

# Design of a Reconfigurable Multi-Sensor Testbed for Autonomous Vehicles and Ground Robots

Zheng Gong, Wuyang Xue, Ziang Liu, Yimo Zhao, Ruihang Miao, Rendong Ying and Peilin Liu\*

*School of Electronic Information and Electrical Engineering*

*Shanghai Jiao Tong University*

*Shanghai, China*

email address: \*liupeilin@sjtu.edu.cn

**Abstract**—In the field of autonomous vehicles and robotics research, a well-designed testbed can provide a convenient and safe development environment. In this paper, we propose and implement a reconfigurable testbed equipped with multiple heterogeneous sensors and a compact but powerful local computation unit. The proposed testbed benefits from a self-sustaining design, and can be smoothly reconfigured to different varieties of vehicles. These features make the testbed a novel and ideal platform for testing and verifying navigation, recognition and control algorithms under diverse scenarios.

**Index Terms**—Testbed, Robotic, Autonomous Vehicle, Edge Computing

## I. INTRODUCTION

Autonomous vehicle and ground robot systems are usually a composition of several sub-systems, from sensor hardware and power modules to softwares including control, navigation and human interface, etc.. Performing online tests on those sub-systems individually, within the entire architecture, can be troublesome, or even dangerous. For instance, sensing and navigation modules, whose outputs are directly coupled with the control and power sub-systems, are fundamental parts in autonomous applications. However, during the development stage, testing these modules with the actuator system directly in-loop can cause control-failure and lead to hazardous consequences. Therefore, a testbed that is free from all the risky parts, but as homogeneous as possible from the original system, is one of the preferred solutions.

In this work, we design and implement a reconfigurable multi-sensor testbed, as shown in Fig.1, which presents the features of:

- Reconfigurable on different vehicle carriers and sensors. For disparate applications the carrier vehicle and the applied sensors can be diverse, but they may share the same algorithms for core function module. This situation can be revealed and supported by multiple existing projects, including vehicle-based [1] [2], stationary [3], airborne [4] and robot platforms [5]. Therefore designing a testbed that can be smoothly reconfigured to different vehicles can make the test for different scenarios more feasible.
- Self-sustaining all-in-one setup. Required by the first feature mentioned above, the proposed testbed possesses a characteristic of self-sustaining all-in-one setup. All sensors, along with the on-board processing unit and



Fig. 1: Proposed testbed (central red box) with its different reconfigurable ground vehicles.

human-interface are contained in the testbed. Further, a large capacity battery with a high discharge rate and its related Battery Elimination Circuit (BEC) module are also embedded to serve all the on-board modules with different voltage and current requirements. These efforts make the testbed able to bootstrap and connect to related carrier vehicle through only one communication bus.

- Edge computing compatibility. Thanks to the increasing locally deployed computation power and the advanced communication data-link, our proposed testbed performs a full compatibility with Edge Computing [6] in Internet of Things (IoT) applications. The local onboard processing unit with 4-Gen connection, which is on the edge side of the cloud computation, has the capability of processing all the sensor data on-board, including Vision Simultaneous Localization and Mapping (V-SLAM) and semantic segmentation neuron network.

The rest of this paper is organized as follows: we introduce the entire system architecture and setup in Section II, with the detailed description of equipped sensors and power system in its subsection. Then the software architecture is revealed in Section III, including the data stream and synchronization of sensors. Finally, in Section IV we draw our conclusions and discuss about future work.

## II. SYSTEM DESIGN AND ARCHITECTURE

The top-level design for the proposed system is shown in Fig. 2(a). The system is composed of three major parts: mechanical structure, electronics sub-systems including power and sensors, and the onboard processing unit with its corresponding software.

The system's central onboard processing unit is a compact GPU computing system provided by Vecow®, which is equipped with an Intel® Core™ i7 processor and NVIDIA® GeForce® GTX 1070 MXM graphic module. This computing system provides rich I/O interface resources, including 6 external USB 3.0 ports, 3 RS-232 COM ports, 32 isolated general purpose Digital I/O (DIO) and 3 external SIM card sockets for wide-range data-link.

The mechanical structure is designed under three general guidelines: First, the testing unit should be conveniently transferred from vehicle to vehicle. Therefore, a quick release/mount mechanism is designed. Four pairs of aluminum profiles compatible quick release brackets are applied at the bottom of the testbed as well as the top of the vehicle. Then two aluminum tubes with outer diameter of 22mm and wall thickness of 5mm are in charge of connecting those quick release brackets. This design allows a mount and re-mount time under 5 minutes, which can be carried out by one person.

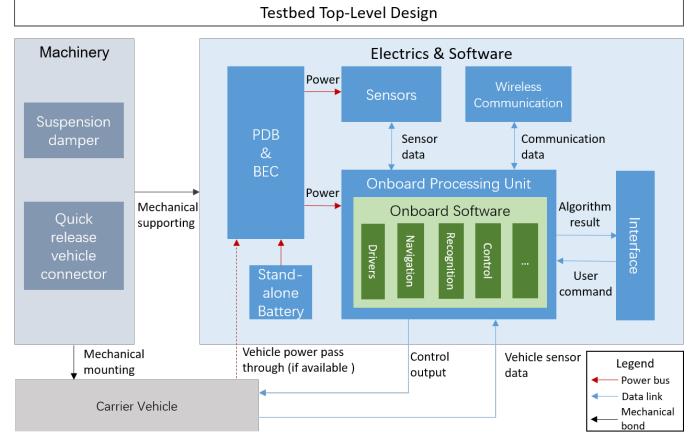
Secondly, as the testbed is designed for working with different sizes of ground vehicles, damping is required. The suspension module can be seen in the bottom of Fig. 2(b). For brevity, however, we cannot discuss it further into the design of the wire suspension damper.

Finally, to ensure we can reconfigure sensors according to different platforms, a multi-layer structure is designed. Each layer holds its own functionality, connecting to each other by aluminum stands. Extension on sensor modalities or updates of other modules can be manipulated by replacing the related plate.

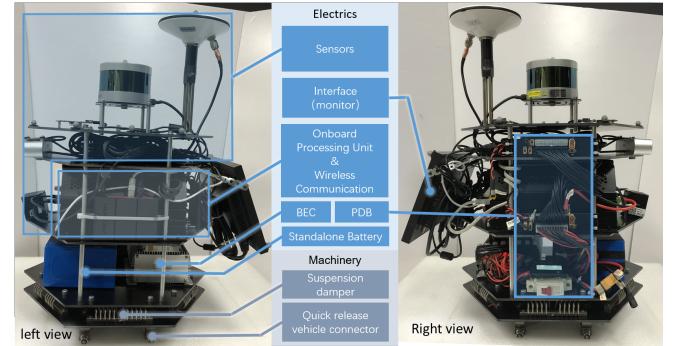
### A. Sensor Setup and Output

The sensor modalities selection plays a crucial role in the adaptability. For this reason, our testbed currently incorporates Lidar, stereo camera, TOF depth camera, GNSS receiver, IMU and microphone array to holistically observe and understand the scene. The full suite of sensors is listed below.

- 1 × **ZED® stereo camera**. The front facing stereo camera is the essential vision sensor for localization and recognition. It is interfaced using a USB 3.0 connection with the host machine with maximum resolution of 2208 × 1242@15fps. When facing high-dynamic scenarios, the fps can be boosted to 100 frame-per-second with WVGA (672 × 376) resolution. The baseline of this stereo camera is 120mm, which guarantees a maximum depth detection range of 20 m. Additionally, we attach a *SparkFun 9DoF Razor IMU* on top of the left camera to accommodate applications such as Vision Inertial Odometry [7]. This IMU module is based on Invensense®'s *MPU9250*, and can measure the ego 3-axis linear acceleration, 3-axis angular velocity and 3-axis



(a) Top-level design diagram



(b) Corresponding physical modules

Fig. 2: Top-level design of the testbed. Noticing that the PDB in power management module stands for Power Deliver Board meanwhile the BEC is the abbreviation for Battery Elimination Circuit.

magnetic compass in frequencies up to 100 Hz. Both of these two modules require a power supply at 5V drawing a current of less than 500mA, so they can be powered directly by the USB port.

- 2 × **SparkFun® 9DoF Razor IMU**. In total, two IMUs are installed on the current testbed. One of them, as introduced in last entry, is attached to the stereo camera for VIO applications. The other one is placed in the center of the testbed.
- 1 × **Data Miracle® SmartToF™ TC-E2 Depth Camera**. One forward-facing Time of Flight (ToF) depth camera is deployed in front of the testbed for obstacle detection and territory mapping. The ToF camera can provide stable but computation-efficient depth estimation among complex lighting environments. Running under a 12V power supply, the camera can output a depth point cloud of 320 × 240@120fps with maximum detection range of 8 meters. The data is transferred through a USB 2.0 port.
- 1 × **uBlox® M8P GNSS Receiver**. A Global Navigation Satellite System (GNSS) receiver is equipped with a disk antenna located on the posterior top surface of the

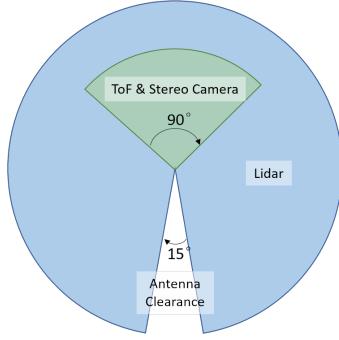


Fig. 3: Exteroceptive sensors field of view

testbed. This module supports a Real Time Kinematic (RTK) differential GNSS solution which can achieve a sub-meter accuracy up to  $10Hz$ . It communicates with the host system through serial port.

- **$1 \times \text{Velodyne}^{\circledR} \text{ VLP-16 Lidar}$ .** For high accuracy localization and mapping under large scale, a 16-line Lidar is installed on the top center of the platform. Working under  $12V$ , this module communicates with the onboard processing unit using an RJ45 Ethernet cable through an 1000M switcher. 16 lines of point cloud with azimuthal resolution of  $0.1^{\circ}$  is reported at  $50Hz$ . Although it can detect a scan echo up to 100 meters with a  $360^{\circ}$  view, the rear 15 degree of view is dismissed because of the GNSS antenna stand, as shown in Fig.3.
- **$1 \times \text{Seeed}^{\circledR} \text{ ReSpeaker Mic Array v2.0}$ .** To expand our testbed's sensing spectrum and interaction interface, a 4-channel microphone array is equipped on front top of the device, which is used for sound source localization and natural language interaction.

#### B. Sensor Calibration

As mentioned above, there are various sensors including GNSS, Lidar, IMU, stereo camera and ToF in our platform. These sensors use different coordinate systems. This leads to ambiguity when using their information directly. Therefore, calibration into an unified coordinate is required. Since the accuracy of the onboard GNSS is much lower than the other sensors, only Lidar, IMU and camera have to be calibrated precisely. The relative coordinate configuration of different sensors is shown in Fig.4.

The calibration between camera and IMU follows the method in [8]. It provides an estimator to calibrate the transformation and temporal offset between camera and IMU. The transformation and rotation error between camera and IMU is  $1.06mm$  and  $0.012$  degree [8].

As for the calibration between Lidar and IMU, the method provided in [9] is applied. It uses a probabilistic formulation to calibrate and localize Lidar-IMU system jointly. It reprojects the 3D-points to the first Lidar frame and minimizes the point-to-plane distance to get the calibration parameters. The transformation and rotation error between Lidar and IMU is about  $0.57mm$  and  $0.016$  degree [9].

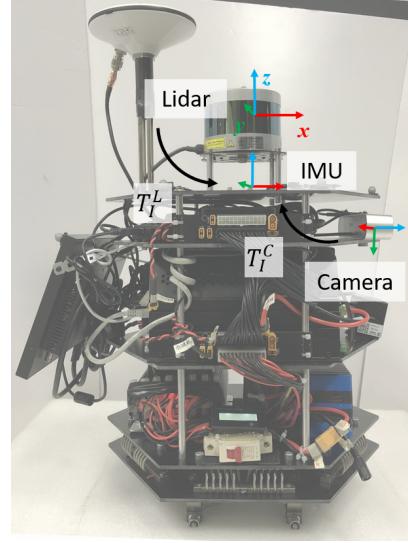


Fig. 4: Sensor frames for ego-motion estimation

#### C. Battery Elimination and Power Distribution

As we strive to implement the proposed testbed as a self-sustaining all-in-one setup and to keep the maximum compatibility for all kinds of sensors, we need a steady, as well as abundant, power supply with multiple voltages. Additionally, in order to satisfy the edge computing compatible level, the onboard processing unit can consume up-to  $400W$ , which also raises the requirement of the power system. Based on these reasons, we design a power supply module, as shown in Fig. 5. The entire module is divided into two parts, the Battery Elimination Circuit (BEC) and the Power Deliver Board (PDB). In BEC, we convert and assign all the needed voltages and currents to the power bus. Here we utilize a 24 pin ATX power connection as the power bus, considering the total current flow on each phase and the total number of voltage phase is required. Then PDBs are placed on the side of each layer, and connected with each other also through ATX cables. The PDBs are in charge of splitting the power bus from ATX ports into individual XT30 and XT60 connectors.

As for the battery choice, we use a 3-series-7-parallel Li-Ion battery on this testbed. This is under the consideration that the 3-series Li-Ion battery has the closest discharge voltage ( $9-12.6V$ ) comparing to the Lead battery set ( $9-14.4V$ ) and vehicular power outlet voltage ( $12V$ ). This battery can keep the maximal compatibility when reconfigured to other platforms.

### III. SOFTWARE ARCHITECTURE

As the main purpose of the testbed is to benchmark different algorithms in navigation, recognition and control area, the software system of the testbed need to provide a easy-to-use, portable interface. To achieve this, we utilize the Robot Operating System (ROS) [10] in our system and the overall architecture is shown in Fig.6. This entire system is built on a desktop release version of Ubuntu 16.04, which includes all the

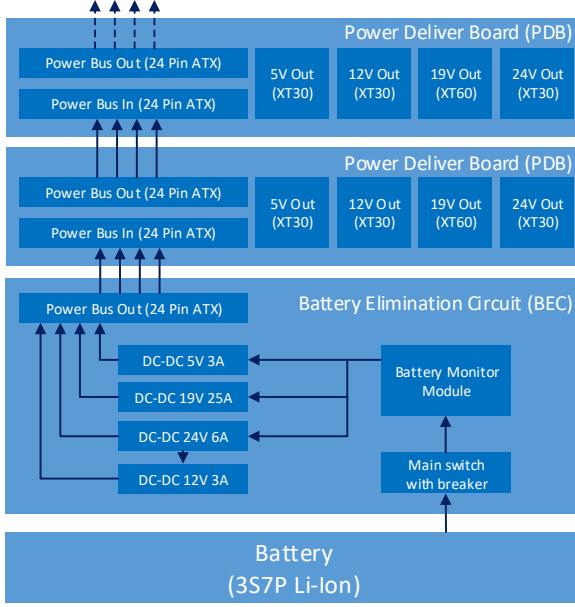


Fig. 5: Power supply module design

drivers for sensors and hardware devices, as well as the high-level processing libraries such as OpenCV and TensorFlow. Then, to uncouple the algorithms from specific drivers and the inter-program communication method, ROS is introduced as a middleware layer between the operating system and the algorithm utilities. The Robot Operating System, strictly speaking, is not an operating system, but a set of software libraries and tools that provides synchronization, inter-program communication through topic publish/subscribe model and data abstraction etc.. This gives users extra flexibility and portability on implementation of algorithms.

Based on this middleware, we wrap all the algorithm supporting programs as ROS-based programs, or ROS nodes. This way, all sensors are abstracted into the related message type through the correspondence node, and all the nodes communicate with each other through the topic publish/subscribe mechanism provided by ROS. Algorithm supporting programs also include the record/replay interface and the visualization tools. Then upon this, an algorithm being designed for benchmark can be platform independent. For instance, a neuron-network based multi-sensor recognition and navigation algorithm can be constructed as follow: It subscribes the needed sensor data including a camera image stream and the Lidar point cloud, calls the related processing libraries like TensorFlow or Caffe, and then publishes the processed result back into ROS topics for other nodes.

#### IV. CONCLUSION AND FUTURE WORKS

In this paper, we address a testbed platform equipped with a wide range of sensors and introduce the system setup as

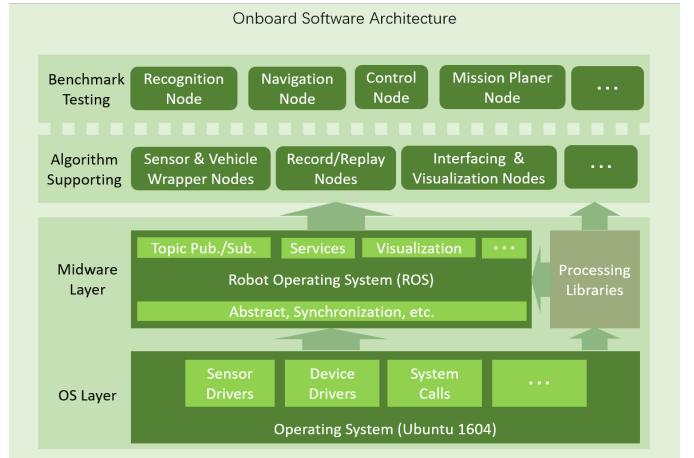


Fig. 6: Onboard software architecture

well as the onboard sensor module list. The capabilities and reconfigurability are presented by the introduction on hardware to software design. Holding the features of self-sustaining and high-performance local computation power, we believe that this testbed can yield an extraordinary effect and become an ideal testing platform for multiple research such as like object detection, ego-motion estimation and autopilot etc.. Novel dataset based on this proposed testbed with various carrier vehicles and scenarios can be expected as an anticipated future work.

#### REFERENCES

- [1] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 3354–3361.
- [2] A. Rangesh, K. Yuen, R. K. Satzoda, R. N. Rajaram, P. Gunaratne, and M. M. Trivedi, "A multimodal, full-surround vehicular testbed for naturalistic studies and benchmarking: Design, calibration and deployment," *CoRR*, vol. abs/1709.07502, 2017.
- [3] D. A. Cooper, R. J. Ledoux, K. Kamieniecki, S. E. Korby, J. Costales, R. Niyazov, D. Hempstead, M. Gallagher, L. Janney, N. D’Olympia, C. Monnier, and R. Wronski, "Advanced algorithm development for detection, tracking, and identification of vehicle-borne radiation sources in a multi-sensor, distributed testbed," in *2015 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, Oct 2015, pp. 1–3.
- [4] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The EuRoC micro aerial vehicle datasets," *Int. J. Rob. Res.*, vol. 35, no. 10, pp. 1157–1163, Sep. 2016.
- [5] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of michigan north campus long-term vision and lidar dataset," *The International Journal of Robotics Research*, vol. 35, no. 9, pp. 1023–1035, 2016.
- [6] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.
- [7] T. Qin, P. Li, and S. Shen, "VINS-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.
- [8] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013, pp. 1280–1286.
- [9] C. Le Gentil, T. Vidal-Calleja, and H. Shoudong, "3d lidar-imu calibration based on upsampled preintegrated measurements for motion distortion correction," in *International Conference on Robotics and Automation*, 2018.

- [10] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “Ros: an open-source robot operating system,” in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.