The International Journal of
Robotics Research
000(00):1–16
©The Author(s) 2010
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI:doi number
http://mms.sagepub.com

University of Michigan North Campus Long-Term Vision and Lidar Dataset

Nicholas Carlevaris-Bianco*, Arash K. Ushani†, and Ryan M. Eustice‡

Abstract

This paper documents a large scale, long-term autonomy dataset for robotics research collected on the University of Michigan's North Campus. The dataset consists of omnidirectional imagery, 3D lidar, planar lidar, GPS, and proprioceptive sensors for odometry collected using a Segway robot. The dataset was collected to facilitate research focusing on long-term autonomous operation in changing environments. The dataset is comprised of 27 sessions spaced approximately biweekly over the course of 15 months. The sessions repeatedly explore the campus, both indoors and outdoors, on varying trajectories, and at different times of the day across all four seasons. This allows the dataset to capture many challenging elements including: moving obstacles (e.g., pedestrians, bicyclists, and cars), changing lighting, varying viewpoint, seasonal and weather changes (e.g., falling leaves and snow), and long-term structural changes caused by construction projects. To further facilitate research, we also provide ground-truth pose for all sessions in a single frame of reference.

Keywords

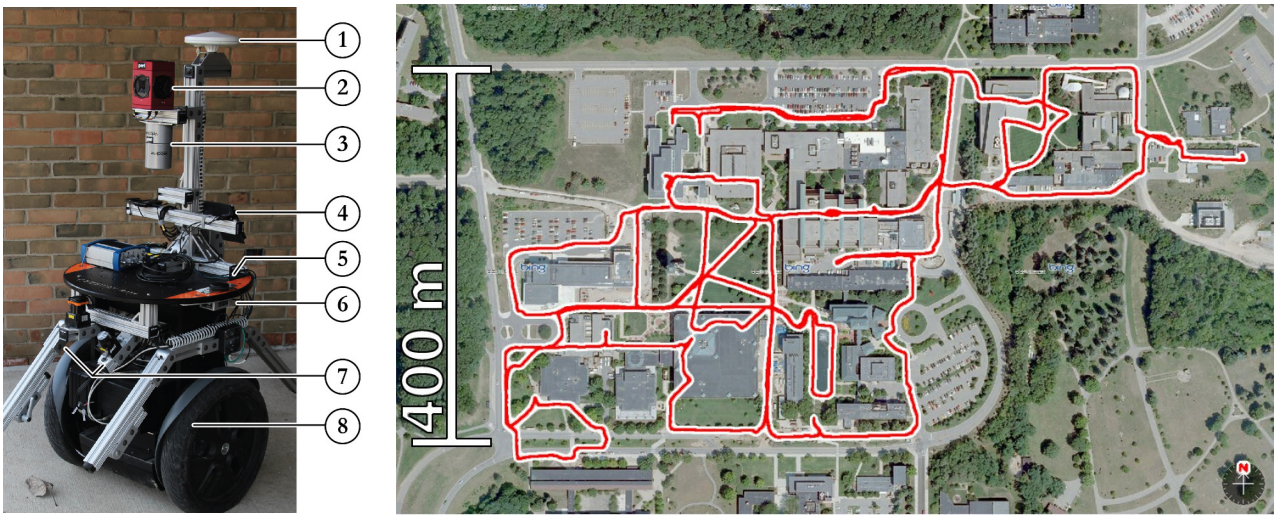
Long-term SLAM, place recognition, lidar, computer vision, field and service robotics

1. Introduction

The North Campus Long-Term (NCLT) dataset consists of data collected by a Segway robotic platform, Fig. 1(a), approximately biweekly, between January 8, 2012 and April 5, 2013, on the University of Michigan's North Campus. The Segway is outfitted with a Ladybug3 omnidirectional camera, a Velodyne HDL-32E 3D lidar, two Hokuyo planar lidars, an inertial measurement unit (IMU), a single-axis fiber optic gyro (FOG), a consumer grade global positioning system (GPS), and a real-time kinematic (RTK) GPS.

The NCLT dataset contains 34.9 hours of logs covering 147.4 km of robot trajectory, and was collected in 27 discrete mapping sessions, Fig. 1(b). Each session covers roughly the entire mapped area and contains both indoor and outdoor environments. Though the same area is repeatedly explored, the path for each session is varied, as is the time of day for each session—from early morning to just after dusk. The length, time-of-day, and environmental conditions during each session are described in Table 1. While the sessions include a variety of environmental conditions, sessions are biased

* N. Carlevaris-Bianco was with the Department of Electrical Engineering & Computer Science, University of Michigan, Ann Arbor, Michigan, USA during the tenure of this work. carlevar@umich.edu.
† A. Ushani is with the Department of Electrical Engineering & Computer Science, University of Michigan, Ann Arbor, Michigan, USA aushani@umich.edu.
‡ R. Eustice is with the Department of Naval Architecture & Marine Engineering, University of Michigan, Ann Arbor, Michigan, USA eustice@umich.edu.



(a) Segway robot

(b) Sample trajectory

Fig. 1. The Segway robotic platform used for experimental data collection. (a) Outfitted with an RTK GPS (1), omni-directional camera (2), 3D lidar (3), IMU (4), consumer-grade GPS (5), 1-axis FOG (6), 2D lidars (7), and CPU (8). (b) Sample trajectory from one session of data collection, overlaid on satellite imagery.

toward favorable weather conditions and do not include adverse weather such as rain or snow as to not risk damaging the robot. The dataset contains many dynamic elements, including pedestrians, bicyclists, and vehicle traffic. Because we repeatedly traverse the same environment, the dataset also captures longer-term dynamics, including moving furniture (indoors), weather and lighting conditions, seasonal changes, and two large construction projects. Sample imagery and lidar data are depicted in Fig. 2. The data is hosted at `http://robots.engin.umich.edu/SoftwareData/NCLT`.

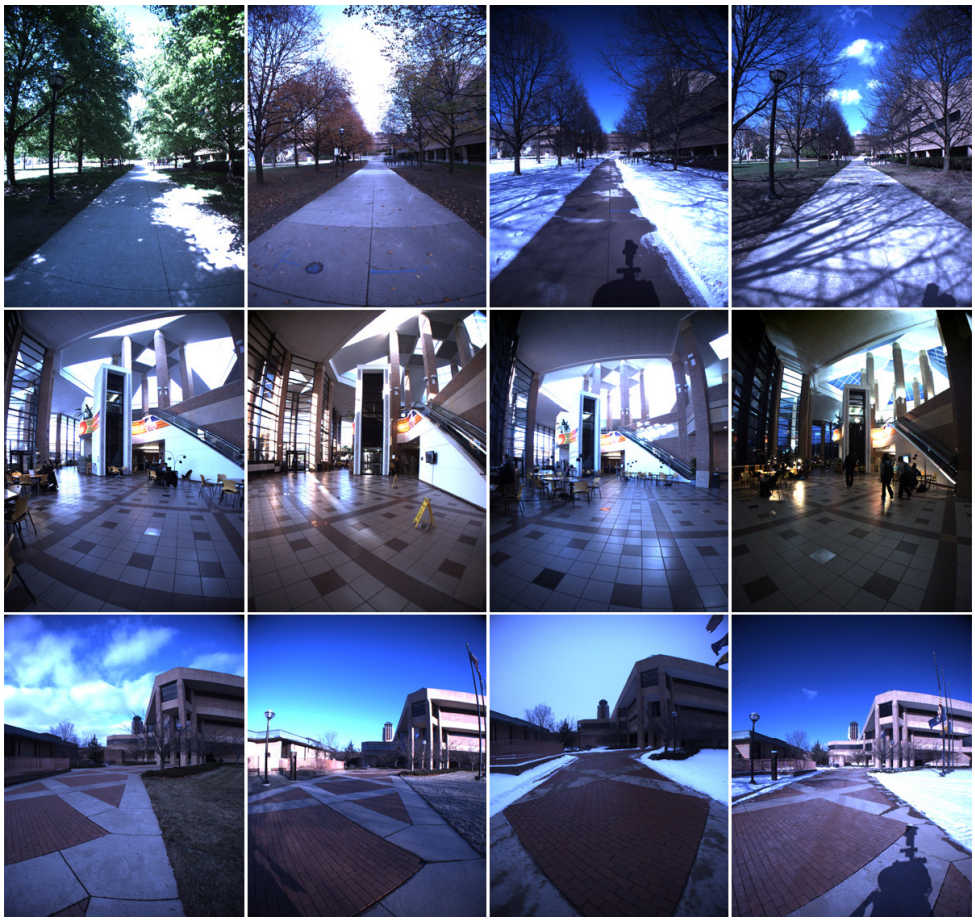
1.1. Applications

The NCLT dataset provides a challenging opportunity to evaluate many robotic algorithms and facilitate research in several areas that have been receiving significant interest in the mobile robotics community. First and foremost are tasks related to long-term mapping, navigation, and localization using vision and/or lidar in changing environments. This includes long-term simultaneous localization and mapping (SLAM) (Carlevaris-Bianco and Eustice 2013, Johannsson et al. 2013, Kanji et al. 2014, Konolige and Bowman 2009, Kretzschmar and Stachniss 2012, Walcott-Bryant et al. 2012), experience-based navigation (Churchill and Newman 2013), teach-and-repeat navigation systems (Barfoot et al. 2012, Krüsi et al. 2015), localization (Badino et al. 2012, Dayoub and Duckett 2008, Johns and Yang 2013a, Krajnik et al. 2014, Lowry et al. 2014, Ranganathan et al. 2013), and place recognition (Cummins and Newman 2008, Johns and Yang 2013b, Masatoshi et al. 2015, Milford and Wyeth 2012, Mishkin et al. 2015). Beyond these spatial tasks, the dataset could be used for obstacle detection and tracking (using either the lidar data or the imagery), and for computer vision tasks including learning visual feature descriptors for things such as changing lighting (Carlevaris-Bianco and Eustice 2014, Lategahn et al. 2013), weather and seasons (Krajnik et al. 2015, McManus et al. 2015), or predicting appearance change (Carlevaris-Bianco and Eustice 2012, Linegar et al. 2015, Neubert et al. 2013).

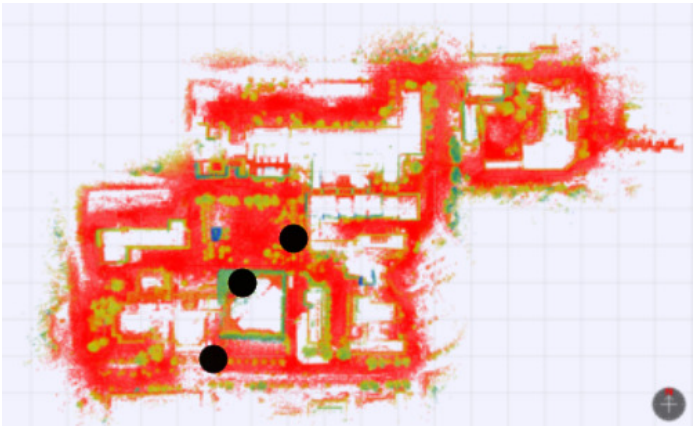
1.2. Comparison with existing datasets

There are several other related datasets already available to the robotics community (see Table 2), however, the NCLT dataset provides a unique combination of attributes including:

- Large scale—both spatially and temporarily (roughly a square kilometer over 15 months)



(a) Sample imagery



(b) Sample single-session point cloud

Fig. 2. The University of Michigan North Campus Long-Term dataset. (a) Sample images from the dataset (only forward camera shown). (b) Sample coverage provided by the lidar point cloud as collected from a single session (colored by height above ground). The grid represents 100 m. The black circles indicate where the sample images were taken.

Table 1: Data Collection Sessions

Date	Length	Time	Sky	Foliage	Snow
2012-01-08	6.4 km	Midday	Partly cloudy	No	No
2012-01-15	7.5 km	Afternoon	Sunny	No	Yes
2012-01-22	6.1 km	Afternoon	Cloudy	No	Yes
2012-02-02	6.2 km	Afternoon	Sunny	No	No
2012-02-04	5.5 km	Afternoon	Sunny	No	No
2012-02-05	6.5 km	Morning	Sunny	No	No
2012-02-12	5.8 km	Midday	Sunny	No	Yes
2012-02-18	6.2 km	Evening	Sunny	No	No
2012-02-19	6.2 km	Midday	Partly cloudy	No	No
2012-03-17	5.8 km	Morning	Sunny	No	No
2012-03-25	5.8 km	Midday	Sunny	No	No
2012-03-31	6.0 km	Midday	Cloudy	No	No
2012-04-29	3.1 km	Morning	Sunny	Yes	No
2012-05-11	6.0 km	Midday	Sunny	Yes	No
2012-05-26	6.3 km	Evening	Sunny	Yes	No
2012-06-15	4.1 km	Morning	Sunny	Yes	No
2012-08-04	5.5 km	Morning	Sunny	Yes	No
2012-08-20	6.0 km	Evening	Sunny	Yes	No
2012-09-28	5.6 km	Evening	Sunny	Yes	No
2012-10-28	5.6 km	Midday	Cloudy	No	No
2012-11-04	4.8 km	Morning	Cloudy	No	No
2012-11-16	4.8 km	Evening	Sunny	No	No
2012-11-17	5.7 km	Midday	Sunny	No	No
2012-12-01	5.0 km	Evening	Sunny	No	No
2013-01-10	1.1 km	Afternoon	Cloudy	No	Yes
2013-02-23	5.2 km	Afternoon	Cloudy	No	Yes
2013-04-05	4.5 km	Afternoon	Sunny	No	Yes

- Vision and lidar data (hundreds of thousands of images and hundreds of gigabytes of lidar data)
- Repeated exploration of the same environment (roughly every other week)
- Indoor and outdoor coverage (e.g., inside buildings and hallways as well as on streets, sidewalks, and pathways)
- Large variety of environmental changes observed:
  - People, bikes and cars
  - Moving furniture
  - Changing lighting (datasets collected at different times of day)
  - Seasonal changes
  - Structural changes (e.g., construction of buildings)
- Intra- and inter-session ground-truth (generated via SLAM).

The New College Vision and Laser Dataset (Smith et al. 2009) provides stereo vision, omnidirectional vision, and lidar from a single 2.2 km session. The Rawseeds project (Ceriani et al. 2009) provides two datasets with omnidirectional and stereo vision, and planar laser; both datasets have repeated sessions through the same environment, though significantly fewer and smaller in scale than the NCLT dataset (the Bicocca dataset provides five indoor sessions while the Bovisa dataset provides six outdoor sessions). The CMU Visual Localization Dataset (Badino et al. 2011) includes monocular vision from

Table 2: Summary of Available Datasets

Dataset	Vision	Lidar	Large Scale	Seasonal Change	Ground-Truth Pose
New College Vision and Laser Dataset	Y	Y	Spatially	N	N
Rawseeds Project	Y	Y	No	N	Y
CMU Visual Localization Dataset	Y	N	Spatially & Temporally	Y	N
Ford Campus Vision and Lidar Dataset	Y	Y	Spatially	N	N
Alderly Day/Night Dataset	Y	N	Spatially	N	N
Nordland Dataset	Y	N	Spatially	Y	N
Malaga Urban Dataset	Y	Y	Spatially	N	N
VPRiCE Dataset	Y	N	Spatially	N	N
KITTI Dataset	Y	Y	Spatially & Temporally	N	N
Cross Season Dataset	Y	N	Spatially	N	N
MIT Stata Center Dataset	Y	Y	Spatially & Temporally	Y	Y
NCLT Vision and Lidar Dataset	Y	Y	Spatially & Temporally	Y	Y

16 sessions covering the same trajectory over the course of a year. The Ford Campus Vision and Lidar Dataset (Pandey et al. 2011) provides three sessions with omnidirectional vision and 3D lidar. The Alderley Day/Night Dataset (Milford and Wyeth 2012) contains two sessions on the same route, one collected during the day and one at night. The Nordland Dataset (Norwegian Broadcasting Corporation 2013), promoted by (Sunderhauf et al. 2013), contains monocular vision for four $3000\ \mathrm{km}$ sessions collected in each of the four seasons—because the data was collected from a train, each session follows exactly the same trajectory. The Malaga Urban Dataset (Blanco-Claraco et al. 2014) contains a single trajectory with stereo vision and planar lidar. The VPRiCE Challenge Dataset (Suenderhauf 2015) provides two sets of imagery aimed toward place recognition contests. The Cross Season Dataset (Masatoshi et al. 2015) provides imagery on a university campus once per each of four seasons. The KITTI Dataset (Geiger et al. 2013) provides six hours of stereo vision and 3D lidar data. This ambitious and large-scale dataset also provides ground-truth for a number of vision tasks including object labels in the form of 3D tracklets and benchmarks for stereo, optical flow, and object detection; however, the KITTI dataset was collected from a car and is therefore confined to outdoor street scenes. Additionally, the KITTI dataset does not focus on repeated exploration of the same environment. The MIT Stata Center Dataset (Fallon et al. 2013) is most similar in spirit to the NCLT dataset in that it provides multiple sessions with stereo vision over the course of a year. The dataset is similar in scale to the NCLT with 38 hours and $42\ \mathrm{km}$ of repeated exploration. The key differences are omnidirectional versus stereo vision and that the Stata Center dataset was collected entirely indoors, limiting its exposure to changing light levels and other temporal changes that occur outdoors.

The remainder of this paper is outlined as follows. In Section 2 we provide details of the sensors used for data collection. Section 3 describes the rigid-body transform and coordinate frame conventions used in the dataset. The Segway's odometry model is described in Section 4. In Section 5 and Section 6 we describe how the Segway's sensors are calibrated, and how the ground-truth is generated. Finally, in Section 7, we describe the dataset directory structure and file formats.

2. Sensors

The sensors collected in the NCLT dataset include:

2.1. Perception sensors

(a) *Velodyne HDL-32E lidar*: The HDL-32E has 32 lasers mounted on a rotating head that spins about its vertical axis to provide a full 360° azimuthal field of view (Vel 2012). The range of the sensor is $100\ \mathrm{m}$. We captured our dataset

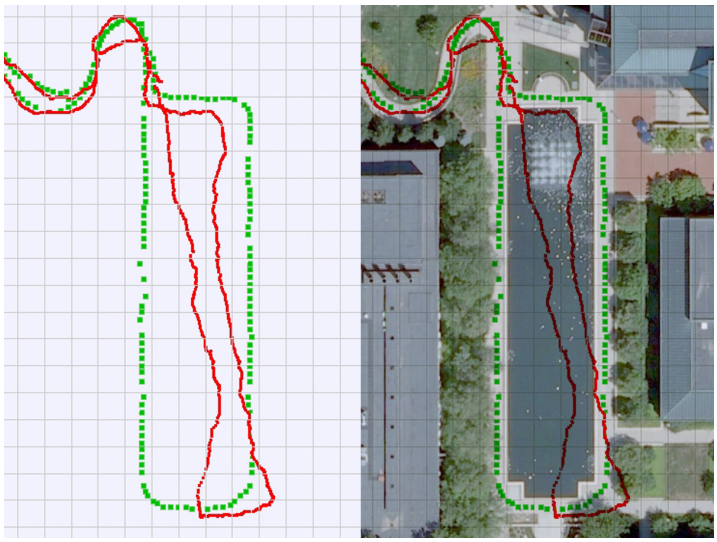


Fig. 3. Comparison of GPS performance between RTK GPS (green) and consumer grade GPS (red) in the NCLT dataset. The true trajectory follows the RTK solution around the fountain. This performance is typical for many of the outdoor portions of the dataset as the consumer GPS is more affected by the multipath from campus buildings.

with the laser spinning at 10 Hz. The sensor is mounted in an upside-down configuration to better image tall vertical structures like buildings and trees.

(b) *Pointgrey Ladybug3 omnidirectional camera*: The Pointgrey Ladybug3 (LB3) is a high resolution omnidirectional camera system (Poi 2011). It has six 2-Megapixel (1600×1200) cameras with five charge coupled devices (CCDs) positioned in a horizontal ring and one positioned vertically, which enables the system to collect video from more than 80% of the full sphere. The camera has a global shutter, and the exposure was set to auto. We collected our dataset at full resolution at 5 Hz as-is from the hardware and stored the images in a JPEG compressed format.

(c) *Hokuyo UTM-30LX lidar*: The UTM-30 is a single beam lidar with 30 m range and a 270° field of view (Hok 2012). The UTM-30 is mounted horizontally on the front of the Segway platform.

(d) *Hokuyo URG-04LX lidar*: The URG-04 is a single beam lidar with a 4 m range and a 240° field of view (Hok 2005). The URG-04 is mounted in a "push-broom" configuration to sweep out the ground plane in front of the vehicle.

2.2. Navigation sensors

(a) *Microstrain 3DM-GX3-45 IMU*: The GX3 IMU contains 3-axis accelerometers, gyroscopes, and magnetometers, and an integrated GPS receiver (Mic 2012). Its internal signal processor provides filtered 3D position, velocity, and attitude at 100 Hz.

(b) *KVH DSP-3000 single-axis FOG*: The KVH single-axis FOG provides highly accurate rotation measurements (KVH 2009). On the Segway platform it is used to measure integrated yaw.

(c) *Garmin 18x 5Hz*: The 18x provides consumer grade GPS (Fig. 3) at 5 Hz (Gar 2011).

(d) *NovAtel DL-4 plus RTK GPS*: The DL-4 GPS receiver provides highly accurate, RTK corrected GPS at 1 Hz (Nov 2005). A NovAtel RTK base station was installed on campus to provide corrections. Outdoors this provides highly accurate position information to ground-truth the robot trajectory (Fig. 3).

3. Coordinate frame conventions

In this section we describe the coordinate frame conventions used in the NCLT dataset. We define the 6 degree of freedom (DOF) pose of frame j with respect to frame i as

$$\mathbf{x}_{ij} = \left[{}^{i}\mathbf{t}_{ij}{}^{\top}, \mathbf{\Theta}_{ij}^{\top} \right]^{\top} = [x_{ij}, y_{ij}, z_{ij}, \phi_{ij}, \theta_{ij}, \psi_{ij}]^{\top}.$$

Here, ${}^{i}\mathbf{t}_{ij}$ is a translation 3-vector from i to j as expressed in frame i, and $\mathbf{\Theta}_{ij}$ is a 3-vector of Euler angles with ϕ representing roll about the x axis, θ as pitch about y, and ψ as yaw about z. To produce the 3×3 orthonormal rotation matrix that rotates frame j into frame i the Euler angles are applied in $\mathrm{rotz}(\psi) \rightarrow \mathrm{roty}(\theta) \rightarrow \mathrm{rotx}(\phi)$ order yielding

$$\begin{aligned}
{}^{i}_{j}\mathrm{R} &= \mathrm{rotxyz}(\mathbf{\Theta}_{ij}) \\
&= \mathrm{rotz}(\psi_{ij})^{\top}\, \mathrm{roty}(\theta_{ij})^{\top}\, \mathrm{rotx}(\phi_{ij})^{\top} \\
&= \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}^{\top} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}^{\top} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix}^{\top}.
\end{aligned}$$

The 4×4 homogeneous coordinate transformation matrix from frame j to frame i defined by $\mathbf{x}_{ij}$ is then given as

$${}^{i}_{j}\mathrm{H} = \begin{bmatrix} {}^{i}_{j}\mathrm{R} & {}^{i}\mathbf{t}_{ij} \\ \mathbf{0}^{\top} & 1 \end{bmatrix}.$$

This 6-DOF convention is used for representing robot pose in the dataset and the rigid-body transformations between the vehicle and sensor coordinate frames.

Robot poses are represented in a local coordinate frame aligned with the cardinal directions, with x pointing north, y east, and z down. The origin of this coordinate frame is fixed in GPS coordinates as described in Table 3. Converting between the local frame and GPS coordinates is done by linearizing around this origin. To define the transformation from GPS coordinates to the local frame we first compute an approximation of the earth's radius in the north-south direction, r_{ns}, and in the east-west direction, r_{ew}, at the origin of the linearization,

$$\begin{aligned}
r_{ns} &= \frac{(r_e r_p)^2}{\left((r_e \cos lat_o)^2 + (r_p \sin lat_o)^2 \right)^{\frac{3}{2}}}, \\
r_{ew} &= \frac{r_e^2}{\sqrt{(r_e \cos lat_o)^2 + (r_p \sin lat_o)^2}},
\end{aligned} \tag{1}$$

where r_e and r_p are the equatorial and polar radii of the earth, respectively, defined in Table 3. Using the local radii we can convert from GPS coordinates to the local frame using

$$\begin{aligned}
x &= \sin(lat - lat_o)\, r_{ns}, \\
y &= \sin(lon - lon_o)\, r_{ew} \cos lat_o, \\
z &= alt_o - alt.
\end{aligned} \tag{2}$$

Table 3: GPS Linearization Constants

Latitude Origin	lat_o	42.293227°
Longitude Origin	lon_o	-83.709657°
Altitude Origin	alt_o	270 m
Earth Equatorial Radius	r_e	6 378 135 m
Earth Polar Radius	r_p	6 356 750 m

Table 4: Sensor Coordinate Frames

Sensor	**Transform**	x m	y m	z m	ϕ°	θ°	ψ°
Velodyne lidar	$\mathbf{x}_{\text{body,vel}}$	0.002	−0.004	−0.957	0.807	0.166	−90.703
Ladybug3 Base	$\mathbf{x}_{\text{body,lb3}}$	0.035	0.002	−1.23	−179.93	−0.23	0.50
Microstrain IMU	$\mathbf{x}_{\text{body,imu}}$	−0.11	−0.18	−0.71	0	0	0
KVH FOG	$\mathbf{x}_{\text{body,fog}}$	0	−0.25	−0.49	0	0	0
Garmin GPS	$\mathbf{x}_{\text{body,gps}}$	0	−0.25	−0.51	–	–	–
Novatel RTK GPS	$\mathbf{x}_{\text{body,rtk}}$	−0.24	0	−1.24	–	–	–
Hokuyo UTM30-LX lidar	$\mathbf{x}_{\text{body,h30}}$	0.28	0	−0.44	180	0	0
Hokuyo URG04-LX lidar	$\mathbf{x}_{\text{body,h04}}$	0.31	0	−0.38	180	−40	0

Conversely, we can convert from the local frame to GPS coordinates using

$$\begin{aligned} lat &= \arcsin\left(\frac{x}{r_{ns}}\right) + lat_o\,, \\ lon &= \arcsin\left(\frac{y}{r_{ew}\cos lat_o}\right) + lon_o\,, \\ alt &= alt_o - z\,. \end{aligned} \tag{3}$$

Note that this projection is fast and efficient to evaluate. For additional information, the reader is referred to (Vincenty 1975).

The robot's body-frame is centered on the axle between the Segway's wheels with x pointing forward, y to the right, and z down. Each sensor's frame of reference is defined with respect to this body frame. An illustration of the body and sensor frames is provided in Fig. 4. The 6-DOF transformations for each sensor, relative to the body frame, are given in Table 4.

4. Odometry model

In order to make the NCLT dataset easier to use, we provide two forms of precomputed odometry. The first provides the mean and covariance of the odometry estimate in the local frame at 100 Hz. The second provides the mean and covariance of the *relative* odometry between the timestamps associated with imagery and pre-extracted lidar scans. This allows users to easily build SLAM graphs with pose nodes associated with images and lidar scans.

Odometry is estimated with an extended Kalman filter (EKF) that uses a differential-drive process model to integrate measurements from the Segway's wheel encoders and a single-axis FOG that observes change in heading. Measurement updates are derived from a commodity IMU that observes roll, pitch, and body-frame angular rates.

We define the Segway's state at time t as,

$$\mathbf{x}_t = [x, y, \phi, \theta, \psi, p, q, r]^\top\,,$$

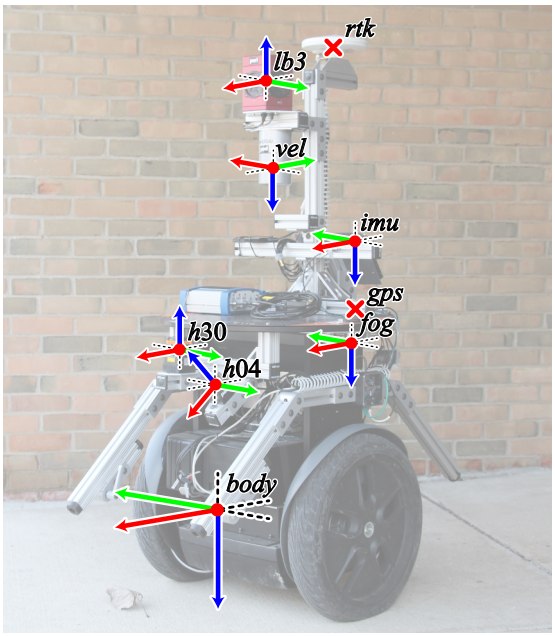


Fig. 4. Illustration of the Segway's sensor frames. For each frame, the x, y, and z axes are colored red, green, and blue, respectively.

where $[x, y]^\top$ represent the robot's translation position in a local frame, $[\phi, \theta, \psi]^\top$ are the Euler angles representing orientation, and $[p, q, r]^\top$ are the body-frame angular rates. We do not estimate the robot's altitude, z, in the local frame because change in z is not observable using the Segway's odometry sensors. Altitude is estimated in our ground-truth by fusing RTK GPS and lidar scan matching with the odometry.

The Segway process model predicts the translation and heading of the robot using a differential drive model, and the roll and pitch using a constant velocity model. For a given time step, the process model takes as input

$$\mathbf{u}_t = [v_r, v_l, \delta_\psi]^\top,$$

where v_r and v_l represent the speeds of the left and right wheels and δ_ψ denotes the change in heading measured by the single-axis FOG. Given the two wheel speeds, we can compute the speed of the vehicle at the center of the wheelbase as

$$v_c = \frac{1}{2}(v_r + v_l). \tag{4}$$

The relationship between the body-frame angular rates and the roll and pitch rates can be derived, as described in (Eustice 2005, §A.3), by considering the inverse relationship where the Euler angle rotation sequence $\mathrm{rotz}(\psi) \rightarrow \mathrm{roty}(\theta) \rightarrow \mathrm{rotx}(\phi)$ is used to map Euler rates to body rates as

$$\begin{aligned}
\begin{bmatrix} p \\ q \\ r \end{bmatrix} &= \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \mathrm{rotx}(\phi) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \mathrm{rotx}(\phi)\,\mathrm{roty}(\theta) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \\
&= \underbrace{\begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix}}_{\mathcal{J}^{-1}} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}.
\end{aligned} \tag{5}$$

Thus, the mapping from body-frame rates to the Euler roll and pitch rates is given by

$$\begin{aligned}\mathcal{J} &= \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix}^{-1} \\ &= \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix}.\end{aligned} \tag{6}$$

The Segway's process model updates the roll and pitch of the vehicle using a constant velocity model. We compute the required angular velocities for roll and pitch as

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \tag{7}$$

Using (4) and (7), the process model is then defined as

$$\hat{\mathbf{x}}_{t+\delta_t} = f(\mathbf{x}_t, \mathbf{u}_t) + \boldsymbol{\omega}_t = \begin{bmatrix} x + v_c\cos(\theta)\delta_t \\ y + v_c\sin(\theta)\delta_t \\ \phi + \dot{\phi}\delta_t \\ \theta + \dot{\theta}\delta_t \\ \psi + \delta_\psi \\ p \\ q \\ r \end{bmatrix} + \boldsymbol{\omega}_t\,, \tag{8}$$

where $\boldsymbol{\omega}_t \sim \mathcal{N}(\mathbf{0}, \mathrm{Q})$ is the process model noise and δ_t is the duration of the time step. The process model noise is comprised of two terms—one capturing the uncertainties associated with the control vector and another capturing uncertainties in the constant velocity terms, which results in the covariance matrix:

$$\begin{aligned}\mathrm{Q} = \frac{\partial f}{\partial \mathbf{u}_t}\,\mathrm{diag}([\sigma_{v_r}^2, \sigma_{v_l}^2, \sigma_{\delta_\phi}^2])\frac{\partial f}{\partial \mathbf{u}_t}^\top \\ + \mathrm{diag}([0, 0, \sigma_\phi^2, \sigma_\theta^2, 0, \sigma_p^2, \sigma_q^2, \sigma_r^2])\,.\end{aligned} \tag{9}$$

Measurement updates are derived from the Microstrain IMU, which observes the platform's roll, pitch, and body-frame angular rates. This leads to linear observation models

$$\begin{aligned}\hat{\mathbf{z}}_{\phi\theta} &= h_{\phi\theta}(\mathbf{x}_t) = \begin{bmatrix} \phi \\ \theta \end{bmatrix} + \boldsymbol{\nu}_{\phi\theta}\,, \\ \hat{\mathbf{z}}_{pqr} &= h_{pqr}(\mathbf{x}_t) = \begin{bmatrix} p \\ q \\ r \end{bmatrix} + \boldsymbol{\nu}_{pqr}\,,\end{aligned} \tag{10}$$

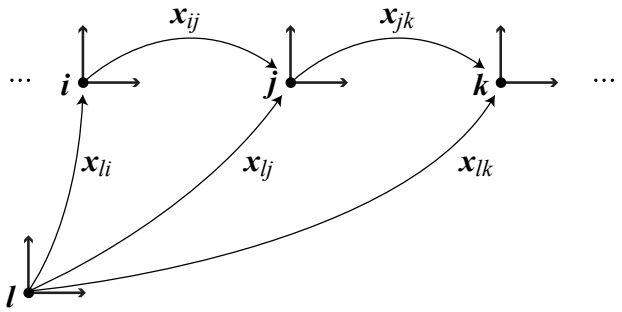


Fig. 5. The delayed-state EKF estimates the pose of the last node added to the graph and the current pose of the robot. The joint distribution of these two poses is used to produce the relative odometry factor.

where

$$\boldsymbol{\nu}_{\phi\theta} \sim \mathcal{N}\left(\mathbf{0}, \mathrm{diag}([\sigma_\phi^2, \sigma_\theta^2])\right) ,$$
$$\boldsymbol{\nu}_{pqr} \sim \mathcal{N}\left(\mathbf{0}, \mathrm{diag}([\sigma_p^2, \sigma_q^2, \sigma_r^2])\right) .$$

In order to produce the relative odometry factors between poses that are used in our SLAM system, the EKF tracks the current pose of the robot and the pose of the last node added to the graph within a delayed-state framework (Eustice et al. 2006). Letting $\mathbf{x}_{li}$ denote the pose associated with the last node added to the SLAM graph, and $\mathbf{x}_{lj}$ denote the current robot pose, the EKF estimates the distribution

$$p(\mathbf{x}_{li}, \mathbf{x}_{lj}) \sim \mathcal{N}\left(\begin{bmatrix}\boldsymbol{\mu}_{li}\\ \boldsymbol{\mu}_{lj}\end{bmatrix}, \begin{bmatrix}\Sigma_{li,li} & \Sigma_{li,lj}\\ \Sigma_{lj,li} & \Sigma_{lj,lj}\end{bmatrix}\right) . \tag{11}$$

When we wish to add a new node associated with the pose $\mathbf{x}_{lj}$ to the graph, we can compute the relative transform from the last node to the current robot pose using the "tail-to-tail" function described by Smith, Self, and Cheeseman (Smith et al. 1990). Because the delayed-state EKF tracks the correlation between the current robot pose and the last node added to the graph, we can also compute a first-order approximation of the covariance of the odometry factor. This yields the relative factor

$$p(\mathbf{x}_{ij}) \sim \mathcal{N}\left(\ominus\mathbf{x}_{li} \oplus \mathbf{x}_{lj},\ {}_{\ominus}\mathrm{J}_{\oplus} \begin{bmatrix}\Sigma_{li,li} & \Sigma_{li,lj}\\ \Sigma_{lj,li} & \Sigma_{lj,lj}\end{bmatrix} {}_{\ominus}\mathrm{J}_{\oplus}^{\top}\right) , \tag{12}$$

where ${}_{\ominus}\mathrm{J}_{\oplus}$ is the Jacobian of the tail-to-tail function. We then marginalize the old pose, $\mathbf{x}_{li}$, from the delayed-state filter and augment the state with a new vector of variables to track the current pose of the robot, $\mathbf{x}_{lk}$. This process is illustrated in Fig. 5.

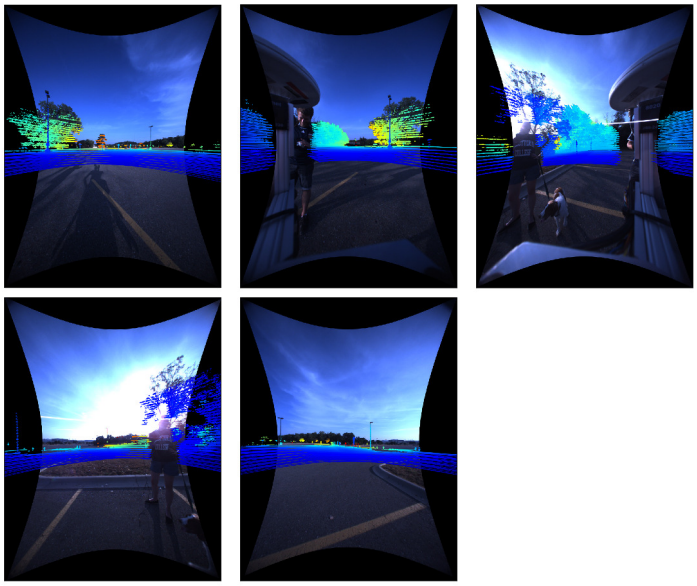
5. Sensor calibration

5.1. Velodyne-frame to body-frame calibration

The transformation between the Velodyne-frame and body-frame is calibrated by solving a SLAM problem that includes the robot's pose and the uncertain Velodyne-to-body transform. Odometry factors constrain the motion of the robot, while lidar scan matching factors constrain the motion of the robot and the Velodyne-to-body transformation.

5.2. Velodyne-frame to Ladybug3-frame calibration

The transformation between the Velodyne and the Ladybug3 is calibrated by minimizing the mutual information between the two modalities as described in (Pandey et al. 2015). By compounding the Velodyne-to-body and the Velodyne-to-Ladybug3



(a) Sample imagery shown with lidar point cloud data overlaid

(b) Zoomed in regions of the above

Fig. 6. (a) Sample images from the dataset shown with 10 revolutions of lidar point cloud data that has been motion compensated. Points are colored by distance. (b) Zoomed in regions of the images.

transformations we can compute the Ladybug3-to-body transformation. The quality of the calibration is demonstrated in Fig. 6.

5.3. Ladybug3 intrinsic calibration

The Ladybug3 is comprised of six individual cameras each with its own camera frame. The extrinsic calibration between each camera and the base frame and the intrinsic calibrations for each camera, including lens distortion, are provided by the manufacturer. The transform between the Ladybug3 base frame and each camera frame is given in Table 5. The cameras' intrinsic calibrations are provided in Table 6 where the standard pinhole intrinsics matrix is defined as

$$K = \begin{bmatrix} f_x & \alpha_c & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}.$$

The transformations to distort and undistort the images are also provided as a dense pixelwise mapping. The file format of the distort and undistort maps are described in Section 7.6.

5.4. Additional Sensors

The coordinate frames for the other sensors, such as the Hokuyo planar lidar and IMU sensors, were manually calibrated.

Table 5: Ladybug3 Camera Coordinate Frames

	Transform	x m	y m	z m	ϕ°	θ°	ψ°
Camera0	$\mathbf{x}_{\text{lb3,c0}}$	0.000920	−0.000569	0.062413	−0.028132	0.196467	0.248664
Camera1	$\mathbf{x}_{\text{lb3,c1}}$	0.014543	0.039337	0.000398	−138.449751	89.703877	−66.518051
Camera2	$\mathbf{x}_{\text{lb3,c2}}$	−0.032674	0.025928	0.000176	160.101024	89.836345	−56.101163
Camera3	$\mathbf{x}_{\text{lb3,c3}}$	−0.034969	−0.022993	0.000030	95.603967	89.724274	−48.640335
Camera4	$\mathbf{x}_{\text{lb3,c4}}$	0.011238	−0.040367	−0.000393	−160.239278	89.812338	127.472911
Camera5	$\mathbf{x}_{\text{lb3,c5}}$	0.041862	−0.001905	−0.000212	160.868615	89.914152	160.619894

Table 6: Ladybug3 Camera Intrinsic Calibration

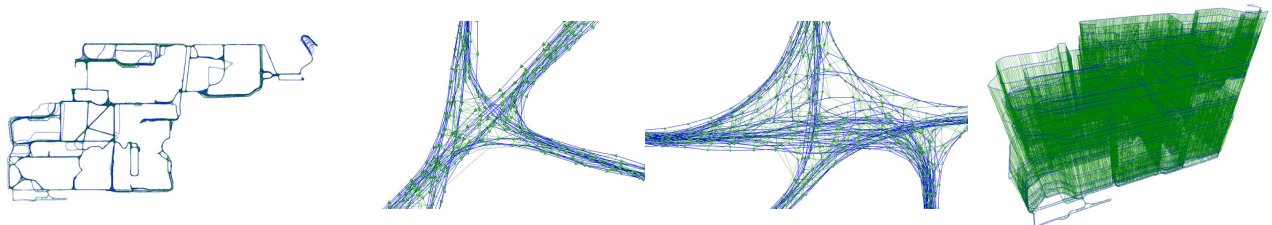
	f_x	f_y	c_x	c_y	α_c
Camera0	410.777504	410.777504	613.277472	805.879152	0.0
Camera1	409.719024	409.719024	624.237344	813.947840	0.0
Camera2	408.385824	408.385824	623.058320	793.959536	0.0
Camera3	406.802144	406.802144	617.352928	810.874208	0.0
Camera4	403.920816	403.920816	601.929520	823.270544	0.0
Camera5	399.433184	399.433184	621.668624	826.361952	0.0

5.5. Time Synchronization

The Ladybug3 provides time synchronization over the IEEE1394 hardware interface. The KVH is timestamped according to the arrival time of the sensor message. Other sensors are time synchronized according to the method proposed in (Olson 2010). All data is logged and timestamped using one computer on the Segway.

6. Ground-truth

We have preprocessed a large SLAM solution, Fig. 7, with all sessions using lidar scan matching and high-accuracy RTK GPS (Fig. 3) to provide ground-truth robot pose. Note that scan matching constraints are added within each session and between sessions, to ensure that the pose is accurate where RTK GPS is unavailable or inaccurate and to align all sessions into a consistent frame. For example, the RTK GPS is often unavailable indoors or can be off by over ten meters near certain buildings due to multipath issues that arise around campus. For this reason, we also provide a measure of the error in the RTK GPS as compared to our provided ground-truth solution. Scan matching constraints were automatically added between all pairs of nodes within 8 m of each other, and then bad scan matching factors were manually identified and removed from the graph. To compute ground-truth poses between nodes in the graph we interpolate based on the odometry.



(a) Top View (b) Top View (zoom) (c) Top View (zoom) (d) Time Scaled

Fig. 7. Ground-truth SLAM graph comprised of all sessions (a). Links include odometry (blue) and 3D lidar scan matching (green). Close up views in (b) and (c) show the graph at two commonly-visited intersections. An oblique view scaled by time in the z-axis is shown in (d). Each layer along the z-axis represents a mapping session.

Table 7: GPS CSV File Format

Field	Description	Units
1	UTIME of the GPS fix	µs
2	Fix mode, as reported by the GPS unit: 0 means that the mode update is not yet seen, 1 means that there is no GPS fix, 2 means that the fix is good for longitude and latitude, 3 means that the fix is also good for altitude.	–
3	The number of satellites used in the fix	–
4	Latitude	rad
5	Longitude	rad
6	Altitude	m
7	Track	m
8	Speed	m/s

Table 8: Microstrain IMU CSV File Format

Field	Description	Units
1	UTIME of the measurements	µs
2–4	3-DOF magnetic field strength vector	Gauss
5–7	3-DOF acceleration vector	m/s^2
8–10	3-DOF angular rotation rate for roll, pitch, and heading	rad/s

7. Directory structure and file formats

In this section we describe the directory structure of the NCLT dataset and the file formats. We use the placeholders "YYYY-MM-DD" to represent the date format used to differentiate each of the 27 runs and "UTIME" to represent the 16 digit integer timestamp (microseconds since the UNIX epoch).

7.1. Images directory

The images directory contains a tar file for each run, YYYY-MM-DD_lb3.tar.gz, that contains the imagery from the Ladybug3 camera system. Within each tar file there are six camera folders, one for each camera in the Ladybug3 system, labeled Cam0 through Cam5. The images collected for each camera are stored in the TIFF file format and are labeled using the timestamp when they were collected (UTIME.tiff).

7.2. Sensor data directory

The sensor_data directory contains a tar file for each run, YYYY-MM-DD_sen.tar.gz, that contains the sensor data (excluding Velodyne data). Within each tar file there are a number of comma-separated values (CSV) files, described below.

- gps.csv and gps_rtk.csv are CSV files that contain the sensor data from the consumer grade GPS and the RTK GPS, respectively. The fields are described in Table 7. Additionally, we provide a file, gps_rtk_err.csv, that contains the error between the RTK GPS fix and the SLAM ground-truth solution, as described in Section 6. Each line contains the UTIME of the RTK GPS fix followed by the error in meters to the ground-truth.

- ms25.csv is a CSV file that contains the sensor data from the Microstrain IMU. These values use the Microstrain's internal filter and are described in Table 8. Additionally, ms25_euler.csv contains the Euler angles read from the

Table 9: Velodyne Packet Header Binary Format

Byte	Description	Units
1–8	Magic Sequence (`9C AD 9C AD 9C AD 9C AD`)	–
9–12	Number of hits in packet	–
13–20	`UTIME` associated with packet	μs
21–24	Padding	–

Microstrain IMU, these values are the instantaneous readings. Each line of the CSV file contains the `UTIME` associated with the measurement, followed by the three Euler angles in radians (roll, pitch, and heading).

- `odometry_mu_100hz.csv` is a CSV file that contains 6-DOF odometry measurements calculated from the starting position of each run. These measurements are computed at about 100 Hz. Each line of the CSV file contains the `UTIME` associated with the measurement, followed by the 6-DOF odometry (x, y, z in meters, ϕ, θ, ψ in radians). The accompanying file, `odometry_cov_100hz.csv`, is a CSV file that contains the covariance of the measurements in `odometry_mu_100hz.csv`. We record the upper diagonal of the covariance matrix. Each line of the CSV file contains the `UTIME` associated with the measurement, followed by the upper diagonal of the covariance matrix (row major).

- `odometry_mu.csv` and `odometry_cov.csv` are CSV files that contain 6-DOF odometry measurements and covariances, respectively, synchronized with each image event as described in Section 4. This is calculated relative to the previous image event and follows the same format as above.

7.3. Velodyne data directory

The `velodyne_data` directory contains the sensor data from the Velodyne lidar. Within each tar file, there is a file that contains all of the observed points and a folder that contains the Velodyne scans associated with each image. As a convenience to the user, the raw range measurements are reported directly as (x, y, z) Cartesian positions within the sensor frame using the Velodyne intrinsics. Due to file size considerations, we encode this data in a binary format—all values are in little-endian. For each point, we scale each of x, y, z, to an integer between 0 and 40 000 by adding 100 m and discretize the result at 5 mm. For example, -90 m gets scaled to 2000. This scaled value is written out for each of x, y, z as a 2 byte integer. Hits that have a measurement out of range due to sensor error are set to $(0, 0, 0)$ in the sensor frame.

- `velodyne_hits.bin` contains all of the recorded points collected during the session. The points are grouped into "packets" consistent with one Ethernet packet as reported by the Velodyne sensor. For each packet, we begin with a 24 byte header described in Table 9. Following this 24 byte header, each point within the packet is recorded by its (x, y, z) Cartesian location in the sensor frame (0–40 000, 2 bytes), the intensity of the return (0–255, 1 byte), as well as the laser id number (0–31, 1 byte), for a total of 8 bytes per point. Each packet comprises at most a collection of 384 points where 384 is the result of 12 groups of 32-laser firings; however, our upside-down Velodyne mounting causes many lasers to point into the sky yielding no measured range return—these no-return points are not recorded in the packet. The total time to collect one packet is 552.96 μs, which at 1 m/s max velocity of the Segway results in (at most) a negligible 0.5 mm of motion blur—hence the laser returns within a packet, for all practical purposes, can be treated as near instantaneous within the NCLT dataset. The reported `UTIME` in the header is the timestamp of the *last* laser fired within the packet; further details of the Velodyne laser timing can be found in the product manual (Vel 2012, Appendix E).

- The `velodyne_sync` directory contains a list of files, one associated with each image. Each file is named according to the image with which it is associated (`UTIME.bin`). Each file contains one revolution's worth of Velodyne hits

(corresponding to the previous 0.1 seconds), these points are motion compensated for the Segway's egomotion during the scan and then recorded in the Segway body frame. Each point is written using the 8 byte format described above.

We include two sample python scripts to demonstrate reading these binary files and how to perform the scaling and discretization. In `read_vel_hits.py`, we read all of the hits in the `velodyne_hits.bin` file. In `vel_sync_to_csv.py`, we show how to convert a `velodyne_sync/UTIME.bin` file to a CSV format. Additionally, we also provide a file, `laser_angles.csv`, that contains the angles, in radians, of the 32 laser beams of the Velodyne with respect to the horizontal plane.

7.4. Hokuyo data directory

The `hokuyo_data` directory contains the sensor data from the two Hokuyo lidar sensors. Within each tar file, there are two binary files that contain the observations for each sensor: `hokuyo_30m.bin` contains the observations for the Hokuyo UTM-30LX lidar, and `hokuyo_4m.bin` contains the observations for the Hokuyo URG-04LX lidar.

Each file contains the data associated with all of the scans recorded by each sensor, respectively. The format is as follows. First, four bytes are used to represent the timestamp at which the scan was collected. Then, all of the observations for the scan are recorded sequentially by the angle of the laser beam. For the Hokuyo UTM-30LX lidar, there are 1081 observations, starting at an angle of $-135°$ with a step of $0.25°$ per observation. For the Hokuyo URG-04LX, there are 726 observations, starting from an angle of $-119.5312°$ with a step of $0.3516°$ per observation. Each observation is recorded using two bytes. Distances are scaled to an integer between 0 and $40\,000$ by adding 100 m to each distance and discretizing the result at 5 mm. (Hits that have an out of range measurement are set to 0.) All scans are recorded sequentially in the binary file as described above. Note that in each scan (i.e., array of ranges), all of the ranges share the same timestamp.

We include two sample python scripts to demonstrate reading these binary files. In `read_hokuyo_30m.py`, we read all of the hits collected by the Hokuyo UTM-30LX lidar scan by scan and plot each scan. In `read_hokuyo_4m.py`, we do the same for the Hokuyo URG-04LX lidar.

7.5. ROSbag

In addition to providing the sensor data in CSV format, we provide support for ROS. We provide several python scripts to generate ROSbags from the CSV files we provide. These are: `gps_to_rosbag.py`, `ms25_to_rosbag.py`, `ms25_euler_to_rosbag.py`, `vel_to_rosbag.py`, `hokuyo_30m_to_rosbag.py`, `hokuyo_4m_to_rosbag.py`, and `sensordata_to_rosbag.py`. Furthermore, these can be used in conjuction with the coordinate transform package in ROS to apply the coordinate frames discussed in Section 3.

7.6. Ladybug3 calibration directory

The `ladybug3_calib` directory contains the lens distortion mappings for the individual cameras. `U2D_Cam[0-5]_1616X1232.txt` provides the undistort maps for each of the Ladybug3 cameras. The first line is a header that gives the image size. All subsequent lines contain four space-delimited fields. The first two fields are the pixel coordinates in the source image while the second two fields are the pixel coordinates in the destination image. The `D2U_Cam[0-5]_1616X1232.txt` files describe the inverse transform, which distorts the images, in the same format.

7.7. CAD Model

We provide a computer aided design (CAD) model of the Segway robot in `segway_cad.tar.gz`, which contains a directory of SolidWorks parts and assemblies for the Segway robot.

8. Conclusions

In this paper we have presented a large-scale, long-term vision and lidar dataset. This dataset captures a wide variety of temporal changes in the environment, and provides the opportunity to explore many important topics in robotics research, including long-term localization, mapping, and navigation. We have made this dataset available to the community in the hopes that it will be of value to others working in this area.

Funding

This work was supported in part by the National Science Foundation under award IIS-0746455, by the Office of Naval Research under award N00014-12-1-0092, and by the Naval Sea Systems Command through the Naval Engineering Education Center under award N65540-10-C-0003.

References

H. Badino, D. Huber, and T. Kanade. The CMU visual localization data set, 2011.

H. Badino, D. Huber, and T. Kanade. Real-time topometric localization. In *Proc. IEEE Int. Conf. Robot. and Automation*, pages 1635–1642, Saint Paul, MN, USA, May 2012.

T. D. Barfoot, B. Stenning, P. Furgale, and C. McManus. Exploiting reusable paths in mobile robotics: Benefits and challenges for long-term autonomy. In *Proc. Comp. and Robot Vision*, pages 388–395, Toronto, Canada, 2012.

J.-L. Blanco-Claraco, F.-A. Moreno-Duenas, and J. Gonzalez-Jimenez. The Malaga urban dataset: High-rate stereo and LiDAR in a realistic urban scenario. *Int. J. Robot. Res.*, 33(2):207–214, 2014.

N. Carlevaris-Bianco and R. M. Eustice. Learning temporal co-observability relationships for lifelong robotic mapping. In *IROS Workshop on Lifelong Learning for Mobile Robotics Applications*, Vilamoura, Portugal, October 2012.

N. Carlevaris-Bianco and R. M. Eustice. Long-term simultaneous localization and mapping with generic linear constraint node removal. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pages 1034–1041, Nov. 2013.

N. Carlevaris-Bianco and R. M. Eustice. Learning visual feature descriptors for dynamic lighting conditions. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pages 2769–2776, Chicago, IL, Sept. 2014.

S. Ceriani, G. Fontana, A. Giusti, D. Marzorati, M. Matteucci, D. Migliore, D. Rizzi, D. G. Sorrenti, and P. Taddei. Rawseeds ground truth collection systems for indoor self-localization and mapping. *Auton. Robot.*, 27(4):353–371, 2009.

W. Churchill and P. Newman. Experience-based navigation for long-term localisation. *Int. J. Robot. Res.*, 32(14):1645–1661, 2013.

M. Cummins and P. Newman. FAB-MAP: Probabilistic localization and mapping in the space of appearance. *Int. J. Robot. Res.*, 27(6): 647–665, June 2008.

F. Dayoub and T. Duckett. An adaptive appearance-based map for long-term topological localization of mobile robots. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pages 3364–3369, Nice, France, Sept. 2008.

R. M. Eustice. *Large-area visually augmented navigation for autonomous underwater vehicles*. PhD thesis, Massachusetts Inst. Tech. / Woods Hole Ocean. Inst. Joint Program, June 2005.

R. M. Eustice, H. Singh, and J. J. Leonard. Exactly sparse delayed-state filters for view-based SLAM. *IEEE Trans. Robot.*, 22(6): 1100–1114, 2006.

M. Fallon, H. Johannsson, M. Kaess, and J. J. Leonard. The MIT Stata Center dataset. *Int. J. Robot. Res.*, 32(14):1695–1699, 2013.

GPS 18x Technical Specifications. Garmin International, Inc., Olathe, KS, USA, 190-00879-08 Revision D edition, Oct. 2011.

A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.*, 32(11):1231–1237, 2013.

Scanning Laser Range Finder URG-04LX Specification. Hokuyo Automatic Co., LTD, Chuo-Ku, Osaka, Japan, Version 5 edition, Oct. 2005.

Scanning Laser Range Finder UTM-30LX/LN Specification. Hokuyo Automatic Co., LTD, Chuo-Ku, Osaka, Japan, Version 8 edition, Nov. 2012.

H. Johannsson, M. Kaess, M. Fallon, and J. J. Leonard. Temporally scalable visual SLAM using a reduced pose graph. In *Proc. IEEE Int. Conf. Robot. and Automation*, pages 54–61, Karlsruhe, Germany, May 2013.

E. Johns and G.-Z. Yang. Feature co-occurrence maps: Appearance-based localisation throughout the day. In *Proc. IEEE Int. Conf. Robot. and Automation*, pages 3212–3218, Karlsruhe, Germany, May 2013a.

E. Johns and G.-Z. Yang. Generative methods for long-term place recognition in dynamic scenes. *Int. J. Comput. Vis.*, 106(3):297–314, 2013b.

T. Kanji, C. Yuuto, and A. Masatoshi. Mining visual phrases for long-term visual SLAM. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pages 136–142, Chicago, USA, 2014.

K. Konolige and J. Bowman. Towards lifelong visual maps. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pages 1156–1163, St. Louis, MO, USA, Oct. 2009.

T. Krajnik, J. P. Fentanes, O. M. Mozos, T. Duckett, J. Ekekrantz, and M. Hanheide. Long-term topological localisation for service robots in dynamic environments using spectral maps. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pages 4537–4542, Chicago, IL, USA, 2014.

T. Krajnik, P. deCristoforis, M. Nitsche, K. Kusumam, T. Duckett, et al. Image features and seasons revisited. In *European Conference on Mobile Robots*, Troia, Portugal, 2015.

H. Kretzschmar and C. Stachniss. Information-theoretic compression of pose graphs for laser-based SLAM. *Int. J. Robot. Res.*, 31: 1219–1230, 2012.

P. Krüsi, B. Bücheler, F. Pomerleau, U. Schwesinger, R. Siegwart, and P. Furgale. Lighting-invariant adaptive route following using iterative closest point matching. *J. Field Robot.*, 32(4):534–564, 2015.

KVH DSP-3000 Fiber Optic Gyro Technical Manual. KVH Industires, Inc., Middletown, RI, Revision D edition, 2009.

H. Lategahn, J. Beck, B. Kitt, and C. Stiller. How to learn an illumination robust image feature for place recognition. In *Proc. IEEE Intell. Veh. Symp.*, pages 285–291, Gold Coast, Australia, June 2013.

C. Linegar, W. Churchill, and P. Newman. Work smart, not hard: Recalling relevant experiences for vast-scale but time-constrained localisation. In *Proc. IEEE Int. Conf. Robot. and Automation*, Seattle, WA, USA, May 2015.

S. Lowry, G. Wyeth, and M. Milford. Unsupervised online learning of condition-invariant images for place recognition. In *Proc. Australasian Conf. on Robot. and Automation*, Melbourne, Australia, 2014.

A. Masatoshi, C. Yuuto, T. Kanji, and Y. Kentaro. Leveraging image-based prior in cross-season place recognition. In *Proc. IEEE Int. Conf. Robot. and Automation*, pages 5455–5461, Seattle, USA, 2015.

C. McManus, B. Upcroft, and P. Newman. Learning place-dependant features for long-term vision-based localisation. *Autonomous Robots, Special issue on Robotics Science and Systems 2014*, pages 1–25, 2015.

3DM-GX3-45 Theory of Operation. MicroStrain, Inc., Williston, VT, Revision 001 edition, Mar. 2012.

M. Milford and G. Wyeth. SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights. In *Proc. IEEE Int. Conf. Robot. and Automation*, pages 1643–1649, Saint Paul, MN, USA, May 2012.

D. Mishkin, M. Perdoch, and J. Matas. Place recognition with WxBS retrieval. In *CVPR 2015 Workshop on Visual Place Recognition in Changing Environments*, Boston, MA, USA, 2015.

P. Neubert, N. Sunderhauf, and P. Protzel. Appearance change prediction for long-term navigation across seasons. In *Proc. European Conf. Mobile Robot.*, pages 198–203, Barcelona, Spain, Sept. 2013.

Norwegian Broadcasting Corporation. Nordlandsbanen: Minute by minute, season by season, 2013. URL `https://nrkbeta.no/2013/01/15/nordlandsbanen-minute-by-minute-season-by-season/`.

DL-4plus User Manual. NovAtel, Inc., Calgary, AB., Canada, Revision 6 edition, Dec. 2005.

E. Olson. A passive solution to the sensor synchronization problem. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pages 1059–1064, Taipei, Taiwan, 2010.

G. Pandey, J. R. McBride, and R. M. Eustice. Ford campus vision and lidar data set. *Int. J. Robot. Res.*, 30(13):1543–1552, Nov. 2011.

G. Pandey, J. R. McBride, S. Savarese, and R. M. Eustice. Automatic extrinsic calibration of vision and lidar by maximizing mutual information. *J. Field Robot., Special Issue on Calibration for Field Robotics*, 2015. In Press.

Ladybug3 Technical Reference Manual. Point Grey Research, Richmond, BC, Canada, Version 1.8 edition, Mar. 2011.

A. Ranganathan, S. Matsumoto, and D. Ilstrup. Towards illumination invariance for visual localization. In *Proc. IEEE Int. Conf. Robot. and Automation*, pages 3791–3798, Karlsruhe, Germany, May 2013.

M. Smith, I. Baldwin, W. Churchill, R. Paul, and P. Newman. The new college vision and laser data set. *Int. J. Robot. Res.*, 28(5): 595–599, 2009.

R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In I. Cox and G. Wilfong, editors, *Autonomous Robot Vehicles*, pages 167–193. Springer-Verlag, 1990.

N. Suenderhauf. The VPRiCE challenge 2015 — visual place recognition in changing environments. `https://roboticvision.atlassian.net/wiki/pages/viewpage.action?pageId=14188617`, 2015.

N. Sunderhauf, P. Neubert, and P. Protzel. Are we there yet? Challenging SeqSLAM on a 3000 km journey across all four seasons. In *ICRA Workshop on Long-Term Autonomy*, pages 1–3, Karlsruhe, Germany, May 2013.

User's manual and programming guide: HDL-32E High Definition LiDAR Sensor. Velodyne LiDAR, Morgan Hill, CA, Rev E edition, Nov. 2012.

T. Vincenty. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey Review*, 23(176): 88–93, 1975.

A. Walcott-Bryant, M. Kaess, H. Johannsson, and J. J. Leonard. Dynamic pose graph SLAM: Long-term mapping in low dynamic environments. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pages 1871–1878, Vilamoura, Portugal, Oct. 2012.