINS pipelines, including open keyframe-based visual-inertial SLAM (OKVIS) (Leutenegger et al., 2015), VINS-Mono (Qin et al., 2018), and oriented FAST and rotated BRIEF-based simultaneous localization and mapping 3 (ORB-SLAM3) (Campos et al., 2021). To further study the performance of VINSs in challenging outdoor urban canyons, our previous work evaluated and analyzed VINS-Mono (Qin et al., 2018) based on datasets collected in urban canyons of Hong Kong. The results (X. Bai et al., 2020) showed that the VINS accuracy was significantly decreased in the evaluated urban canyons, with the accumulated error reaching approximately 34.21 m for a driving distance of 2.1 km. The large errors are primarily attributed to outliers caused by dynamic objects and motion blur. Conversely, an active sensor, i.e., 3D lidar unit, can provide distance measurements for surrounding environments that are invariant to illumination. Therefore, lidar odometry (Shan & Englot, 2018) provides a more accurate and robust odometry estimation than VINSs. Unfortunately, lidar odometry shares the same drawback as VINSs, which are sensitive to dynamic objects. According to our previous evaluations (Wen, Hsu et al., 2018), lidar odometry can drift more than 2 m in a 2-km test in numerous urban scenarios.

Given these challenges raised by urban scenarios for existing key navigation techniques, it is desirable to provide an integrated multi-sensor dataset for both academic and industry fields to further improve existing navigation algorithms. Due to the rapid development of the autonomous driving industry, some datasets are open-source, such as the ApolloScape dataset from Baidu (Huang et al., 2018) and the Waymo open dataset from Google's autonomous driving research group (Sun et al., 2020). Unfortunately, these datasets share several similar drawbacks: (1) These datasets focus on object detection and perception algorithm developments that are not suitable for navigation algorithm development. For example,

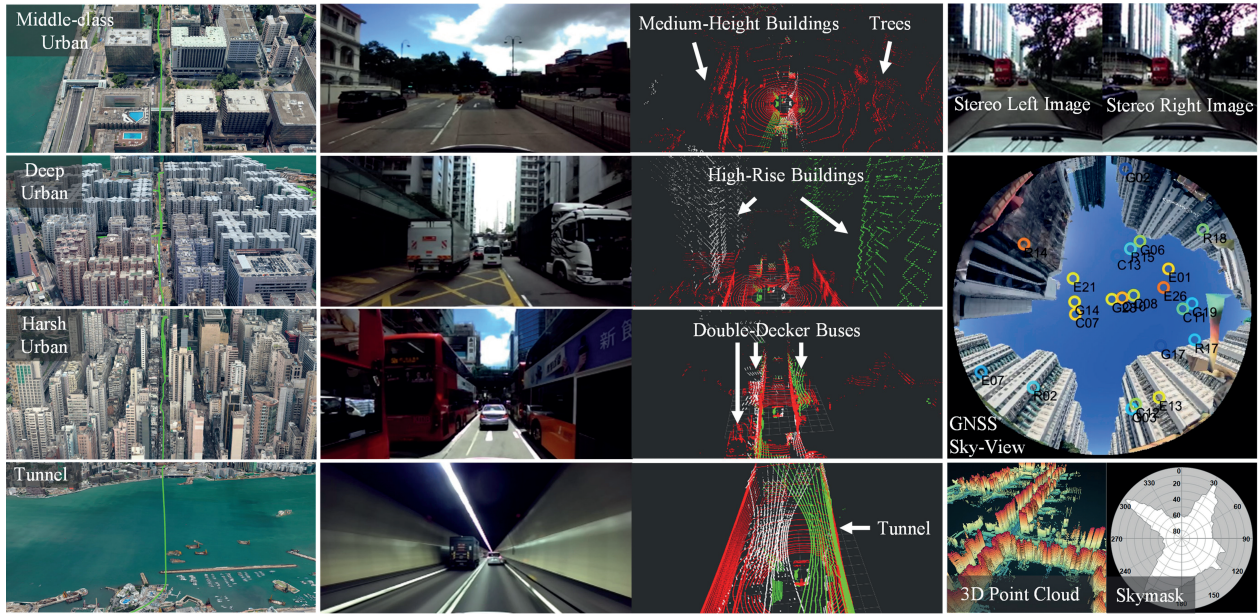


FIGURE 1 An overview of the *UrbanNav* dataset

The *UrbanNav* dataset focuses on typical urbanized scenarios in Hong Kong with navigation data from different grades of GNSS receivers: an IMU, multiple lidar units, cameras, and a high-accuracy ground truth system. Offline sky-view images are also provided for urban GNSS satellite visibility classification.

raw GNSS measurements (pseudorange, carrier phase, and Doppler frequency) are not provided, making it difficult for researchers to improve urban GNSS positioning. (2) The environments for dataset collection are primarily collected in suburban or open areas, which do not provide sufficient challenges for positioning and navigation algorithms. Realizing the strong need for accurate and cost-effective urban positioning, in 2019, a joint working group was formed under the joint efforts of the International Association of Geodesy (IAG) and The Institute of Navigation (ION). This working group is under Sub-Commission 4.1: *Emerging Positioning Technologies and GNSS Augmentations of IAG*. After consolidating suggestions and comments from international navigation researchers, this workgroup established the following objectives:

- Open-sourcing positioning sensor data, including GNSS, IMU, lidar, and camera data collected in Asian urban canyons;
- Raising awareness of the urgent navigation requirement for highly urbanized areas, especially in Asian-Pacific regions;
- Providing an integrated online platform for data sharing to facilitate the development of navigation solutions for the research community; and
- Benchmarking positioning algorithms based on open-source data.

Based on these objectives, we built a multisensory dataset, *UrbanNav*, collected in diverse challenging urban scenarios in Hong Kong and Tokyo (Hsu et al., 2021). In this paper, we focus on the Hong Kong dataset (see Figure 1). The key contributions of this paper are as follows:

1. The *UrbanNav* dataset in this paper introduces diversity in both sensor sources and scenarios. Specifically, *UrbanNav* includes raw sensor measurements from GNSS, lidar units, IMUs, and cameras, together with reliable ground truth of

positioning. Moreover, the dataset includes scenarios with different degrees of challenges for benchmarking existing navigation algorithms, such as middle-class, deep, and harsh urban scenarios.

2. *UrbanNav* provides raw GNSS measurements generated by different grades of receivers (smartphone, automobile, and geodetic levels of GNSS receivers). Moreover, multiple 3D lidar sensors are provided for research with different applications, ranging from robotics to large-scale mapping.
3. This paper benchmarks the majority of existing localization algorithms, including GNSS positioning with real-time kinematic library (RTKLIB) (Takasu & Yasuda, 2009), lidar odometry (Zhang & Singh, 2017), lidar inertial odometry (Shan et al., 2020), and VINS (Qin et al., 2018), using the *UrbanNav* dataset to provide a baseline performance for the community.

The remainder of this paper is structured as follows. Related works are reviewed in Section 2. The details of the data collection platform are introduced in Section 3. Details regarding the dataset and its evaluation are discussed in Section 4. Lessons learned during dataset development are presented in Section 5, and conclusions and directions for future work are presented in Section 6. Baseline positioning performance evaluations of lidar odometry, lidar inertial odometry, VINS, and GNSS are provided in the appendix of this paper.

2 | RELATED WORK

Over the past decades, numerous datasets have been released by research groups from both academia and industry to advance navigation research. These datasets have enabled researchers worldwide to engage in navigation research without system or data limitations. Table 1 shows a detailed comparison of the existing datasets related to navigation research. A new college dataset (Smith et al., 2009) was released in 2009, based on an IMU, two-dimensional (2D) lidar, camera, and ground truth positioning; this dataset was collected for campus scenarios. This dataset has been widely used in the robotics field for 2D lidar simultaneous localization and mapping (SLAM) research. However, the sensor sources are limited and are insufficient for urban navigation research. In 2010, the urban challenge dataset of the Defense Advanced Research Projects Agency (Huang et al., 2010) was released, based on an autonomous driving competition. The dataset was collected in urban and forest scenarios, obtained from lidar units and IMUs together with ground truth from GNSS real-time kinematic (RTK) positioning. However, the DARPA urban challenge dataset shares the same drawback as the new college dataset, in that the sensor sources are highly limited. In 2011, 3D lidar units became available, although expensive for the typical research community. During this time, the Ford campus dataset (Pandey et al., 2011) was released, with increased diversity in sensors. However, the raw GNSS measurements were not available, and the diversity of scenarios was very limited. As a result, this dataset is not widely used.

Due to the catalyzing effect of autonomous driving technologies, the KITTI dataset (Geiger et al., 2012), one of the most well-known datasets within the autonomous driving and robotics community, was released in 2012. Specifically, this dataset includes images from multiple cameras, 3D point clouds captured by expensive 64-channel 3D lidar sensors, global positioning data from a geodetic GNSS receiver (but no raw GNSS measurements), and raw measurements from an IMU sensor. Moreover, the KITTI dataset launched several benchmark tools,

TABLE 1
Comparison of Sensor Setups for Existing and Proposed Datasets

Existing Dataset	Environments	Raw GNSS Data	Ground Truth	IMU	Lidar	Sensors			Reference
						Camera	Geodetic GNSS	Automobile GNSS	Smartphone GNSS
New College	Campus Scenarios		✓	✓	✓	✓			(Smith et al., 2009)
DARPA Urban Challenge	Urban		✓	✓	✓				(Huang et al., 2010)
Ford Campus Dataset	Middle-class Urban		✓	✓	✓	✓	✓		(Pandey et al., 2011)
KITTI Dataset	Urban		✓	✓	✓	✓	✓		(Geiger et al., 2012)
NCLT Dataset	Campus Scenarios		✓	✓	✓	✓		✓	(Carlevaris-Bianco et al., 2016)
Oxford Dataset	Urban		✓	✓	✓	✓		✓	(Maddern et al., 2017)
ApolloScape Dataset	Urban		✓	✓	✓	✓	✓		(Huang et al., 2018)
Waymo Dataset	Urban		✓	✓	✓	✓	✓		(Sun et al., 2020)
nuScenes Dataset	Urban		✓	✓	✓	✓	✓		(Caesar et al., 2020)
A2D2 Dataset	Urban		✓	✓	✓	✓	✓		(Geyer et al., 2020)
Ford Multi-AV Seasonal Dataset	Urban		✓	✓	✓	✓	✓		(Agarwal et al., 2020)
EU Long-term Dataset	Urban		✓	✓	✓	✓	✓		(Yan et al., 2020)
UrbanLoco	Urban	✓	✓	✓	✓	✓		✓	(Wen et al., 2020)
Google Smartphone Dataset	Urban	✓	✓	✓					(Zangenehnejad & Gao, 2021)
M2DGR	Diverse Campus Scenarios	✓	✓	✓	✓	✓		✓	(Yin et al., 2022)
UrbanNav (Proposed Dataset)	Diverse Urban Scenarios	✓	✓	✓	✓	✓	✓	✓	

enabling the community to verify and compare algorithms on lidar odometry, visual odometry, environment understanding, object detection, and scenario segmentation. Meanwhile, the KITTI dataset includes numerous scenarios ranging from open to urban areas. However, the dataset is less applicable to the satellite navigation research community, as raw GNSS measurements are unavailable. Moreover, the scenarios are not sufficiently challenging for GNSS positioning, as the typical GNSS positioning error in the KITTI dataset is within 4 m. In 2016, the north campus long-term (NCLT) dataset (Carlevaris-Bianco et al., 2016), collected for campus scenarios, was released. Importantly, the automobile-level GNSS solution was provided, but raw GNSS measurements were still not available. In 2017, the Oxford dataset (Maddern et al., 2017) was released, focusing on autonomous driving vehicles in urban scenarios. Similar datasets, such as the ApolloScape dataset (Huang et al., 2018) from Baidu, Waymo dataset (Sun et al., 2020) from Google, nuScenes dataset (Caesar et al., 2020), A2D2 dataset (Geyer et al., 2020), Ford multi-autonomous vehicle (AV) seasonal dataset (Agarwal et al., 2020), and European Union (EU) long-term dataset (Yan et al., 2020), were also released. These datasets were mainly developed for autonomous driving vehicles, focusing on SLAM, object detection, and scene segmentation. As a result, these datasets do not provide raw GNSS measurements for navigation research. In 2020, the UrbanLoco dataset was released (Wen et al., 2020), including data for downtown San Francisco and Hong Kong. Notably, the dataset provides raw GNSS measurements collected by a u-blox M8T receiver (automobile-level). Currently, researchers in the fields of robotics and intelligent transportation systems widely use this dataset. However, the UrbanLoco dataset has several key drawbacks. First, only single-frequency global positioning system (GPS)/BeiDou measurements were collected, and the diversity of GNSS receivers is limited, as only an automobile-level GNSS receivers was applied. Second, the trajectories of the released dataset are short because the dataset was originally designed for SLAM research with multi-sensor fusion. In response to this challenge and the urgent need for high-accuracy urban GNSS positioning, the Google Smartphone Challenge dataset (Zangenehnejad & Gao, 2021) was released in 2021 at the ION GNSS+ conference. Specifically, this dataset provides raw GNSS measurements collected by a smartphone together with ground truth positioning for urban scenarios in California. This dataset represented a substantial step toward urban GNSS positioning. Unfortunately, lidar and camera measurements are not available from the Smartphone Challenge dataset, and thus, this dataset does not satisfy the requirements of the autonomous driving community. In 2022, the multi-modal and multi-scenario dataset for ground robots (M2DGR) (Yin et al., 2022) dataset was released; this dataset was collected in diverse campus scenarios, including both indoor and outdoor environments, using a ground robot. However, this dataset focuses on the ground robot and is therefore not suitable for autonomous driving research.

In short, autonomous driving research has led to the release of many multi-sensor datasets with GNSS/IMU/lidar/camera measurements. However, these datasets were not specifically designed for navigation research, as they lack raw GNSS measurements and diversity in different levels of GNSS receivers and urbanization. Therefore, it is highly desirable to provide an integrated, navigation-oriented multi-sensor dataset collected from urban scenarios for the research community, which is the key objective of this paper. Both Hong Kong and Tokyo *UrbanNav* datasets were first presented in the commercial-track ION GNSS+ 2021 (Hsu et al., 2021). We introduce the technical details in this paper to help fellow researchers better understand the dataset collected in Hong Kong.

3 | OVERVIEW OF THE DATA COLLECTION PLATFORM

3.1 | System Configuration

3.1.1 | Platform and Hardware Connection

Data collection platform and hardware configuration: The data collection platform is shown in Figure 2. All of the sensors are integrated into a compact sensor kit (see Figure 2(b)) installed on the roof of an automobile (*Honda Jazz/Fit*). Data from the GNSS receivers, cameras, lidar units, and IMU are collected by a desktop computer (i7 processor, 512-GB solid-state drive for data logging) and the Linux system installed with a robot operation system (ROS) (Quigley et al., 2009). The framework of the hardware connection for data collection is shown in Figure 3. The details of the sensors used in this dataset are listed in Table 2.

GNSS receivers for raw GNSS measurement collection: In this dataset, we provide raw GNSS measurements (e.g., pseudorange, carrier-phase, and Doppler frequency measurements) from different levels of GNSS receivers. We also provide a [tutorial](#) on how to download the ephemeris data from continuously operating reference stations in Hong Kong.

- **Smartphone-level GNSS receiver:** The Xiaomi 8 smartphone is employed to collect the multi-constellation and multi-frequency raw GNSS measurements, including measurements from GPS (L1, L5), GLONASS (G1), GALILEO (E1, E5a), BeiDou (B1), and quasi-zenith satellite system (QZSS) (L1, L5) at 1 Hz.
- **Automobile-level GNSS receiver:** The u-blox M8T and u-blox F9P are employed to collect raw GNSS measurements at 1 Hz. First, a u-blox M8T is employed to collect data from GPS (L1), GLONASS (G1), GALILEO (E1), and BeiDou (B1) at 1 Hz. Second, a u-blox F9P is employed to collect data from GPS (L1, L2), GLONASS (G1, G2), GALILEO (E1, E5b), BeiDou (B1, B2), and QZSS (L1, L2). Third, another u-blox F9P is attached with a geodetic antenna to evaluate the performance of the u-blox F9P receiver with a high-end GNSS antenna.
- **Geodetic-level GNSS receiver:** The NovAtel FlexPak6 receiver is employed to collect raw GNSS measurements, including data from GPS (L1, L2), GLONASS (L1, L2), GALILEO (E1, E5b), and BeiDou (B1I, B2I), at 1 Hz.

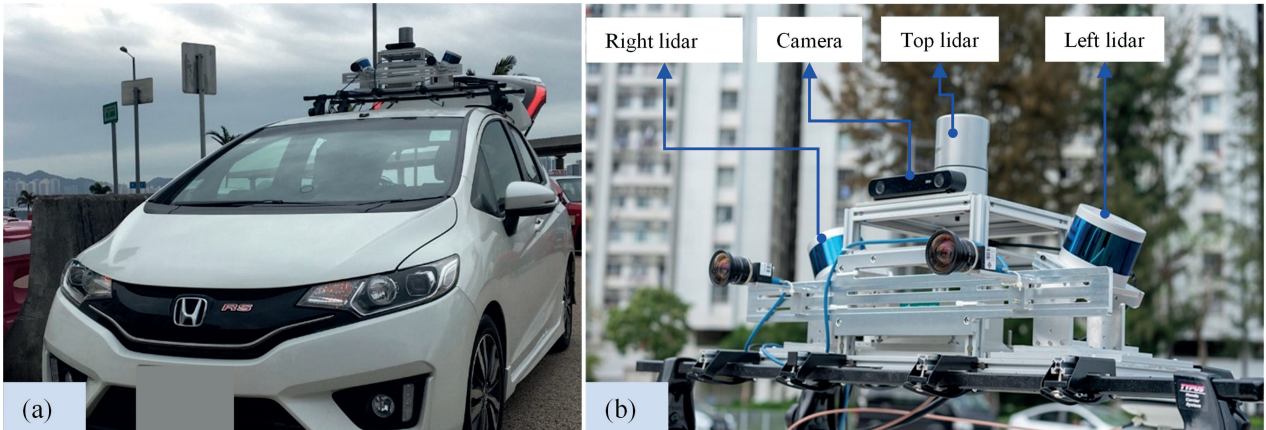


FIGURE 2 Dataset collection platform

(a) The data collection vehicle (Honda Jazz/Fit) with the sensor kit installed on top and (b) the sensor kit with GNSS/IMU/lidar/camera sensors.

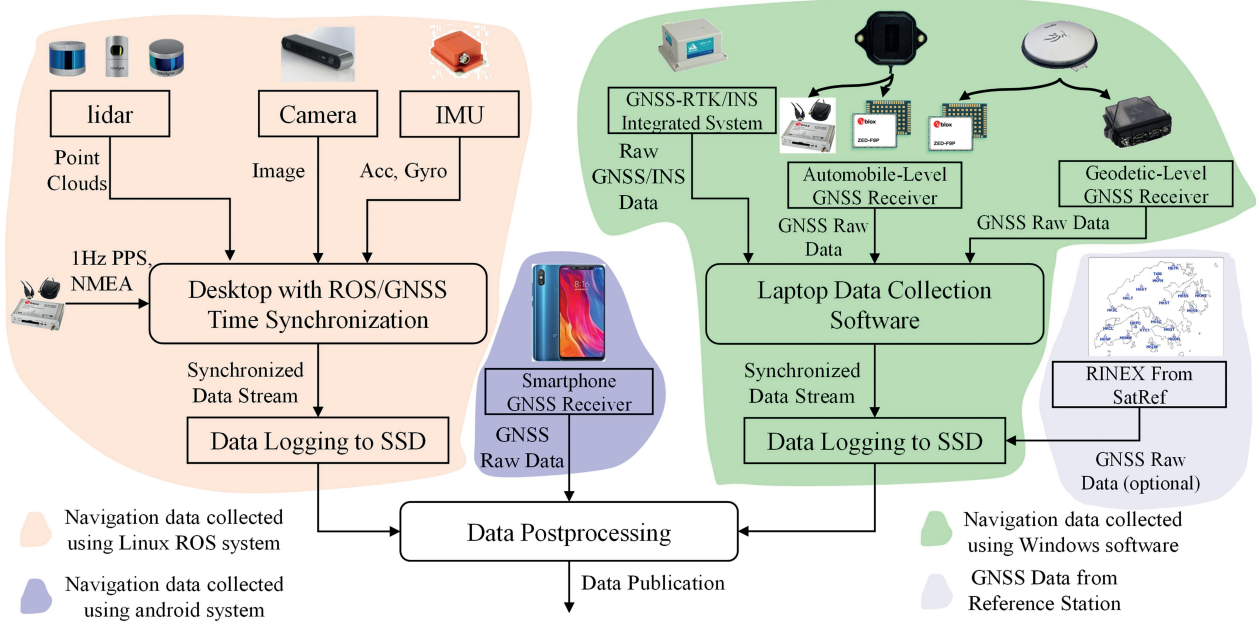


FIGURE 3 Flowchart of the hardware connection for the data collection platform. All sensor measurements are carefully synchronized to the GPS time. SatRef represents the Hong Kong satellite positioning reference station network (<https://www.geodetic.gov.hk/en/satref/satref.htm>). Acc=accelerometer, Gyro=gyroscope measurements, SSD=solid-state disk.

Lidar for 3D point cloud collection: In this dataset, three lidar sensors are employed to collect 3D point clouds from the surrounding environments. First, a 32-channel 3D lidar unit from Velodyne (Velas et al., 2014) is installed on top of the data collection vehicle to horizontally capture the surroundings. Second, a slant 16-channel 3D lidar unit from Velodyne is installed on the right side of the sensor kit. Third, a low-cost slant 16-channel 3D lidar device from Leishen Technologies (Z. Bai et al., 2020) is installed on the left side of the sensor kit. The lidar units installed on both sides are tilted outward to capture the high-rise building structures for mapping and SLAM applications. All of the lidar data are collected at 10 Hz.

Camera for image collection: In this dataset, a stereo camera from ZED (Varma et al., 2018) is employed to collect images at a frequency of 27 Hz. The raw image data can be used for visual odometry-based positioning research.

IMU for acceleration and gyroscope data collection: In this dataset, an Xsens-MTI-30 IMU is employed to collect raw acceleration and angular velocity measurements at a frequency of 400 Hz.

Ground truth of positioning in GNSS-available scenarios: In this dataset, the ground truth of positioning is provided by a NovAtel SPAN-CPT (Kennedy et al., 2006), a GNSS (GPS, GLONASS, and BeiDou) RTK/inertial navigation system (INS) (fiber-optic gyroscope [FOG]) integrated navigation system. The gyro bias in-run stability of the FOG is 1° per hour, and its random walk is 0.067° per hour. The baseline between the rover (SPAN-CPT) and the GNSS base station is within 7 km. According to the specifications of the NovAtel SPAN-CPT, centimeter-level accuracy can be obtained when the RTK correction is available with the correct fixed solution. However, accuracy is not guaranteed in urban canyons with inferences from building reflections. Therefore, we collect raw measurements from the SPAN-CPT and postprocess the data using state-of-the-art Inertial Explorer software from NovAtel (Kennedy et al., 2006), which maximizes the accuracy of the trajectory by processing forward and backward in time, performing a backward

TABLE 2
Sensors Used in the *UrbanNav* Dataset
The GPS time and ROS time are synchronized using Chrony (Soares et al., 2020).

Sensor	Grade	Sensor Model	Update Frequency	Time Standard	Remarks
GNSS Receiver	Automobile-Level	u-blox ZED-F9P	1 Hz	GPS Time	GPS (L1, L2), GLONASS (G1, G2), GALILEO (E1, E5b), BeiDou (B1I, B2I), QZSS (L1, L2). A low-cost patch antenna is used .
	Automobile-Level	u-blox ZED-F9P	1 Hz	GPS Time	GPS (L1, L2), GLONASS (G1, G2), GALILEO (E1, E5b), BeiDou (B1I, B2I), QZSS (L1, L2). A NovAtel GPS-703-GGG antenna is used .
	Automobile-Level	u-blox EVK-M8T	1 Hz	GPS Time	GPS (L1), GLONASS (G1), GALILEO (E1), BeiDou (B1I). A low-cost patch antenna is used .
GNSS Receiver	Smartphone-Level	Xiaomi 8	1 Hz	GPS Time	GPS (L1, L5), GLONASS (G1), GALILEO (E1, E5a), BeiDou (B1I), QZSS (L1, L5)
GNSS Receiver	Geodetic-Level	NovAtel FlexPak6	1 Hz	GPS Time	GPS (L1, L2), GLONASS (G1, G2), GALILEO (E1, E5b), BeiDou (B1I, B2I). A NovAtel GPS-703-GGG antenna is used .
Lidar	Automobile-Level	Velodyne 32E	10 Hz	ROS Time	https://github.com/ros-drivers/velodyne/tree/1.6.1
Lidar (Right)	Automobile-Level	Velodyne VLP16	10 Hz	ROS Time	https://github.com/ros-drivers/velodyne/tree/1.6.1
Lidar (Left)	Automobile-Level	Lslidar C16	10 Hz	ROS Time	http://wiki.ros.org/lslidar_c16
Stereo Camera	Automobile-Level	ZED2	27Hz	ROS Time	https://github.com/willdeng/zed_cpu_ros
IMU	Automobile-Level	Xsens-MTI-30	400 Hz	ROS Time	https://github.com/ethz-asl/ethzasl_xsens_driver

smoothing step, and combining the results. Inertial Explorer can significantly improve the overall accuracy of the ground truth of positioning.

Ground truth of positioning in GNSS-denied scenarios: One of our datasets involves scenarios in a tunnel, where GNSS positioning is not available. As a result, the NovAtel SPAN-CPT cannot collect reliable GNSS measurements for the tunnel dataset. Fortunately, the NovAtel SPAN-CPT involves high-accuracy INS, which can provide low-drift dead-reckoning. To this end, we initialize the NovAtel SPAN-CPT from an open area in which the GNSS-RTK can easily obtain a fixed solution. As a result, the INS bias can be effectively calibrated. Then, the data collection vehicle enters the tunnel and exits the tunnel within 7 min. Next, the fixed GNSS-RTK solution can be obtained again by the NovAtel SPAN-CPT. In short, high-accuracy positioning can be achieved immediately before and after the data collection vehicle enters and exits the tunnel. Finally, the Inertial Explorer software is used to postprocess the collected raw data from the NovAtel SPAN-CPT to obtain reliable ground truth positioning. We carefully check the ground truth positioning of the tunnel data from the NovAtel SPAN-CPT based on the geodetic map from the Survey and Mapping Office (SMO) of Hong Kong. At least meter-level accuracy can be guaranteed for the challenging tunnel dataset.

GNSS reference station data: The SMO of the Lands Department of the Hong Kong government provides GNSS reference station service, named the Hong Kong Satellite Positioning Reference Station Network (SatRef). SatRef consists of 16 reference stations and 2 integrity monitoring stations evenly distributed throughout Hong Kong. SatRef provides raw data for postprocessing in receiver independent exchange format (RINEX) format for web or file transfer protocol (FTP) download; details can be found at <https://www.geodetic.gov.hk/en/satref/satref.htm>. This resource provides both RINEX 2 and 3 versions, with a file length of 1 h or 24 h. For RINEX 3.02, the file length of 1 h provides a data interval of 1 s or 5 s; for 24 h, only a data interval of 30 s is provided.

3.1.2 | Time Synchronization of Different Sensors

Different sensors have different time systems, and it is important to effectively synchronize the diverse measurements. To achieve this, we use Chrony (Soares et al., 2020) to acquire the National Marine Electronics Association (NMEA) pulse per second (PPS) (Niu et al., 2015) signal from a u-blox EVK-M8T GNSS receiver to synchronize the ROS time from the desktop computer with the GPS time. Meanwhile, an NMEA driver is also used to record the time difference between ROS time and GPS time. In this case, the time between the ROS time (desktop computer) and the standard GPS time is synchronized. Therefore, the timestamps for those sensors are recorded as follows:

- *Timestamp for the camera sensor:* The images from the camera are collected using the driver in ROS provided by the manufacturer. Therefore, the images are recorded with ROS timestamps upon arrival of the image.
- *Timestamp for lidar sensors:* The 3D point clouds from the lidar units are collected using the driver in ROS. Similar to the images, the data are recorded with ROS timestamps upon arrival of the 3D point clouds.
- *Timestamp for the IMU sensor:* Raw acceleration and angular velocity measurements from the IMU are collected using the default ROS driver. Similar to the images, the data are recorded with ROS timestamps upon arrival of the raw IMU data.

- *Timestamp for GNSS smartphone, u-blox (automobile-level), and NovAtel (geodetic-level) receivers:* Raw GNSS measurements from the smartphones are collected using the *Geo++ RINEX Logger* and *GNSSLogger* application (Wanninger & Heßelbarth, 2020). Raw GNSS measurements from the u-blox and NovAtel receivers are collected in Windows using a laptop through the u-center and NovAtel Connect software, respectively, provided by the manufacturer. All of these GPS measurements are stamped with the GPS time.
- *Timestamp for the ground truth system:* The ground truth positioning is provided by the SPAN-CPT system. Raw data (including raw GNSS and IMU data) are collected using the NovAtel Connect software provided by NovAtel under the Windows system using a laptop. Therefore, the data are timestamped with the GPS time.

3.2 | Sensor Calibration

The extrinsic parameters of different sensors are important for sensor integration. Moreover, the intrinsic parameters associated with the camera are also required, such as the projection factor and distortion parameters. This section presents the process for calibrating associated intrinsic and extrinsic parameters. Computer-aided design software is utilized to design the 3D model of the sensor rack. The machining of the sensor rack is supported by a precise manufacturer. Measurements from the drawing are used as initial guesses for the calibration methods described below. In this dataset, we align all of the sensor measurements to the IMU sensor, which is not impacted by environmental changes. An illustration of the coordinate system of the platform is shown in Figure 4.

3.2.1 | IMU Calibration

Noise parameters are important for integration of the IMU with additional sensors. In this dataset, the noise parameters for both the gyroscopes and the

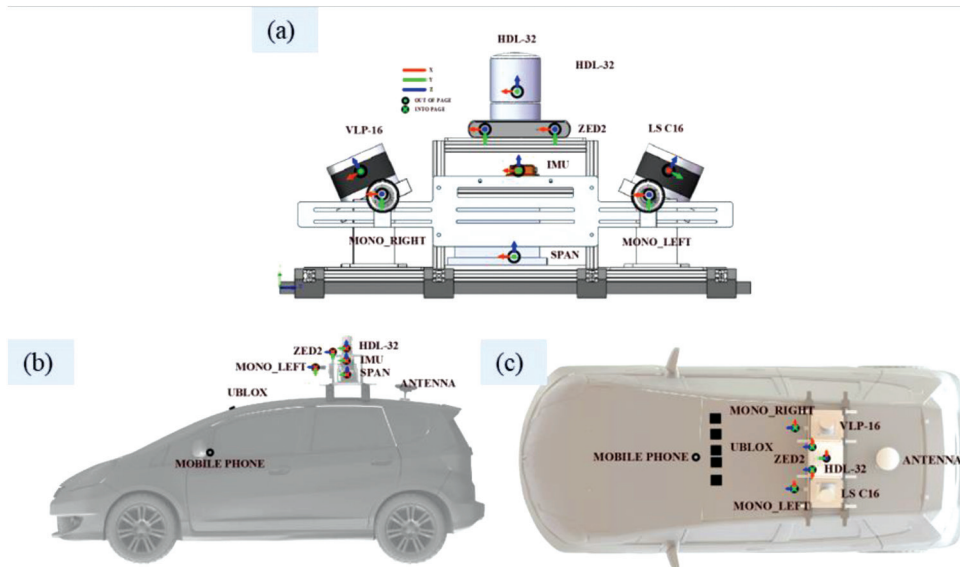


FIGURE 4 (a) Configuration of the sensor installation, (b) side view, and (c) top view

accelerometers are calibrated using the Allan variance (El-Sheimy et al., 2007). To achieve this, static data from the IMU are collected within 2 h and then processed via Allan variance analysis. The details of the IMU noise parameters can be found on our designed *UrbanNav* [page](#).

3.2.2 | *Visual-Inertial Calibration*

The accuracy of the intrinsic camera parameters is significant for the performance of the visual-based positioning and integration with other sensors. In this dataset, Kalibr (Furgale et al., 2013), a toolbox that can resolve intrinsic camera calibration and visual-inertial calibration problems, was employed to calibrate the intrinsic camera parameters. Kalibr can also enable the calibration of extrinsic parameters between the IMU and the camera. To achieve this, an Aprilgrid board, which is recommended by Kalibr, is designed to collect the raw image data together with the IMU data from a constrained area with a clear background. All of these data are recorded using a rosbag (Quigley et al., 2009). Finally, the recorded data are processed by Kalibr to estimate the following parameters: 1) intrinsic camera parameters, 2) extrinsic parameters between the left camera of ZED2 and the IMU, and 3) extrinsic parameters between the right camera of ZED2 and the IMU. All of these parameters can be found on our designed *UrbanNav* [page](#). Moreover, in the evaluation of the quality of extrinsic parameter calibration between the cameras and the IMU, the mean reprojection error of extrinsic parameters between the left camera and the IMU is 0.57 pixels, and the mean reprojection error of extrinsic parameters between the right camera and the IMU is 0.68 pixels.

3.2.3 | *Lidar Calibration*

For extrinsic parameters between the top lidar unit and the IMU sensor, this dataset uses the continuous-time batch optimization-based method (Lv et al., 2020), using the extrinsic parameters from the 3D drawing as an initial guess. Because the top lidar is a 32-channel that can provide 3D point clouds, the lidar odometry can be locally accurate over a short period. Therefore, the calibration of extrinsic parameters between the top lidar unit and the IMU sensor can be guaranteed with this method (Lv et al., 2020). Then, we calibrate the extrinsic parameters of the left and right lidar units to the top lidar device using Autoware (Kato et al., 2018) based on their 3D point clouds. Finally, the extrinsic parameters calibrated between the three lidar sensors and the IMU are obtained. The details of these parameters can be found on our designed *UrbanNav* [page](#). Moreover, we confirm the quality of the estimated extrinsic lidar parameters by checking the degree of overlap between the 3D point clouds from different lidar units scanning the same regular wall.

3.2.4 | *GNSS Receiver Level-Arm Calibration*

Because the calibration of extrinsic parameters (also called level-arm) between the GNSS antenna (u-blox receiver) and the IMU only involve translation, we mechanically calibrate the extrinsic parameters based on the 3D drawing with an accuracy of better than 5 cm. The details of these parameters can be found on our designed *UrbanNav* [page](#).

4 | DATASET DESCRIPTION

4.1 | Scenarios

The degree of urbanization of a given scenario is primarily determined by the density and height of the buildings on site. This paper uses the mean masking elevation angle (MMEA), which is introduced in Section 3 of our previous work (Wen et al., 2020), to describe the degree of urbanization. Specifically, a tall building will lead to a higher masking elevation angle. Based on the urbanization level evaluated from the 3D building models, we selected four representative scenarios that cover typical environments for urban navigation, as shown in Figure 5: (a) middle-class urban, (b) deep urban, (c) harsh urban, and (d) tunnel. The key features of these datasets are shown in Table 3. A video demonstration of the sky-view images based on a virtual fisheye camera simulated from Google Earth Studio (Suzuki & Kubo, 2015) is available [here](#).

- **Middle-class urban:** In this scenario, the vehicle starts data collection in a wide street environment for better initial positioning performance. Then, the vehicle enters an environment with middle-class-height buildings (50 m) on two sides of a normal-width street (two-lane road). Next, the vehicle enters the main road, where the west side has a line of uniformly arranged buildings and the east side has a relatively clear sky-view. Finally, the vehicle returns to a wide street environment and achieves a loop closure. Thus, this dataset represents a typical urban worldwide scenario with a limited sky-view and numerous dynamic objects (cars, pedestrians, etc.).
- **Deep urban:** In this scenario, the vehicle starts in an open-sky environment near the seashore to ensure ground truth quality. Then, the vehicle travels along a wide street (three-lane road) and enters a narrow street with two lanes closely adjacent to buildings. Here, nearly 70% of the sky-view is blocked by

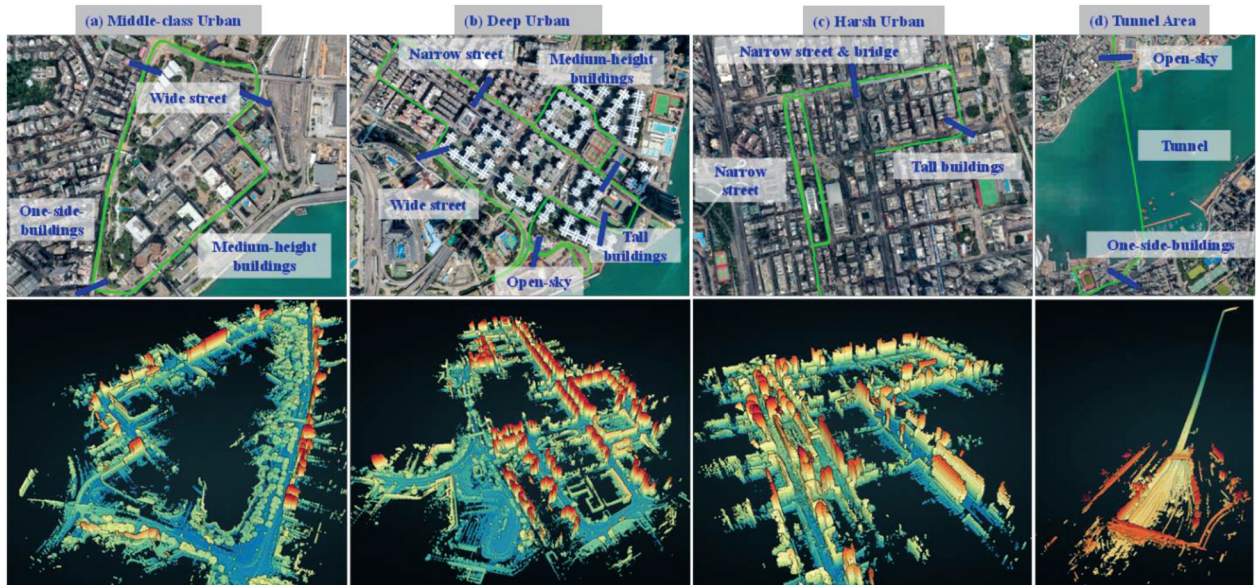


FIGURE 5 Illustration of the trajectories (top panel) and the associated environment reconstruction (bottom panel) of the four selected datasets

The trajectory is generated from the ground truth positioning given by the SPAN-CPT. (a) Middle-class urban canyon, (b) deep urban canyon, (c) harsh urban canyon, and (d) tunnel.

TABLE 3
Comparison of the Four Selected Datasets

<i>Dataset</i>	<i>Location</i>	<i>Skymask Indicator (MMEA)</i>	<i>Path Length</i>	<i>Total Size</i>
Middle-Class Urban	Tsim Sha Tsui, Hong Kong	44.3°	3.64 km	33.7 GB (785 s)
Deep Urban	Whampoa, Hong Kong	46.9°	4.51 km	63.9 GB (1,536 s)
Harsh Urban	Nathan Road, Hong Kong	55.2°	4.86 km	147 GB (3,367 s)
Tunnel	Cross Harbor Tunnel, Hong Kong	N/A	3.15 km	17 GB (398 s)

buildings, resulting in poor GNSS performance. Next, the vehicle enters a residential area with medium-height buildings (50 m) and a few tall buildings. Finally, the vehicle returns to an open-sky environment for loop closure. These data represent a deep urban scenario in which the GNSS cannot obtain satisfactory accuracy due to the poor sky-view. A large number of dynamic objects are present during data collection.

- **Harsh urban:** For this dataset, the vehicle starts data collection on a narrow street environment, with a road width of approximately 10 m and tall buildings (ranging from 30 to 90 m) lined up closely on either side. After two closed loops, the vehicle enters another narrow street environment with a footbridge right over the vehicle, blocking most of the sky-view. Finally, the vehicle travels to a wider street surrounded by tall buildings. This scenario has an extremely limited sky-view, significantly reducing GNSS accuracy and availability. The scenario includes busy traffic loads and crowds, which degrade the performance of camera- or lidar-based navigation. This scenario represents the most challenging environment for any navigation aid.
- **Tunnel:** In this scenario, the vehicle starts in an open-sky environment and then enters the tunnel. The tunnel is 1.7 km long and four lanes wide. Next, the vehicle leaves the tunnel and ends the data collection in an environment with buildings on one side of the lane. This scenario represents a typical tunnel environment for autonomous driving, where the GNSS is completely blocked and there are insufficient features for localization by other sensors.

4.2 | Data Format

GNSS data: In this dataset, raw GNSS measurements are archived in RINEX3.02 format. To benefit the field of robotics research, we released an open-source tool, GraphGNSSLib (Wen & Hsu, 2021), to decode the RINEX3.02 data to the typical ROSTOPIC format (Quigley et al., 2009). Robotics researchers can easily access raw GNSS data for further study and algorithm development. We also provide some examples for decoding the RINEX3.02 data using GraphGNSSLib and for performing single-point positioning based on pseudorange measurements. More details can be found at this [GitHub](#) link.

Lidar data: In this dataset, the lidar data are recorded in the rosbag (Quigley et al., 2009) with a topic (/velodyne_points) for the top 32-channel lidar, a topic (/left/lslidar_point_cloud) for the left 16-channel lidar, and a topic (/right/

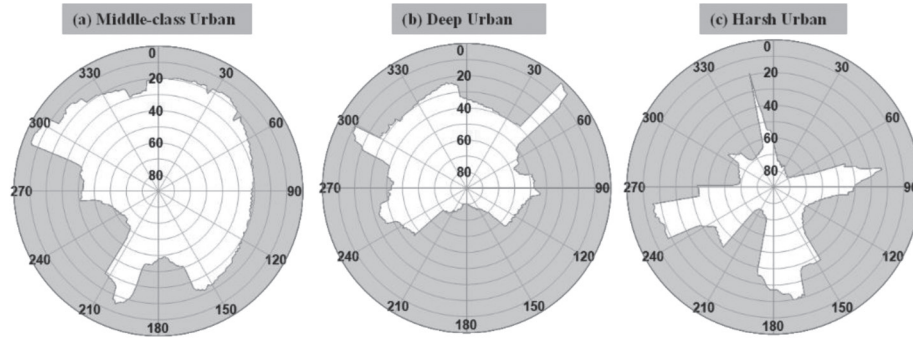


FIGURE 6 Illustration of a skymask in three randomly selected locations for (a) middle-class, (b) deep, and (c) harsh urban scenarios

velodyne_points) for the left 16-channel lidar. An illustrative video of using lidar data for lidar inertial odometry (Shan et al., 2020) can be found at this [link](#).

Camera data: In this dataset, the camera data are recorded in the rosbag (Quigley et al., 2009) with a topic (/zed2/camera/left/image_raw) for the left camera and a topic (/zed2/camera/right/image_raw) for the right camera. An illustrative video of using the image data for a visual-inertial navigation system (Qin et al., 2018) can be found at this [link](#).

IMU data: In this dataset, the IMU data are recorded in the rosbag (Quigley et al., 2009) with one topic (/imu/data).

Ground truth data: In this dataset, the ground truth positioning data are stacked in the rosbag (Quigley et al., 2009) with one topic (/novatel_data/inspvax) by postprocessing.

Skymask data: Recently, the 3DMA GNSS positioning method (Groves, 2016) has attracted attention because of its high potential to improve urban GNSS positioning performance. The skymask is the key component generated by the 3D building model, providing a skyplot with the building boundaries at a given location. The skymask is useful for satellite visibility prediction. It stores the highest elevation angle of the surrounding building boundaries, with 0.1° resolution, for each corresponding azimuth angle with 1° resolution. The skymask is usually generated offline and stored in a formatted CSV file to reduce the computational load (Ng et al., 2020). Each CSV file stores all potential locations and the skymask that covers the whole area of a single dataset location. Each row represents one single location in the file, consisting of WGS84 coordinates and the skymask. There are 364 entries in each line, separated by a comma. The first three entries are WGS84 coordinates, with latitude and longitude in degrees and altitude in meters. The remaining entries are the skymask of the highest elevation angle for building boundaries at an azimuth angle of 0° to 360° , giving a total of 361 entries. The skymask at 0° and 360° has the same value for convenient indexing based on the azimuth angle. Skymask data (see example in Figure 6) are provided for the middle-class, deep, and harsh urban datasets. More details can be found through this [link](#).

5 | BENCHMARK OF STATE-OF-THE-ART METHODS

5.1 | GNSS Single-Point Positioning

The weighted least square (WLS) method is the most widely used approach for GNSS single-point positioning. A flowchart of this method, which utilizes a straightforward algorithm, is shown in Figure 7.

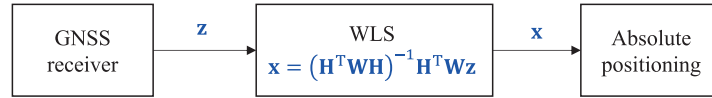


FIGURE 7 Flowchart of a general GNSS WLS method

$\mathbf{x}$, $\mathbf{H}$, $\mathbf{W}$, and $\mathbf{z}$ denote the state vector (position and time), measurement model (unit line-of-sight vectors), weighting matrix, and measurement vector (pseudoranges), respectively.

TABLE 4
WLS Processed by RTKLIB

Dataset	Receiver	Availability (%)	RMSE (m)	Mean (m)	STD (m)	Max (m)
Middle-Class Urban	NovAtel FlexPak 6	37.74	2.73	2.00	1.87	17.49
	U-blox F9P	83.61	16.67	10.86	12.65	75.59
	Xiaomi Mi 8	80.43	31.73	19.10	25.36	303.93
Deep Urban	NovAtel FlexPak 6	83.11	6.13	3.77	4.84	51.47
	U-blox F9P	99.94	14.57	10.42	10.19	68.47
	Xiaomi Mi 8	97.79	52.72	29.56	43.67	694.15
Harsh Urban	NovAtel FlexPak 6	68.60	45.52	9.92	44.44	1973.72
	U-blox F9P	27.01	85.79	32.78	79.32	820.14
	Xiaomi Mi 8	40.46	257.24	46.94	253.01	7457.34

Note: Max: maximum error; STD: standard deviation of the positioning error.

In this section, the GNSS WLS method is based on the RTKLIB (Takasu & Yasuda, 2009). The processing steps utilize the received GNSS constellations and frequencies depending on the receiver's capabilities. The processed results are shown in Table 4 for the middle-class, deep, and harsh urban scenarios. Note that the tunnel dataset is not evaluated, as the GNSS signal is not available in the tunnel. It can be seen from Table 4 that the mean error of the GNSS WLS positioning gradually increases as the complexity increases. Here, the availability represents the percentage of valid solutions over the whole period with an expected solution output rate of 1 Hz. Based on the results shown in Table 4, the positioning root mean square error (RMSE) is roughly proportional to the complexity of the environment of the dataset. Additionally, the positioning error increases across different receivers, from a geodetic-grade receiver to a low-cost smartphone receiver. The positioning error increased because the measurement quality is worse in a low-cost receiver than in a geodetic-grade receiver, as a result of the sampling rate of the radiofrequency signals and the receiver algorithm and equipped antenna. The trajectories are shown in Figure 8.

5.2 | Urban State-of-the-Art 3DMA GNSS Positioning

This section presents the positioning performance achieved by urban GNSS positioning with the aid of 3D building models, known as 3DMA GNSS. First, the WLS solution is used as an initial guess to distribute the particles. A skymask is generated based on the particles' locations and 3D models. The likelihood of each particle can consider the matching of both predicted and measured satellite visibilities (i.e., GNSS shadow matching) and predicted and measured satellite ranging measurements. Then, the weighted average (considering the likelihood) of the

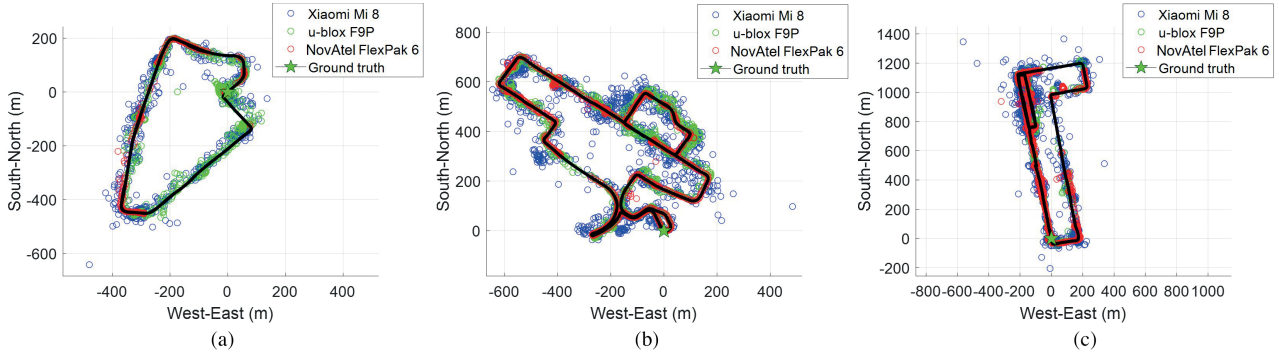


FIGURE 8 Illustration of the WLS positioning results based on RTKLIB for a (a) middle-class urban canyon, (b) deep urban canyon, and (c) harsh urban canyon

particles' locations is regarded as the receiver's absolute location. A flowchart of the 3DMA GNSS method is shown in Figure 9.

The results are processed based on our previous work (Ng et al., 2021) using the datasets collected in middle-class, deep, and harsh urban scenarios, as shown in Table 5. Shadow matching and likelihood-based-ranging 3DMA GNSS data are selected for the integration solution. Compared with the results in Table 4 (WLS results), a reduced positioning error can be found for commercial-grade and smartphone receivers. However, this result is not observed for the geodetic-grade receiver in the deep urban canyon dataset. This difference is attributed to the fact that a limited number of satellites with high elevation angles are visible at the beginning of the dataset or in a very harsh environment, and the sensor does not have adequate satellite visibility for shadow matching to match with the satellite in the ephemeris. Thus, the position estimation results converge to a local minima with a significant error. For more technical details of the 3DMA GNSS method evaluated here, readers are referred to our previous work (Ng et al., 2021). The trajectories are shown in Figure 10.

5.3 | GNSS Positioning via Factor Graph Optimization

The recently investigated factor graph optimization (FGO)-based formulation (Das et al., 2021; Wen & Hsu, 2021) shows improved robustness against the extended Kalman filter because of the global optimization and multiple iterations of FGO. A performance comparison of the extended Kalman filter and FGO for GNSS/IMU integration was presented in our previous work (Wen et al., 2021), which showed the advantages of FGO. Interestingly, the advantage of FGO in GNSS positioning

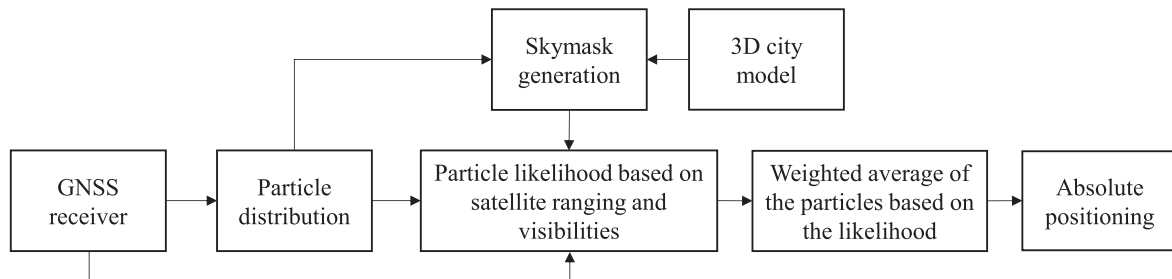


FIGURE 9 Flowchart of a general 3DMA GNSS method

TABLE 5
Integrated 3DMA GNSS Results

<i>Dataset</i>	<i>Receiver</i>	<i>Availability (%)</i>	<i>RMSE (m)</i>	<i>Mean (m)</i>	<i>STD (m)</i>	<i>Max (m)</i>
Middle-Class Urban	NovAtel FlexPak 6	82.34	260.16	202.38	163.60	520.68
	U-blox F9P	83.61	256.22	101.49	235.44	829.41
	Xiaomi Mi 8	98.09	33.00	17.77	27.82	205.45
Deep Urban	NovAtel FlexPak 6	97.27	98.20	37.36	90.85	412.88
	U-blox F9P	100.00	6.28	3.57	5.17	66.84
	Xiaomi Mi 8	100.00	24.22	16.72	17.53	179.39
Harsh Urban	NovAtel FlexPak 6	98.01	261.95	131.77	226.43	1210.86
	U-blox F9P	99.02	32.56	19.15	26.34	223.47
	Xiaomi Mi 8	99.44	47.39	28.41	37.94	209.25

Note: Max: maximum error; STD: standard deviation of the positioning error.

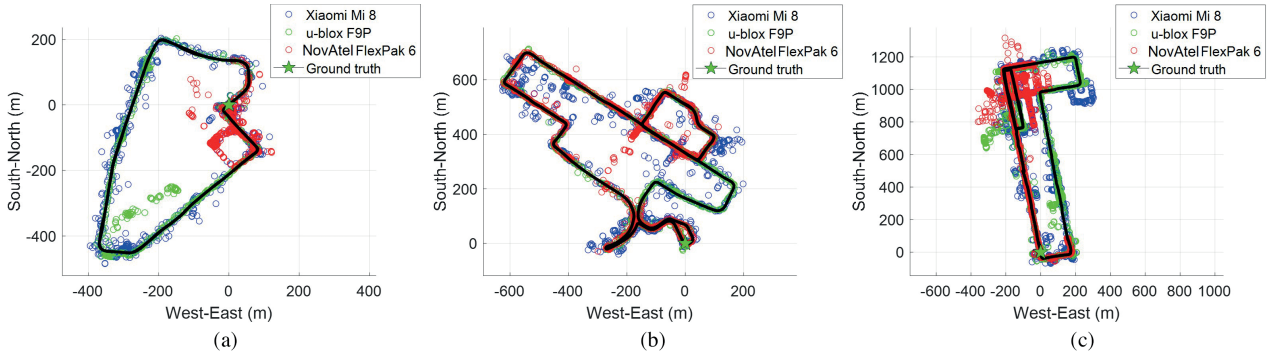


FIGURE 10 Illustration of the 3DMA GNSS (Ng et al., 2021) positioning results for a (a) middle-class urban canyon, (b) deep urban canyon, and (c) harsh urban canyon

was recently verified in the Google Smartphone Challenge (Zangenehnejad & Gao, 2021) organized by Google at ION GNSS+ 2021. A team from Japan (Suzuki, 2021) achieved first place with an algorithm based on FGO. Inspired by the advantages of FGO, our recent work (Wen & Hsu, 2021) presented the development of an FGO-based GNSS positioning framework, GraphGNSSLib, which is provided as an open-source resource to the community. A flowchart of this framework is shown in Figure 11. Multiple epochs of the pseudorange measurements from satellites are used to formulate pseudorange factors, and the Doppler velocity is estimated from a WLS solution. Then, multiple epochs of the estimated Doppler velocity are used to formulate the velocity factors. Finally, a nonlinear optimization problem is formed, and the batch of states χ can be optimized using a nonlinear least squares (NLS) approach.

The resulting performance of the FGO-based GNSS pseudorange/Doppler measurement fusion with the *UrbanNav* dataset based on GraphGNSSLib is shown in Table 6, and the trajectories are presented in Figure 12. Overall, it can be seen that both the availability (100%) and positioning accuracy are improved compared with the conventional WLS. However, the achieved RMSE can still reach more than 10 m because of the numerous expected multipath and NLOS receptions. For more technical details of the GNSS FGO method evaluated here, readers are referred to our previous work (Wen & Hsu, 2021).

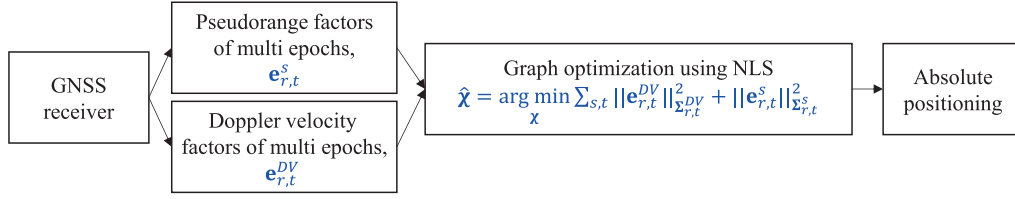


FIGURE 11 Flowchart of a GNSS FGO method, GraphGNSSLib

χ denotes a set of state vectors (position, velocity, and time) in the time window t , and s denotes the index of a satellite. The cap $\hat{\chi}$ denotes the optimized result. The graph is solved by an NLS solver.

TABLE 6
GNSS Pseudorange/Doppler Fusion by FGO

Dataset	Receiver	Availability (%)	RMSE (m)	Mean (m)	STD (m)	Max (m)
Middle-Class Urban	NovAtel FlexPak 6	96.88	19.69	6.15	18.72	280.76
	U-blox F9P	99.74	7.997	6.34	4.86	35.21
	Xiaomi Mi 8	99.68	23.70	19.56	13.39	120.20
Deep Urban	NovAtel FlexPak 6	98.05	12.57	7.71	9.93	72.23
	U-blox F9P	100.00	16.43	13.04	10.00	77.56
	Xiaomi Mi 8	99.78	41.66	30.57	28.32	249.30
Harsh Urban	NovAtel FlexPak 6	89.06	49.74	20.50	45.35	292.65
	U-blox F9P	83.59	8.36	7.02	4.55	32.77
	Xiaomi Mi 8	100.00	17.05	13.94	9.83	78.84

Note: Max: maximum error; STD: standard deviation of the positioning error.

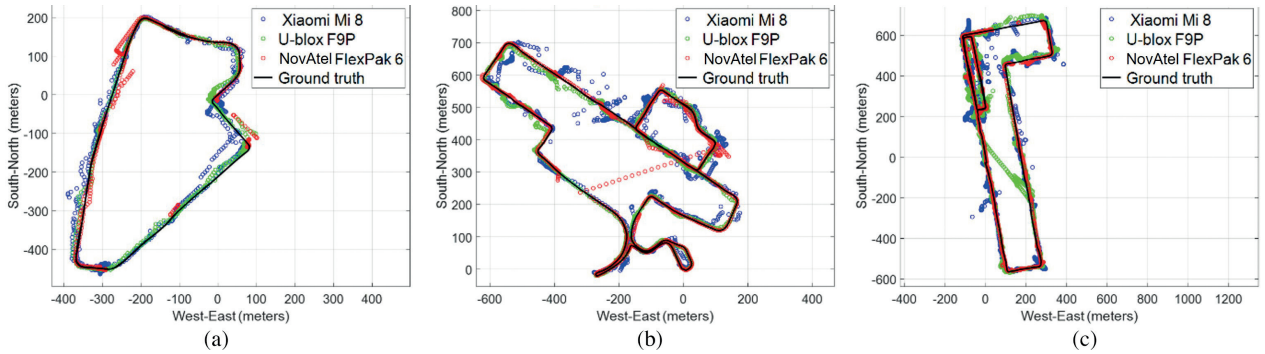


FIGURE 12 Illustration of GNSS FGO positioning results based on GraphGNSSLib for (a) middle-class urban canyon, (b) deep urban canyon, and (c) harsh urban canyon

5.4 | Lidar Odometry and Lidar Inertial Odometry

This section presents the baseline performance achieved for lidar odometry using the popular lidar odometry and mapping (LOAM) algorithm (Zhang & Singh, 2017) and lidar inertial odometry via smoothing and mapping (LIO-SAM) algorithm (Shan et al., 2020). A flowchart of LOAM is shown in Figure 13. The edge and planar features are detected based on feature extraction algorithms. Then, the rough odometry can be estimated by comparing the features detected in the previous and current epochs of data. This rough transformation matrix is used as an initial guess for multi-epoch global optimization. This graph can be solved by NLS,

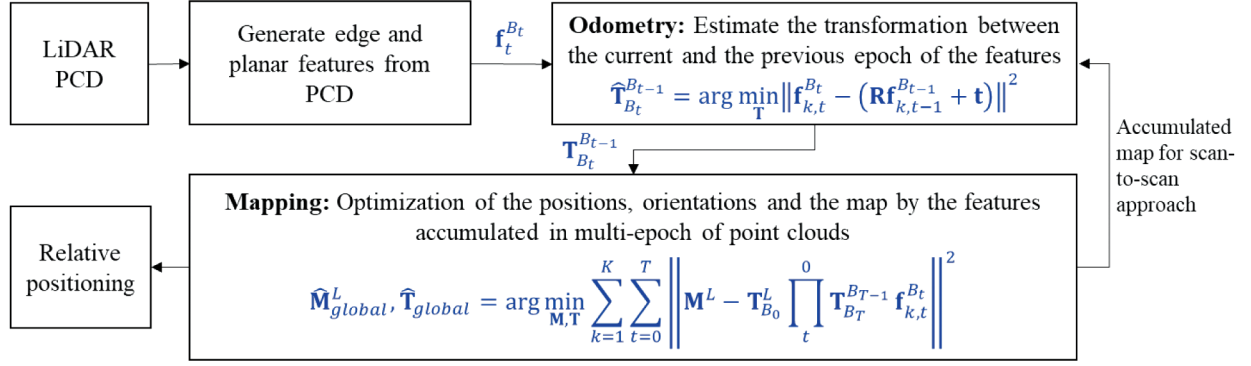


FIGURE 13 Flowchart of a general LOAM method

$f_{k,t}^{B_t}$ denotes the k -th features detected at the t -th epoch and in the lidar body frame, B_t .

$T = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix}$, where $\mathbf{R}$ and $\mathbf{t}$ are rotation and translation matrices, respectively. M^L indicates that the point cloud data (PCD) map is a local frame (usually, the lidar body frame at the first epoch). The cap ^ denotes the optimized result.

and once the map is built or updated, the accumulated map will be used in the next epoch of rough odometry. This method is called the scan-to-map approach, and it may provide a more accurate initial guess for global optimization than conventional methods. In the integration of INS with lidar odometry, the high-frequency INS update is used to provide a good initial guess for the odometry. Thus, a flowchart of the LIO-SAM algorithm is omitted.

The relative positioning errors (RPEs) of the LOAM and LIO-SAM are evaluated using the EVO toolkit (Grupp, 2017), which is a popular evaluation tool in the SLAM research field. A benchmark comparison of publicly available lidar odometry in urban canyons was demonstrated in our previous work (Huang et al., 2022), which demonstrated the computational efficiency and satisfactory positioning accuracy of LOAM. Interestingly, as shown in Table 7, the best lidar odometry performance was achieved in harsh urban scenarios using the center lidar, with an RMSE of 0.17 m (Velodyne 32E). With the help of lidar inertial odometry, the RMSE was further reduced from 0.17 m (LOAM) to 0.15 m (LIO-SAM). The trajectories are shown in Figure 14. A decrease in positioning error is observed for the slant lidars (left and right) for the middle-class urban to harsh urban scenarios, as shown in Table 7. One possible reason for this result is that the higher buildings in the harsh urban scenario can provide dense features for the slant lidar units (the slant lidar unit primarily scans high-rise features), thus leading to improved geometry constraints. Similar to the center lidar, the RMSE of the right lidar decreased from 0.88 m (LOAM) to 0.54 m (LIO-SAM) in the middle-class urban scenario. However, lidar inertial odometry of the slant lidars failed in the deep urban and harsh urban scenarios because of the incorrect estimation of inertial sensor bias with increasing positioning error. Additionally, lidar odometry and lidar inertial odometry failed for all lidar units in the tunnel dataset, as the features in the tunnel scene are quite limited. Determining how to accurately perform mapping in tunnels using onboard sensors remains a challenging problem. Overall, the performance comparisons reveal that the center 32-channel 3D lidar achieves the best performance among the three lidar sensors using LOAM. A similar conclusion can also be obtained for lidar inertial integration using LIO-SAM. For more technical details of the lidar odometry evaluated in urban canyons, readers are referred to our previous work (Huang et al., 2022).

TABLE 7

RPE for Lidar Odometry (Zhang & Singh, 2017) and Lidar Inertial Odometry (Shan et al., 2020) Algorithms

The best performance is highlighted in **bold**. The entry is marked as a fail if the method could not run the full trajectory in the corresponding dataset.

<i>Dataset</i>	<i>Sensor</i>	<i>Method</i>	<i>RMSE (m)</i>	<i>Mean (m)</i>	<i>STD (m)</i>	<i>Max (m)</i>
Middle-Class Urban	Lidar (center)	LOAM	0.35	0.17	0.30	5.33
	Lidar (center) + IMU	LIO-SAM	0.19	0.14	0.13	1.90
	Lidar (left)	LOAM	3.95	2.07	3.36	18.88
	Lidar (left) + IMU	LIO-SAM	Fail	Fail	Fail	Fail
	Lidar (right)	LOAM	0.88	0.31	0.82	6.88
	Lidar (right) + IMU	LIO-SAM	0.54	0.33	0.42	2.88
Deep Urban	Lidar (center)	LOAM	0.80	0.17	0.78	9.98
	Lidar (center) + IMU	LIO-SAM	0.23	0.08	0.22	7.41
	Lidar (left)	LOAM	2.26	0.95	2.05	14.72
	Lidar (left) + IMU	LIO-SAM	Fail	Fail	Fail	Fail
	Lidar (right)	LOAM	0.82	0.26	0.78	11.73
	Lidar (right) + IMU	LIO-SAM	Fail	Fail	Fail	Fail
Harsh Urban	Lidar (center)	LOAM	0.17	0.07	0.16	3.09
	Lidar (center) + IMU	LIO-SAM	0.15	0.07	0.14	2.19
	Lidar (left)	LOAM	1.27	0.30	1.24	10.64
	Lidar (left) + IMU	LIO-SAM	Fail	Fail	Fail	Fail
	Lidar (right)	LOAM	0.40	0.10	0.39	8.19
	Lidar (right) + IMU	LIO-SAM	Fail	Fail	Fail	Fail

Note: Max: maximum error; STD: standard deviation of the positioning error.

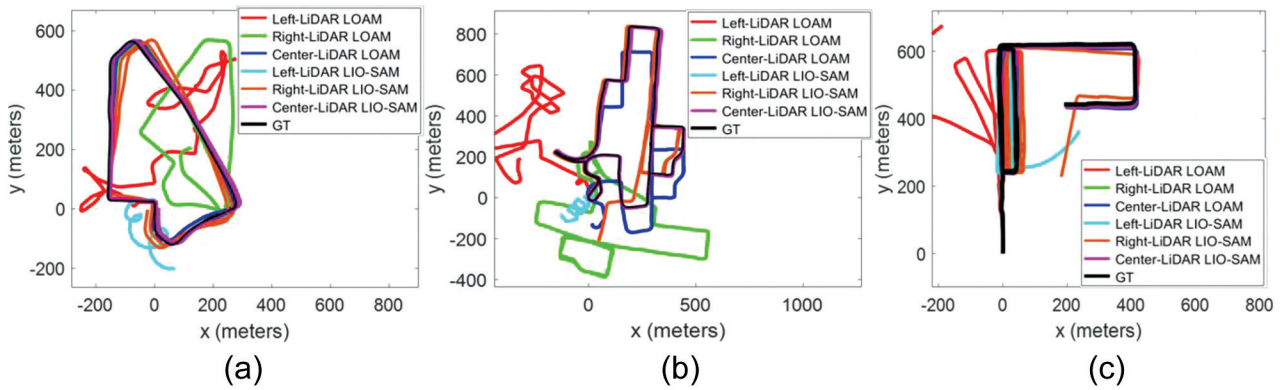


FIGURE 14 Illustration of lidar odometry results based on LOAM and LIO-SAM for a (a) middle-class urban canyon, (b) deep urban canyon, and (c) harsh urban canyon

Note that the LOAM algorithm fails in the tunnel dataset because of the insufficiency of lidar features. Only part of the failed trajectory of the side lidar units using LIO-SAM (see Table 7) is plotted due to the large drift after the estimation failure.

5.5 | VINS and ORB-SLAM3

This section presents the baseline performance of a VINS using the popular VINS-Mono and VINS-Fusion pipeline (Qin et al., 2018), which integrates the camera and IMU measurements by using a sliding-window FGO. Different from the previously mentioned VINS framework, ORB-SLAM3 is also employed to evaluate

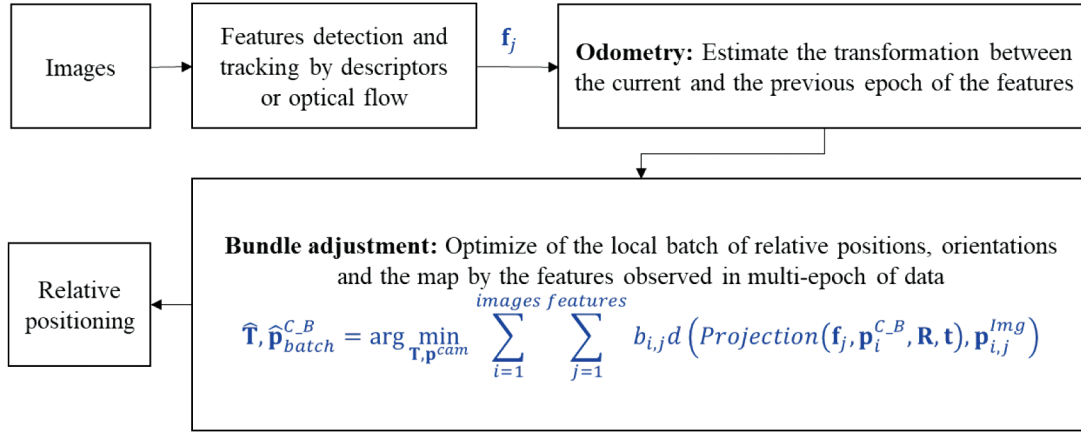


FIGURE 15 Flowchart of a general visual SLAM method that considers bundle adjustment i and j are indices for images and features, respectively. $\mathbf{p}_i^{C-B}$ denotes the camera position and orientation at the camera body frame when image i is taken. $\mathbf{p}_{i,j}^{img}$ denotes the pixel position of feature j projected from image i . $b_{i,j}$ denotes a binary variable. $d()$ denotes the Euclidean distance. $Projection(\mathbf{f}_j, \mathbf{p}_i^{C-B}, \mathbf{R}, \mathbf{t})$ denotes the predicted projection of feature j on image i , considering the rotation $\mathbf{R}$ and translation $\mathbf{t}$ matrices.

the baseline performance; this approach can be applied in pure visual or visual-inertial modes with monocular, stereo, or red green blue-depth (RGB-D) sensors. A flowchart of a typical visual SLAM using the ORB descriptor is shown in Figure 15. The ORB descriptor is used to represent distinctive features, and details regarding this descriptor can be found in Rublee et al. (2011). Similar to the lidar method, the rough transformation matrix is then used as an initial guess for multi-epoch global optimization. In visual SLAM, global optimization is well-known as a bundle adjustment. This approach considers a feature that can be mutually observed in different images. If feature i is visible in image j , then, $b_{i,j} = 1$; otherwise, $b_{i,j} = 0$. Thus, the pairs are identified. By re-projecting feature j on image i , considering the rotation $\mathbf{R}$ and translation $\mathbf{t}$ matrices, the predicted pixel position can be found. Then, the transformation matrix that minimizes the Euclidean distance between the predicted and measured pixel positions of the feature gives the optimized relative positioning. Again, the flowchart of the VINS is omitted.

Similar to lidar odometry, the EVO toolkit (Grupp, 2017) is employed to evaluate the RPE of the VINS. As shown in Table 8, the best performance is achieved in the middle-class urban scenario, with a mean RPE of 0.33 m, based on stereo-inertial integration using VINS-Fusion. Interestingly, ORB-SLAM3 can achieve better performance than VINS-Fusion in the harsh urban and tunnel scenarios. The trajectories are shown in Figure 16 and Figure 17. Unfortunately, the VINS fails in the harsh urban scenario because of the numerous dynamic objects. The VINS also fails in the tunnel dataset, as the texture information in the tunnel is very repetitive. Therefore, further work is needed to improve the robustness of the VINS in a tunnel environment.

5.6 | Discussion on the Complementariness of the Sensors for Multisensory Integration

A qualitative summary of the sensors is given in Table 9. The GNSS sensor is the only one that can provide absolute positioning (i.e., globally referenced solutions), whereas the other sensors can provide only relative positioning. Thus, GNSS is

TABLE 8

RPE of the VINS-Mono and VINS-Fusion (Qin et al., 2018) and ORB-SLAM3 (Campos et al., 2021) Algorithms

The best performance is highlighted in **bold**. Note that the ORB-SLAM3 (stereo), which is marked with *, can only run a short trajectory in the middle-class and deep urban canyon scenarios.

<i>Dataset</i>	<i>Sensor</i>	<i>Method</i>	<i>RMSE (m)</i>	<i>Mean (m)</i>	<i>STD (m)</i>	<i>Max (m)</i>
Middle-Class Urban	Camera (Left of ZED2) + IMU	VINS-Mono	1.14	0.73	0.88	4.72
		VINS-Fusion	0.73	0.48	0.54	2.91
	Stereo (ZED2) + IMU					
	Camera (Left of ZED2) + IMU	ORB-SLAM3	Fail	Fail	Fail	Fail
		ORB-SLAM3	6.55	3.03	5.80	46.45
	Stereo (ZED2) *					
Deep Urban	Camera (Left of ZED2) + IMU	VINS-Mono	1.14	0.68	0.92	15.40
		VINS-Fusion	0.51	0.33	0.38	2.37
	Stereo (ZED2) + IMU					
	Camera (Left of ZED2) + IMU	ORB-SLAM3	1.46	0.82	1.21	21.09
		ORB-SLAM3	2.45	1.29	2.08	6.25
	Stereo (ZED2) *					
Harsh Urban	Camera (Left of ZED2) + IMU	VINS-Mono	Fail	Fail	Fail	Fail
		VINS-Fusion	Fail	Fail	Fail	Fail
	Stereo (ZED2) + IMU					
	Camera (Left of ZED2) + IMU	ORB-SLAM3	1.10	0.53	0.96	8.13
		ORB-SLAM3	Fail	Fail	Fail	Fail
	Stereo (ZED2)					
Tunnel	Camera (Left of ZED2) + IMU	VINS-Mono	Fail	Fail	Fail	Fail
		VINS-Fusion	Fail	Fail	Fail	Fail
	Stereo (ZED2) + IMU					
	Camera (Left of ZED2) + IMU	ORB-SLAM3	6.05	3.47	4.96	12.12
		ORB-SLAM3	Fail	Fail	Fail	Fail
	Stereo (ZED2)					

Note: Max: maximum error; STD: standard deviation of the positioning error.

indispensable in urban areas if no other mapping database is used for other sensors. INS can always provide stable relative positioning with a drifting rate. If the temperature factor is ignored, INS is not affected by the surrounding environment. Consequently, INS is a perfect choice for sensor fusion. In middle-class and deep urban scenarios, most of the solutions (sensor+algorithm) have the potential to provide satisfactory performance. Only the visual and lidar odometry failed at certain epochs when small features were detected. This problem may be overcome if a sophisticated INS integration algorithm is developed. In the harsh urban scenario, only lidar can provide satisfactory performance while the GNSS sensor and camera are strongly hindered by the complexity of the environment. Because of this difference, level-4 autonomous driving employs lidar matching as the primary localization approach. In the tunnel, the GNSS, camera, and lidar sensors are unsatisfactory. The GNSS and lidar sensors are nearly prevented from outputting a solution, as the former and latter sensors are challenged by complete signal blockage and repeated features, respectively. The camera has a small likelihood of performing in the tunnel, but many calibrations using location-known features are required. Thus, to achieve seamless positioning in urban areas, the integration of all sensors is required. In this case, tunnels and indoor areas may require the use of additional sensing technologies.

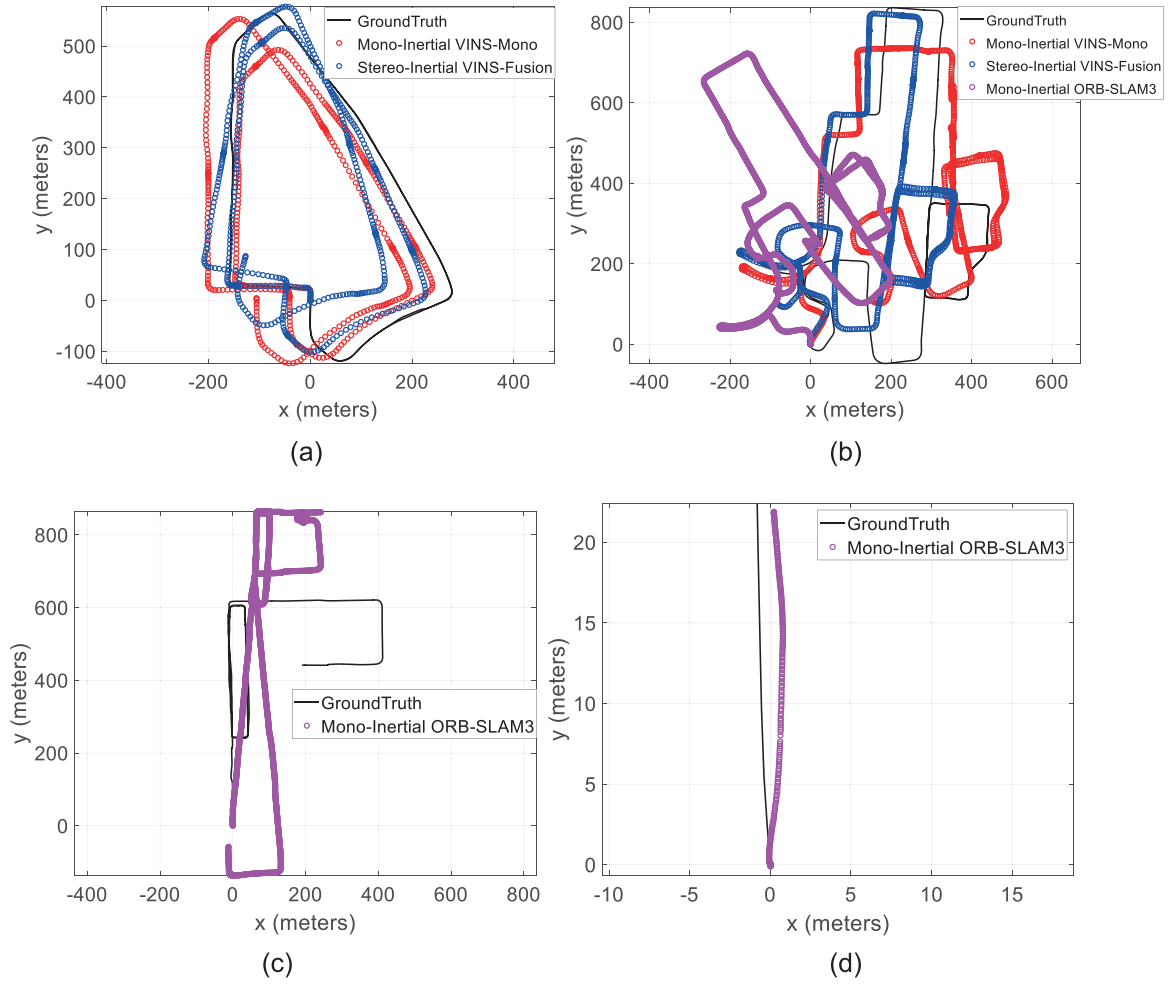


FIGURE 16 Trajectories of VINS-Mono, VINS-Fusion, and ORB-SLAM3

The black curve (trajectory) shows the ground truth positioning given by the NovAtel SPAN-CPT for a (a) middle-class urban canyon, (b) deep urban canyon, (c) harsh urban canyon, and (d) tunnel. Note that VINS-Mono and VINS-Fusion fail in the harsh urban canyon and tunnel scenarios.

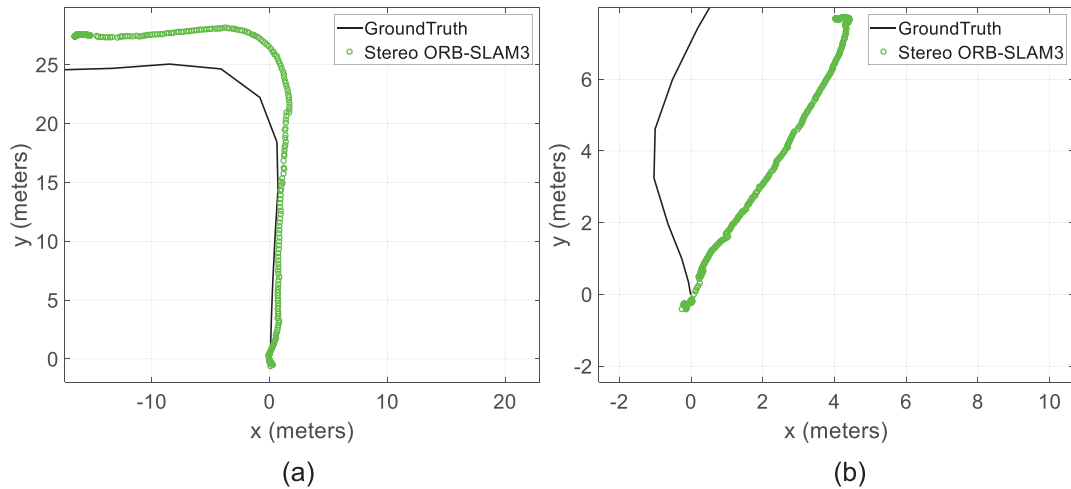


FIGURE 17 Trajectories of ORB-SLAM3 (stereo)

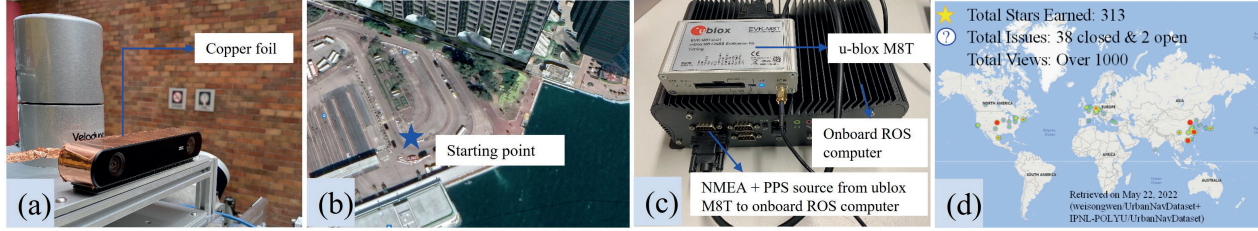
The black curve (trajectory) shows the ground truth positioning given by the NovAtel SPAN-CPT for a (a) middle-class urban canyon and (b) deep urban canyon. Note that the ORB-SLAM3 (stereo) can only run a short trajectory in middle-class and deep urban canyon scenarios.

TABLE 9

Qualitative Comparison of Sensors

√, ☺, ☹, ⊖, and ✖ denote available, satisfactory, normal, unsatisfactory, and unavailable, respectively.

Sensor Technology	Absolute Positioning	Relative Positioning	Middle-Class Urban	Deep Urban	Harsh Urban	Tunnel
GNSS	√	√	☹	☺	☹	✖
Lidar		√	☹	☺	☺	✖
Camera		√	☺	☺	☹	☹
INS		√	☹	☺	☺	☺

FIGURE 18 Illustration of the lessons learned in collecting the *UrbanNav* dataset

(a) Copper foil added around the case and cable to prevent interference, (b) the starting point for data collection in the open sky, (c) time synchronization between ROS time from the desktop computer and GPS time, (d) the state of the *UrbanNav* GitHub page (updated on May 22, 2022). We visualized the stars from GitHub users based on their location via a heatmap. The red spots have a higher demand from researchers.

6 | LESSONS LEARNED

The *UrbanNav* dataset is the accumulation of work performed at the Intelligent Positioning and Navigation Laboratory over the past five years, in the Department of Aeronautical and Aviation Engineering, Hong Kong Polytechnic University. After completing this dataset, we have identified several lessons worth sharing:

- **Sensor interference:** During data collection, we encountered several cases of sensor interference, where the surrounding environment interfered with the camera images, leading to blurred images. Moreover, some GNSS receivers were affected by electromagnetic interference from the camera. To solve this problem, we blocked the signal transmission wires/cables of the camera with tinfoil to prevent unexpected interference from the environment, as shown in Figure 18(a).
- **Ground truth acquisition in GNSS-challenged environments:** Even when using the high-accuracy NovAtel SPAN-CPT, centimeter-level accuracy still cannot be guaranteed when the data collection vehicle stays in a GNSS-denied scenario, such as a tunnel or dense urban canyon, for a long period. It is important to initialize the data collection system in an open area in which the INS bias can be easily calibrated with the help of fixed GNSS-RTK solutions. In the *UrbanNav* dataset, we start the data collection in a harbor side with satisfactory satellite visibility, as shown in Figure 18(b).
- **Time synchronization of multiple sensors:** Synchronizing the time systems of different sensors is of great importance for integrated navigation systems. The smaller offset in timestamps associated with sensor measurements can

lead to a large error in the navigation algorithm, especially in high-speed operations. We adopted the PPS (Niu et al., 2015) signal from a GNSS receiver to synchronize the ROS time from the desktop computer with the GPS time, as shown in Figure 18(c). In other words, all sensor measurements can be aligned to the GPS time with a time synchronization error of less than 1 ms (Vyskocil & Sebesta, 2009).

- **Update and maintenance of the dataset:** Feedback from both academic and industry users can help to continuously improve the completeness of the dataset. Thus, we constructed a GitHub page (initially at <https://github.com/weisongwen/UrbanNavDataset>, recently updated to <https://github.com/IPNL-POLYU/UrbanNavDataset>), as a bridge between our group and the users. We have currently obtained more than 300 stars on GitHub in total. The latest GitHub statistics are presented in Figure 18(d). Different new issues have been raised by users and have been continuously solved by our group through the GitHub pages.

7 | CONCLUSIONS

An accurate and reliable positioning solution for urban scenarios is significant for systems with navigation requirements. The lack of certifiable positioning service presents a bottleneck that prevents the arrival of fully autonomous systems in urban scenarios. In contrast to open areas with good sky visibility and abundant static environment features, urban scenarios introduce additional challenges to existing navigation algorithms. Therefore, an integrated multi-sensor dataset can effectively benefit the navigation research community, but the existing datasets do not satisfy the necessary requirements. This paper introduces the Hong Kong *UrbanNav* dataset to bridge this gap. We wish to raise the awareness of the research community on the urgency of accurate and reliable urban navigation. Finally, we will engage in further work on the *UrbanNav* dataset, including the following:

- Build a website to allow researchers to upload their papers and results evaluated based on the open-source data in terms of proposed criteria.
- Invite experts in the field to design assessment criteria for different positioning algorithms. Report the performance of state-of-the-art positioning and integration algorithms in urban canyons every two years.

ACKNOWLEDGMENTS

The authors appreciate financial support from the Faculty of Engineering, Hong Kong Polytechnic University under the project “Perception-based GNSS PPP-RTK/LVINS integrated navigation system for unmanned autonomous systems operating in urban canyons” (grant number ZVZ8). The authors also thank Dr. Taro Suzuki from the Chiba Institute of Technology, Japan, who provided the algorithm to obtain the sky-pointing fisheye image from Google Earth Studio. Lastly, we thank Jiachen Zhang, Yin-Chiu Kan, Weichang Xu, and Song Yang for their help in developing and evaluating the sensor platform.

REFERENCES

- Agarwal, S., Vora, A., Pandey, G., Williams, W., Kourous, H., & McBride, J. (2020). Ford multi-AV seasonal dataset. *The International Journal of Robotics Research*, 39(12), 1367–1376. <https://doi.org/10.1177/0278364920961451>

- Bai, X., Wen, W., & Hsu, L.-T. (2020). Robust visual-inertial integrated navigation system aided by online sensor model adaption for autonomous ground vehicles in urban areas. *Remote Sensing*, 12(10), 1686. <https://doi.org/10.3390/rs12101686>
- Bai, Z., Jiang, G., & Xu, A. (2020). LiDAR-camera calibration using line correspondences. *Sensors*, 20(21), 6319. <https://doi.org/10.3390/s20216319>
- Bloesch, M., Omari, S., Hutter, M., & Siegwart, R. (2015). Robust visual inertial odometry using a direct EKF-based approach. *Proc. of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany. 298–304. <https://doi.org/10.1109/IROS.2015.7353389>
- Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., & Beijbom, O. (2020). nuScenes: A multimodal dataset for autonomous driving. *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, WA. 11621–11631. <https://doi.org/10.1109/CVPR42600.2020.01164>
- Campos, C., Elvira, R., Rodríguez, J. J. G., Montiel, J. M., & Tardós, J. D. (2021). ORB-SLAM3: An accurate open-source library for visual, visual-inertial, and multimap SLAM. *IEEE Transactions on Robotics*, 37(6), 1874–1890. <https://doi.org/10.1109/TRO.2021.3075644>
- Carlevaris-Bianco, N., Ushani, A. K., & Eustice, R. M. (2016). University of Michigan North Campus long-term vision and lidar dataset. *The International Journal of Robotics Research*, 35(9), 1023–1035. <https://doi.org/10.1177/0278364915614638>
- Das, S., Watson, R., & Gross, J. (2021). Review of factor graphs for robust GNSS applications. *arXiv preprint arXiv:2112.07794*. <https://arxiv.org/abs/2112.07794>
- El-Sheimy, N., Hou, H., & Niu, X. (2007). Analysis and modeling of inertial sensors using Allan variance. *IEEE Transactions on Instrumentation and Measurement*, 57(1), 140–149. <https://doi.org/10.1109/TIM.2007.908635>
- Furgale, P., Rehder, J., & Siegwart, R. (2013). Unified temporal and spatial calibration for multi-sensor systems. *Proc. of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Tokyo, Japan. 1280–1286. <https://doi.org/10.1109/IROS.2013.6696514>
- Geiger, A., Lenz, P., & Urtasun, R. (2012). Are we ready for autonomous driving? The kitti vision benchmark suite. *Proc. of the 2012 IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI. 3354–3361. <https://doi.org/10.1109/CVPR.2012.6248074>
- Geneva, P., Eckenhoff, K., Lee, W., Yang, Y., & Huang, G. (2020). OpenVINS: A research platform for visual-inertial estimation. *Proc. of the 2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France. 4666–4672. <https://doi.org/10.1109/ICRA40945.2020.9196524>
- Geyer, J., Kassahun, Y., Mahmudi, M., Ricou, X., Durgesh, R., Chung, A. S., Hauswald, L., Pham, V. H., Mühlegg, M., & Dorn, S. (2020). A2D2: Audi autonomous driving dataset. *arXiv preprint arXiv:2004.06320*. <https://arxiv.org/abs/2004.06320>
- Groves, P. (2016). It's time for 3D mapping-aided GNSS. *Inside GNSS Magazine*, 50–56. https://discovery.ucl.ac.uk/id/eprint/1542357/1/Groves_Groves_InsideGNSS_3DMA%201_0.pdf
- Groves, P. D., Jiang, Z., Rudi, M., & Strode, P. (2013). A portfolio approach to NLOS and multipath mitigation in dense urban areas. *Proc. of the 26th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2013)*, Nashville, TN. 3231–3247. <https://www.ion.org/publications/abstract.cfm?articleID=11264>
- Grupp, M. (2017). evo: Python package for the evaluation of odometry and slam. <https://github.com/MichaelGrupp/evo>
- Hsu, L.-T. (2018). Analysis and modeling GPS NLOS effect in highly urbanized area. *GPS Solutions*, 22(1), 1–12. <https://doi.org/10.1007/s10291-017-0667-9>
- Hsu, L.-T., Kubo, N., Wen, W., Chen, W., Liu, Z., Suzuki, T., & Meguro, J. (2021). UrbanNav: An open-sourced multisensory dataset for benchmarking positioning algorithms designed for urban areas. *Proc. of the 34th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2021)*, St. Louis, MO. 226–256. <https://doi.org/10.33012/2021.17895>
- Huang, A. S., Antone, M., Olson, E., Fletcher, L., Moore, D., Teller, S., & Leonard, J. (2010). A high-rate, heterogeneous dataset from the darpa urban challenge. *The International Journal of Robotics Research*, 29(13), 1595–1601. <https://doi.org/10.1177/0278364910384295>
- Huang, F., Wen, W., Zhang, J., & Hsu, L. T. (2022). Point wise or feature wise? A benchmark comparison of publicly available lidar odometry algorithms in urban canyons. *IEEE Intelligent Transportation Systems Magazine*, 2–20. <https://doi.org/10.1109/IMITS.2021.3092731>
- Huang, X., Cheng, X., Geng, Q., Cao, B., Zhou, D., Wang, P., Lin, Y., & Yang, R. (2018). The ApolloScape dataset for autonomous driving. *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (IEEE/CVF 2018)*, Salt Lake City, UT. 954–960. <https://doi.org/10.1109/CVPRW.2018.00141>
- Kato, S., Kitamura, M., Suzuki, T., & Amano, Y. (2016). NLOS satellite detection using a fish-eye camera for improving GNSS positioning accuracy in urban area. *Journal of Robotics and Mechatronics*, 28(1), 31–39. <https://doi.org/10.20965/jrm.2016.p0031>
- Kato, S., Tokunaga, S., Maruyama, Y., Maeda, S., Hirabayashi, M., Kitsukawa, Y., Monrroy, A., Ando, T., Fujii, Y., & Azumi, T. (2018). Autoware on board: Enabling autonomous vehicles with embedded systems. *Proc. of the 2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPs)*, Porto, Portugal. 287–296. <https://doi.org/10.1109/ICCPs.2018.00035>

- Kennedy, S., Hamilton, J., & Martell, H. (2006). Architecture and system performance of SPAN-NovAtel's GPS/INS solution. *Proc. of the Position, Location, and Navigation Symposium (IEEE/ION PLANS 2006)*, Coronado, CA. 266–274. <https://doi.org/10.1109/PLANS.2006.1650612>
- Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R., & Furgale, P. (2015). Keyframe-based visual-inertial odometry using nonlinear optimization. *The International Journal of Robotics Research*, 34(3), 314–334. <https://doi.org/10.1177/0278364914554813>
- Levinson, J., Askeland, J., Becker, J., Dolson, J., Held, D., Kammel, S., Kolter, J. Z., Langer, D., Pink, O., & Pratt, V. (2011). Towards fully autonomous driving: Systems and algorithms. *Proc. of the 2011 IEEE Intelligent Vehicles Symposium (IV)*, Baden-Baden, Germany. 163–168. <https://doi.org/10.1109/IVS.2011.5940562>
- Lv, J., Xu, J., Hu, K., Liu, Y., & Zuo, X. (2020). Targetless calibration of LiDAR-IMU system based on continuous-time batch estimation. *Proc. of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV. 9968–9975. <https://doi.org/10.1109/IROS45743.2020.9341405>
- Maddern, W., Pascoe, G., Linegar, C., & Newman, P. (2017). 1 year, 1000 km: The Oxford RobotCar dataset. *The International Journal of Robotics Research*, 36(1), 3–15. <https://doi.org/10.1177/0278364916679498>
- Meguro, J.-i., Murata, T., Takiguchi, J.-i., Amano, Y., & Hashizume, T. (2009). GPS multipath mitigation for urban area using omnidirectional infrared camera. *IEEE Transactions on Intelligent Transportation Systems*, 10(1), 22–30. <https://doi.org/10.1109/TITS.2008.2011688>
- Mourikis, A. I., & Roumeliotis, S. I. (2007). A multi-state constraint Kalman filter for vision-aided inertial navigation. *Proc. of the 2007 IEEE International Conference on Robotics and Automation*, Rome, Italy. 3565–3572. <https://doi.org/10.1109/ROBOT.2007.364024>
- Ng, H.-F., Zhang, G., & Hsu, L.-T. (2020). A computation effective range-based 3D mapping aided GNSS with NLOS correction method. *The Journal of Navigation*, 73(6), 1202–1222. <https://doi.org/10.1017/S037346332000003X>
- Ng, H.-F., Zhang, G., Luo, Y., & Hsu, L.-T. (2021). Urban positioning: 3D mapping-aided GNSS using dual-frequency pseudorange measurements from smartphones. *NAVIGATION*, 68(4), 727–749. <https://doi.org/10.1002/navi.448>
- Niu, X., Yan, K., Zhang, T., Zhang, Q., Zhang, H., & Liu, J. (2015). Quality evaluation of the pulse per second (PPS) signals from commercial GNSS receivers. *GPS Solutions*, 19(1), 141–150. <https://doi.org/10.1007/s10291-014-0375-7>
- Pandey, G., McBride, J. R., & Eustice, R. M. (2011). Ford campus vision and lidar dataset. *The International Journal of Robotics Research*, 30(13), 1543–1552. <https://doi.org/10.1177/0278364911400640>
- Qin, T., Li, P., & Shen, S. (2018). VINS-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics*, 34(4), 1004–1020. <https://doi.org/10.1109/TRO.2018.2853729>
- Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., & Ng, A. Y. (2009). ROS: An open-source robot operating system. In *ICRA Workshop on Open Source Software* (Vol. 3, No. 3.2, p. 5). http://lars.mec.ua.pt/public/LAR%20Projects/BinPicking/2016_RodrigoSalgueiro/LIB/ROS/icraoss09-ROS.pdf
- Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. *Proc. of the 2011 International Conference on Computer Vision*, Barcelona, Spain. 2564–2571. <https://doi.org/10.1109/ICCV.2011.6126544>
- Shan, T., & Englot, B. (2018). LeGO-LOAM: Lightweight and ground-optimized lidar odometry and mapping on variable terrain. *Proc. of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain. 2758–2765. <https://doi.org/10.1109/IROS.2018.8594299>
- Shan, T., Englot, B., Meyers, D., Wang, W., Ratti, C., & Rus, D. (2020). Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. *Proc. of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV. 5135–5142. <https://doi.org/10.1109/IROS45743.2020.9341176>
- Smith, M., Baldwin, I., Churchill, W., Paul, R., & Newman, P. (2009). The new college vision and laser dataset. *The International Journal of Robotics Research*, 28(5), 595–599. <https://doi.org/10.1177/0278364909103911>
- Soares, E., Brandão, P., & Prior, R. (2020). Analysis of timekeeping in experimentation. *Proc of the 2020 12th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, Porto, Portugal. 1–6. <https://doi.org/10.1109/CSNDSP49049.2020.9249632>
- Sun, P., Kretschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., & Caine, B. (2020). Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA. 2446–2454. <https://doi.org/10.1109/CVPR42600.2020.00252>
- Suzuki, T. (2021). First place award winner of the smartphone decimeter challenge: Global optimization of position and velocity by factor graph optimization. *Proc. of the 34th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2021)*, St. Louis, MO. 2974–2985. <https://doi.org/10.33012/2021.18109>

- Suzuki, T., & Kubo, N. (2015). Simulation of GNSS satellite availability in urban environments using Google Earth. *Proc. of the ION 2015 Pacific PNT Meeting*, Honolulu, HI. 1069–1079. <https://www.ion.org/publications/abstract.cfm?articleID=12782>
- Takasu, T., & Yasuda, A. (2009). Development of the low-cost RTK-GPS receiver with an open source program package RTKLIB. In *International Symposium on GPS/GNSS* (Vol. 1, pp. 1–6). https://gpspp.sakura.ne.jp/paper2005/isgps_2009_rtklib_revA.pdf
- Varma, V., Adarsh, S., Ramachandran, K., & Nair, B. B. (2018). Real time detection of speed hump/bump and distance estimation with deep learning using GPU and ZED stereo camera. *Procedia Computer Science*, 143, 988–997. <https://doi.org/10.1016/j.procs.2018.10.335>
- Veľas, M., Španěl, M., Materna, Z., & Herout, A. (2014). Calibration of RGB camera with velodyne lidar. https://www.fit.vut.cz/research/publication-file/10578/Calibration_of_RGB_Camera_With_Velodyne_LiDAR.pdf
- Vyskocil, P., & Sebesta, J. (2009). Relative timing characteristics of GPS timing modules for time synchronization application. *Proc. of the 2009 International Workshop on Satellite and Space Communications*, Siena, Italy. 230–234. <https://doi.org/10.1109/IWSSC.2009.5286378>
- Wan, G., Yang, X., Cai, R., Li, H., Zhou, Y., Wang, H., & Song, S. (2018). Robust and precise vehicle localization based on multi-sensor fusion in diverse city scenes. *Proc. of the 2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia. 2670–2677. <https://doi.org/10.1109/ICRA.2018.8461224>
- Wanninger, L., & Heßelbarth, A. (2020). GNSS code and carrier phase observations of a Huawei P30 smartphone: Quality assessment and centimeter-accurate positioning. *GPS Solutions*, 24(2), 64. <https://doi.org/10.1007/s10291-020-00978-z>
- Wen, W., Bai, X., Kan, Y. C., & Hsu, L.-T. (2019). Tightly coupled GNSS/INS integration via factor graph and aided by fish-eye camera. *IEEE Transactions on Vehicular Technology*, 68(11), 10651–10662. <https://doi.org/10.1109/TVT.2019.2944680>
- Wen, W., & Hsu, L.-T. (2021). Towards robust GNSS positioning and real-time kinematic using factor graph optimization. *Proc. of the 2021 IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China. 5884–5890. <https://doi.org/10.1109/ICRA48506.2021.9562037>
- Wen, W., Hsu, L.-T., & Zhang, G. (2018). Performance analysis of NDT-based graph SLAM for autonomous vehicle in diverse typical driving scenarios of Hong Kong. *Sensors*, 18(11), 3928. <https://doi.org/10.3390/s18113928>
- Wen, W., Pfeifer, T., Bai, X., & Hsu, L.-T. (2021). Factor graph optimization for GNSS/INS integration: A comparison with the extended Kalman filter. *NAVIGATION*, 68(2), 315–331. <https://doi.org/10.1002/navi.421>
- Wen, W., Zhang, G., & Hsu, L. (2018). Exclusion of GNSS NLOS receptions caused by dynamic objects in heavy traffic urban scenarios using real-time 3D point cloud: An approach without 3D maps. In *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Monterey, CA. 158–165. <https://doi.org/10.1109/PLANS.2018.8373377>
- Wen, W., Zhang, G., & Hsu, L. T. (2019). Correcting NLOS by 3D LiDAR and building height to improve GNSS single point positioning. *NAVIGATION*, 66(4), 705–718. <https://doi.org/10.1002/navi.335>
- Wen, W., Zhou, Y., Zhang, G., Fahandezh-Saadi, S., Bai, X., Zhan, W., Tomizuka, M., & Hsu, L.-T. (2020). UrbanLoco: A full sensor suite dataset for mapping and localization in urban scenes. *Proc. of the 2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France. 2310–2316. <https://doi.org/10.1109/ICRA40945.2020.9196526>
- Yan, Z., Sun, L., Krajník, T., & Ruichek, Y. (2020). EU long-term dataset with multiple sensors for autonomous driving. *Proc. of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV. 10697–10704. <https://doi.org/10.1109/IROS45743.2020.9341406>
- Yin, J., Li, A., Li, T., Yu, W., & Zou, D. (2022). M2DGR: A multi-sensor and multi-scenario SLAM dataset for ground robots. *IEEE Robotics and Automation Letters*, 7(2), 2266–2273. <https://doi.org/10.1109/LRA.2021.3138527>
- Zangenehnejad, F., & Gao, Y. (2021). GNSS smartphones positioning: Advances, challenges, opportunities, and future perspectives. *Satellite Navigation*, 2(1), 1–23. <https://doi.org/10.1186/s43020-021-00054-y>
- Zhang, J., & Singh, S. (2017). Low-drift and real-time lidar odometry and mapping. *Autonomous Robots*, 41(2), 401–416. <https://doi.org/10.1007/s10514-016-9548-2>
- Zhong, Q., & Groves, P. D. (2022). Multi-epoch 3D-mapping-aided positioning using Bayesian filtering techniques. *NAVIGATION*, 69(2). <https://doi.org/10.33012/navi.515>

How to cite this article: Hsu, L.-T., Huang, F., Ng, H.-F., Zhang, G., Zhong, Y., Bai, X., & Wen, W. (2023). Hong Kong UrbanNav: An open-source multisensory dataset for benchmarking urban navigation algorithms. *NAVIGATION*, 70(4). <https://doi.org/10.33012/navi.602>