

Simultaneous Localization, Mapping, and Moving Object Tracking Using Helmet-Mounted Solid-State LiDAR

Ikuro Inaga

Graduate School of Science and Engineering
Doshisha University
Kyotanabe, Kyoto, Japan
e-mail: ctwk0128@mail4.doshisha.ac.jp

Masafumi Hashimoto, Kazuhiko Takahashi

Faculty of Science and Engineering
Doshisha University
Kyotanabe, Kyoto, Japan
e-mail: {mhashimo, katakaha}@mail.doshisha.ac.jp

Abstract—This paper presents a Simultaneous Localization And Mapping (SLAM) and Moving Object Tracking (MOT) method using a small and lightweight solid-state Light Detection And Ranging (LiDAR) attached to a rider helmet for micromobilities, such as bicycles, e-bikes, and e-kick scooters. Distortions in LiDAR point cloud data caused by the movement of the micromobility and head motion of the rider are corrected using the data from LiDAR and inertial measurement unit via a quaternion unscented Kalman filter. The corrected LiDAR point cloud data are classified into three classes: 1) point cloud data related to stationary objects, such as buildings and trees, 2) those related to road obstacles, such as curb stones and road debris, and 3) those related to moving objects. The point cloud data related to stationary objects and road obstacles are used for environment mapping using normal distributions transform SLAM, whereas the point cloud data related to moving objects are used for MOT using Kalman filter. Results from experiments conducted at our university campus demonstrate the effectiveness of the proposed method.

Keywords—*helmet LiDAR; solid-state LiDAR, SLAM; moving-object tracking; distortion correction; quaternion UKF; micromobility.*

I. INTRODUCTION

In recent years, many studies have been conducted on active safety and automated driving of vehicles in Intelligent Transportation Systems (ITS) [1]. An important technology for active safety and automated driving of vehicles is Simultaneous Localization and Mapping (SLAM) to build an environment map using vehicle-mounted sensors, such as Light Detection And Ranging sensors (LiDARs) and cameras. Another important technology is Moving Object Tracking (MOT) to avoid collisions with surrounding moving objects. Accordingly, numerous SLAM and MOT (SLAMMOT) methods have been proposed [2]–[4].

In a decarbonized society, micromobilities, such as bicycles, e-bikes, and e-kick scooters, attract attention as a means of short-distance travel through urban regions [5]. Similar to ITS, active safety is necessary to reduce traffic accidents and increase the use of micromobilities.

In our previous study [6], a SLAMMOT method based on information obtained from a LiDAR attached to the rider helmet for micromobility was proposed. In ITS, mechanical LiDARs, such as Velodyne and Ouster LiDARs, are widely used owing to their reliability and accuracy. The LiDAR used in our

previous study for micromobility was bulky mechanical LiDAR, thus posing problems regarding practicality and usability.

From the viewpoint of size and security, it is desirable to mount a small easily removable sensor on the micromobility handlebars or rider helmet. Modern technology includes a solid-state LiDAR that is smaller and lighter than the mechanical LiDAR [7]. Solid-state LiDAR can substantially enhance active safety in micromobility. Recently, various studies have been conducted on SLAM and MOT methods using solid-state LiDAR [8]–[11] in ITS and mobile robotics. However, to the best of our knowledge, there are no studies that tackle SLAMMOT using solid-state LiDAR for micromobility application.

Therefore, this paper presents a SLAMMOT method using a small and lightweight solid-state LiDAR attached to the rider helmet for micromobility.

The LiDAR point cloud data within the sampling period cannot be captured simultaneously because LiDAR captures measurements by scanning a laser beam. Therefore, when the micromobility is moving or the rider head swings, the acquired LiDAR point cloud data are distorted, which deteriorates the SLAMMOT accuracy.

Distortion in LiDAR point cloud data can be corrected by estimating the LiDAR self-pose in a shorter time than the LiDAR sampling period. Most conventional methods for distortion correction were based on linear interpolation and its variants of the LiDAR self-pose obtained at every acquired LiDAR sample [12][13]. In [14][15], distortion correction methods using the Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) [16] were proposed to improve distortion correction. In our previous studies, Euler angles (i.e., roll, pitch, and yaw angles) were used to represent the LiDAR posture. When driving a micromobility, the head posture often changes considerably during safety confirmations, such as right-left and up-down confirmations. Such large head motions of the rider may deteriorate the accuracy of distortion correction using Euler angle-EKF and UKF.

This problem can be addressed using a quaternion instead of Euler angles as the angle representation. To accurately perform SLAMMOT even under large motions of the rider head, this paper proposes a quaternion-UKF-based distortion correction method. The remainder of this paper is organized as follows. Section II describes the experimental system. Section III presents an overview of SLAMMOT. Section IV explains the proposed distortion correction method for LiDAR point cloud

data, and Section V presents the classification method for these data. Section VI illustrates the effectiveness of the proposed method through experiments. Section VII presents our conclusions and future works.

II. EXPERIMENTAL SYSTEM

The overview of the experimental helmet is shown in Figure 1. A MEMS solid-state LiDAR (Livox Mid-360) and Inertial Measurement Unit (IMU) (Xsens Mti-300) are mounted on the helmet. The weight of the LiDAR is 265 g. As shown in Figure 2, the LiDAR has a maximum range of 40 m, horizontal and vertical Field-Of-View (FOV) of 360° and 59° , respectively, and resolution of 1.4° . The LiDAR acquires 96 measurement points every 0.48 ms. The sampling period of LiDAR measurements for SLAMMOT is set to 0.12 s in this study. Approximately 20,000 measurements can be obtained per LiDAR sampling period.

Measurements of attitude (i.e., roll and pitch angles) and angular velocity (i.e., roll, pitch, and yaw angular velocities) are obtained from the IMU every 10 ms. The errors in attitude and angular velocity are less than $\pm 0.3^\circ$ and $\pm 0.2^\circ/\text{s}$, respectively.

III. OVERVIEW OF SLAMMOT

The SLAMMOT process is shown in Figure 3. First, distortion in LiDAR point cloud data caused by the motion of the micromobility and rider head is corrected. Next, the self-pose (i.e., three-dimensional (3D) position and attitude angle) of the rider helmet is calculated by Normal Distributions Transform (NDT) scan matching [17].

As shown in Figure 4, two coordinate systems are defined: world coordinate system ($O_w-x_wy_wz_w$) fixed to the ground and helmet coordinate system ($O_h-x_hy_hz_h$) fixed to the LiDAR. For simplicity, the helmet and LiDAR poses are considered to coincide. In the helmet coordinate system, a 3D voxel map with a cell size of 0.2 m per side is set. The LiDAR point cloud data acquired in one sampling period are mapped onto a voxel map and downsized using a voxel grid filter. In subsequent



Figure 1. Overview of the experimental helmet equipped with LiDAR and IMU.

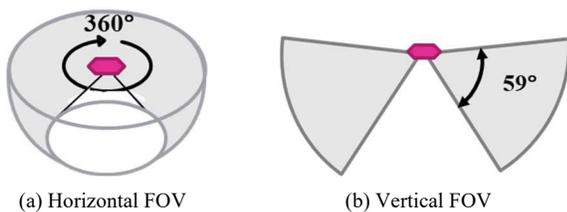


Figure 2. LiDAR FOV.

processing, the downsized point cloud data are used to estimate the helmet self-pose, and LiDAR point cloud data before downsizing are used for environment mapping and MOT.

For the i -th ($i = 1, 2, \dots, n$) measurement in LiDAR point cloud data, the coordinates in the world and helmet coordinate systems are denoted by $\mathbf{p}_{hi} = (x_{hi}, y_{hi}, z_{hi})^T$ and $\mathbf{p}_i = (x_i, y_i, z_i)^T$, respectively. Thus, the following relation is obtained:

$$\begin{pmatrix} \mathbf{p}_i \\ 1 \end{pmatrix} = \mathbf{T}(\mathbf{X}) \begin{pmatrix} \mathbf{p}_{hi} \\ 1 \end{pmatrix} \quad (1)$$

where \mathbf{X} indicates the position and attitude of the helmet, and $\mathbf{T}(\mathbf{X})$ denotes the corresponding homogeneous transformation matrix.

In SLAM using NDT scan matching, a 3D voxel map with a cell size of 0.6 m per side is set in the world coordinate system. By superimposing the LiDAR point cloud data obtained at current time t (referred to as current point cloud data) and those obtained up to the previous time ($t-1$) (referred to as reference map), the helmet self-pose \mathbf{X} at the current time is calculated. The current point cloud data are mapped onto the world coordinate system by performing a coordinate transformation using (1) and then merged into the reference map.

Because LiDAR scans a laser beam, all point cloud data within one LiDAR sampling period cannot be obtained at a single location when the micromobility is moving or the rider

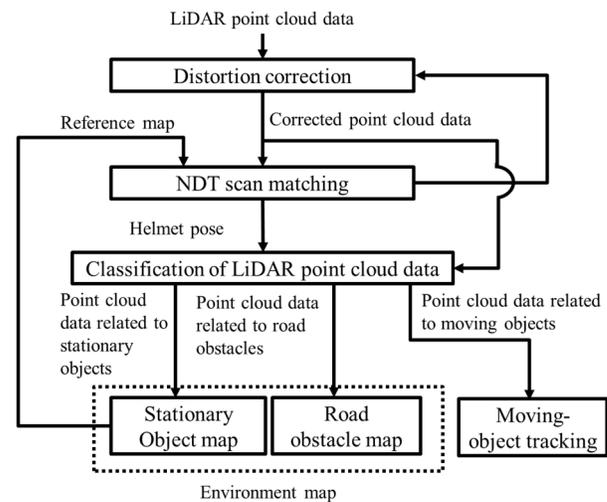


Figure 3. SLAMMOT process.

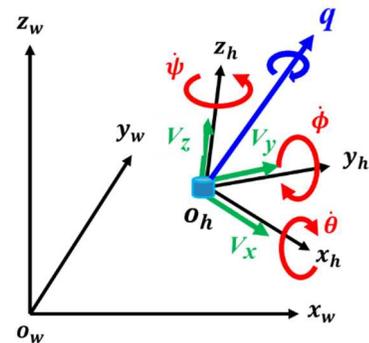


Figure 4. Notation related to helmet motion.

head is swinging. Therefore, if all point cloud data within one LiDAR sampling period is transformed using the pose information of the helmet at the same time, distortion arises in the LiDAR point cloud data mapped onto the world coordinate system using (1). As distortion causes inaccurate results in SLAMMOT, distortion correction of LiDAR point cloud data is required. The proposed distortion-correction method using a quaternion UKF is described in the next section.

Distortion-corrected LiDAR point cloud data are classified into measurements related to the road surface, road obstacles, stationary objects (e.g., buildings and trees) and moving objects (e.g., cars and pedestrians). Unevenness on road surfaces, such as obstacles on the road, ditches, and curbs, which can lead to falling accidents in micromobility, are detected as road obstacles. An environment map is built including stationary objects and road obstacles. LiDAR point cloud data related to moving objects (referred to as moving point cloud data) are used for MOT. The classification method is described in Section V.

MOT is performed using our previous method [18]. The shape of a moving object is represented by a cuboid. The width and length of the object are extracted from moving point cloud data using the rotating caliper method [19], and the height of the object is determined from the height information in the moving point cloud data. A Kalman filter is applied to estimate the two-dimensional (2D) position and velocity of the moving object in the world coordinate system based on the centroid position of the extracted cuboid. When applying the Kalman filter, the object is assumed to be moving at an approximately constant velocity. In crowded environments, the rule-based data association method [18] is used to accurately match multiple moving objects with corresponding moving point cloud data.

IV. DISTORTION CORRECTION OF LiDAR POINT CLOUD DATA

A. Overview

SLAMMOT is performed by mapping LiDAR point cloud data obtained in the helmet coordinate system onto the world coordinate system according to the helmet self-pose information. The self-pose is calculated every 120 ms (LiDAR sampling period) by NDT scan matching. However, all LiDAR point cloud data within the LiDAR sampling period cannot be captured simultaneously because LiDAR acquires measurements by scanning a laser beam. Consequently, when the micromobility is moving or the rider head is swinging, the LiDAR point cloud data mapped onto the world coordinate system are distorted.

The distortion in LiDAR point cloud data is corrected by estimating the helmet self-pose at every LiDAR data acquisition instant in 0.4 ms interval. Distortion correction is based on a quaternion UKF using the self-pose calculated by NDT scan matching every 120 ms, as well as the attitude angle and angular velocity acquired from the IMU every 10 ms.

B. State and Measurement Equations of Helmet

As shown in Figure 4, in the helmet coordinate system, the quaternion [20] is defined by $q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$, where \mathbf{i} , \mathbf{j} , and \mathbf{k} are the unit vectors along the x_h , y_h , and z_h axes, respectively. The translational velocity of the helmet along the x_h , y_h , and z_h axes is denoted by (V_x, V_y, V_z) . The angular velocity (i.e., roll, pitch, and yaw angular velocities) captured from the

IMU is denoted by $(\dot{\phi}, \dot{\theta}, \dot{\psi})$, and its bias is denoted by $(\dot{\phi}_{bias}, \dot{\theta}_{bias}, \dot{\psi}_{bias})$.

It is assumed that the translational velocity of the helmet is nearly constant in a short period. Hence, the state equation for helmet motion is given by:

$$\begin{pmatrix} x(t+1) \\ y(t+1) \\ z(t+1) \\ q_0(t+1) \\ q_1(t+1) \\ q_2(t+1) \\ q_3(t+1) \\ V_x(t+1) \\ V_y(t+1) \\ V_z(t+1) \\ \dot{\phi}_{bias}(t+1) \\ \dot{\theta}_{bias}(t+1) \\ \dot{\psi}_{bias}(t+1) \end{pmatrix} = \begin{pmatrix} x(t) + a_1(t)r_{11}(t) + a_2(t)r_{12}(t) + a_3(t)r_{13}(t) \\ y(t) + a_1(t)r_{21}(t) + a_2(t)r_{22}(t) + a_3(t)r_{23}(t) \\ z(t) + a_1(t)r_{31}(t) + a_2(t)r_{32}(t) + a_3(t)r_{33}(t) \\ q_0(t) \cos \frac{b_0(t)}{2} \\ -\tau(q_1(t) \frac{b_1(t)}{b_0(t)} + q_2(t) \frac{b_2(t)}{b_0(t)} + q_3(t) \frac{b_3(t)}{b_0(t)}) \sin \frac{b_0(t)}{2} \\ q_1(t) \cos \frac{b_0(t)}{2} \\ +\tau(q_0(t) \frac{b_1(t)}{b_0(t)} - q_3(t) \frac{b_2(t)}{b_0(t)} + q_2(t) \frac{b_3(t)}{b_0(t)}) \sin \frac{b_0(t)}{2} \\ q_2(t) \cos \frac{b_0(t)}{2} \\ +\tau(q_3(t) \frac{b_1(t)}{b_0(t)} + q_0(t) \frac{b_2(t)}{b_0(t)} - q_1(t) \frac{b_3(t)}{b_0(t)}) \sin \frac{b_0(t)}{2} \\ q_3(t) \cos \frac{b_0(t)}{2} \\ +\tau(-q_2(t) \frac{b_1(t)}{b_0(t)} + q_1(t) \frac{b_2(t)}{b_0(t)} + q_0(t) \frac{b_3(t)}{b_0(t)}) \sin \frac{b_0(t)}{2} \\ V_x(t) + \tau w_{i_x} \\ V_y(t) + \tau w_{i_y} \\ V_z(t) + \tau w_{i_z} \\ \dot{\phi}_{bias}(t) + w_{\dot{\phi}_{bias}} \\ \dot{\theta}_{bias}(t) + w_{\dot{\theta}_{bias}} \\ \dot{\psi}_{bias}(t) + w_{\dot{\psi}_{bias}} \end{pmatrix} \quad (2)$$

where (x, y, z) is the position of helmet in the world coordinate system. $a_1 = V_x \tau + \tau^2 w_{i_x} / 2$, $a_2 = V_y \tau + \tau^2 w_{i_y} / 2$, $a_3 = V_z \tau + \tau^2 w_{i_z} / 2$, $b_0 = \tau \sqrt{b_1^2 + b_2^2 + b_3^2}$, $b_1 = \dot{\phi} + \dot{\phi}_{bias} + w_{\dot{\phi}}$, $b_2 = \dot{\theta} + \dot{\theta}_{bias} + w_{\dot{\theta}}$, and $b_3 = \dot{\psi} + \dot{\psi}_{bias} + w_{\dot{\psi}}$. ($w_{i_x}, w_{i_y}, w_{i_z}, w_{\dot{\phi}}, w_{\dot{\theta}}, w_{\dot{\psi}}, w_{\dot{\phi}_{bias}}, w_{\dot{\theta}_{bias}}, w_{\dot{\psi}_{bias}}$) indicate disturbances (plant noise). τ is the sampling period. r_{mn} ($m, n = 1, 2, 3$) is element (m, n) of the following rotation matrix:

$$\mathbf{R} = \begin{pmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{pmatrix} \quad (3)$$

Noted that the angular velocities from the IMU are considered as a system input in (2).

The attitude (i.e., roll and pitch angles) of the helmet, which is obtained from the IMU every 10 ms, is denoted by $z_{IMU}(t)$. The measurement equation of $z_{IMU}(t)$ is given by

$$z_{IMU}(t) = \begin{pmatrix} \arctan \frac{r_{32}(t)}{r_{33}(t)} \\ \arcsin(-r_{31}(t)) \end{pmatrix} + \Delta z_{IMU}(t) \quad (4)$$

where $\Delta \mathbf{z}_{IMU}$ represents the measurement noise.

The helmet self-pose obtained by NDT scan matching every 120 ms is denoted by $\mathbf{z}_{NDT}(t)$. The measurement equation of $\mathbf{z}_{NDT}(t)$ is expressed as

$$\mathbf{z}_{NDT}(t) = \begin{pmatrix} x(t) \\ y(t) \\ z(t) \\ \arctan \frac{r_{32}(t)}{r_{33}(t)} \\ \arcsin(-r_{31}(t)) \\ \arctan \frac{r_{21}(t)}{r_{11}(t)} \end{pmatrix} + \Delta \mathbf{z}_{NDT}(t) \quad (5)$$

where $\Delta \mathbf{z}_{NDT}$ represents the measurement noise.

Equations (2), (4), and (5) are represented in the vector form as follows:

$$\xi^{(t+1)} = \mathbf{f}[\xi^{(t)}, \mathbf{u}(t), \mathbf{w}(t)] \quad (6)$$

$$\mathbf{z}_{IMU}(t) = \mathbf{h}_{IMU}[\xi^{(t)}] + \Delta \mathbf{z}_{IMU}(t) \quad (7)$$

$$\mathbf{z}_{NDT}(t) = \mathbf{h}_{NDT}[\xi^{(t)}] + \Delta \mathbf{z}_{NDT}(t) \quad (8)$$

where $\xi = (x, y, z, q_0, q_1, q_2, q_3, q_4, V_x, V_y, V_z, \hat{\phi}_{bias}, \hat{\theta}_{bias}, \hat{\psi}_{bias})^T$, $\mathbf{u} = (\hat{\phi}, \hat{\theta}, \hat{\psi})^T$, and $\mathbf{w} = (w_{i_x}, w_{i_y}, w_{i_z}, w_{\hat{\phi}}, w_{\hat{\theta}}, w_{\hat{\psi}}, w_{\hat{\phi}_{bias}}, w_{\hat{\theta}_{bias}}, w_{\hat{\psi}_{bias}})^T$.

C. Distortion Correction Using Quaternion UKF

The process of distortion correction of LiDAR point cloud data is shown in Figure 5. The LiDAR sampling period of 120 ms is denoted by τ . The IMU sampling period of 10 ms and LiDAR data acquisition period of 0.48 ms are denoted by τ_{IMU} and $\Delta\tau$, respectively. Hence, $\tau = 12\tau_{IMU}$ and $\tau_{IMU} = 21\Delta\tau$.

Distortion in LiDAR point cloud data between times $t\tau$ and $(t+1)\tau$, where $t = 0, 1, \dots$, is corrected in the following five steps:

Step 1. State prediction in IMU sampling period τ_{IMU}

The state estimate and its error covariance at time $t\tau + k\tau_{IMU}$, where $k = 0, \dots, 11$, are denoted by $\hat{\xi}^{(k)}(t)$ and $\mathbf{\Xi}^{(k)}(t)$, respectively. As the dimensions of state variable ξ and plant noise \mathbf{w} in state equation (6) are 13 and 9, respectively, the following 22-dimensional augmented system of $\hat{\xi}^{(k)}(t)$ and $\mathbf{\Xi}^{(k)}(t)$ is defined:

$$\hat{\xi}^a(t) = [\hat{\xi}^{(k)}(t)^T, \mathbf{0}^T]^T \quad (9)$$

$$\mathbf{\Xi}^a(t) = \begin{bmatrix} \mathbf{\Xi}^{(k)}(t) & \mathbf{0} \\ \mathbf{0} & \mathbf{Q} \end{bmatrix} \quad (10)$$

where \mathbf{Q} is the covariance of plant noise \mathbf{w} .

From (9) and (10), 45 sigma points are calculated as follows:

$$\left. \begin{aligned} \chi_0(t) &= \hat{\xi}^a(t) \\ \chi_i(t) &= \hat{\xi}^a(t) + \sqrt{22 + \lambda} \left(\sqrt{\mathbf{\Xi}^a(t)} \right)_i \quad (i = 1, 2, \dots, 22) \\ \chi_i(t) &= \hat{\xi}^a(t) - \sqrt{22 + \lambda} \left(\sqrt{\mathbf{\Xi}^a(t)} \right)_{i-22} \quad (i = 23, 24, \dots, 44) \end{aligned} \right\} \quad (11)$$

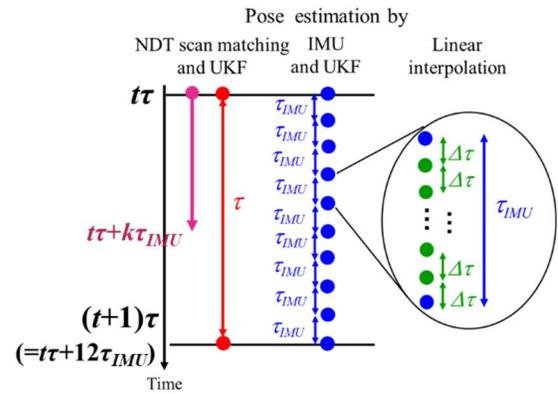


Figure 5. Process of distortion correction.

where $(\sqrt{\mathbf{\Xi}^a(t)})_i$ and $(\sqrt{\mathbf{\Xi}^a(t)})_{i-22}$ are the i -th and $(i-22)$ th column vectors, respectively, of the square root of $\mathbf{\Xi}^a(t)$. Hyperparameter λ is set to 2 in this study.

The state prediction at time $t\tau + (k+1)\tau_{IMU}$ for the sigma points is calculated as

$$\chi_i^{\xi(k+1/k)}(t) = \mathbf{f}[\chi_i^{\xi(k)}(t), \chi_i^{w(k)}(t), \mathbf{u}(t)] \quad (12)$$

where $\chi_i^{\xi(k)}(t)$ and $\chi_i^{w(k)}(t)$ are the components of the 13-dimensional state variable and 9-dimensional plant noise, respectively, of the 22-dimensional sigma points obtained in (11).

Therefore, state prediction $\hat{\xi}^{(k+1/k)}(t)$ and its error covariance $\mathbf{\Xi}^{(k+1/k)}(t)$ at time $t\tau + (k+1)\tau_{IMU}$ are given by

$$\left. \begin{aligned} \hat{\xi}^{(k+1/k)}(t) &= \sum_{i=0}^{44} \mu_i \chi_i^{\xi(k+1/k)}(t) \\ \mathbf{\Xi}^{(k+1/k)}(t) &= \sum_{i=0}^{44} \mu_i \left[\chi_i^{\xi(k+1/k)}(t) - \hat{\xi}^{(k+1/k)}(t) \right] [\bullet]^T \end{aligned} \right\} \quad (13)$$

where $\mu_0 = \lambda / (22 + \lambda)$ and $\mu_i = \lambda / (44 + 2\lambda)$ ($i \neq 0$).

Step 2. State estimate using angular information from IMU

Attitude angle \mathbf{z}_{IMU} is obtained from the IMU at time $t\tau + (k+1)\tau_{IMU}$. Then, the measurement prediction at time $t\tau + (k+1)\tau_{IMU}$ for sigma points in (11) is calculated as

$$\zeta_{IMU_i}^{(k+1/k)}(t) = \mathbf{h}_{IMU}[\chi_i^{\xi(k+1/k)}(t)] \quad (14)$$

The measurement prediction and its error covariance at time $t\tau + (k+1)\tau_{IMU}$ are given by

$$\left. \begin{aligned} \zeta_{IMU}^{(k+1/k)}(t) &= \sum_{i=0}^{44} \mu_i \zeta_{IMU_i}^{(k+1/k)}(t) \\ \mathbf{Z}_{IMU}^{(k+1/k)}(t) &= \sum_{i=0}^{44} \mu_i \left[\zeta_{IMU_i}^{(k+1/k)}(t) - \zeta_{IMU}^{(k+1/k)}(t) \right] [\bullet]^T + \mathbf{R}_{IMU} \end{aligned} \right\} \quad (15)$$

where \mathbf{R}_{IMU} is the covariance of measurement noise $\Delta \mathbf{z}_{IMU}$.

The state estimate and its error covariance are then given by

$$\left. \begin{aligned} \hat{\xi}^{(k+1)}(t) &= \hat{\xi}^{(k+1/k)}(t) + \mathbf{K}(t) \left[\mathbf{z}_{IMU}^{(k+1)}(t) - \zeta_{IMU}^{(k+1/k)}(t) \right] \\ \mathbf{\Xi}^{(k+1)}(t) &= \mathbf{\Xi}^{(k+1/k)}(t) - \mathbf{K}(t) \mathbf{Z}_{IMU}^{(k+1/k)}(t) \mathbf{K}(t)^T \end{aligned} \right\} \quad (16)$$

where Kalman gain \mathbf{K} is expressed as

$$\mathbf{K}(t) = \sum_{i=0}^{44} \mu_i \left[\chi_i^{\xi^{(k+1/k)}(t)} - \hat{\xi}^{(k+1/k)}(t) \right] \left[\zeta_{IMU_i}^{(k+1/k)}(t) - \zeta_{IMU}^{(k+1/k)}(t) \right]^T (\mathbf{Z}_{IMU}^{(k+1/k)}(t))^{-1} \quad (17)$$

Of the state estimate $\hat{\xi}^{(k+1)}(t)$, the state estimate for the helmet self-pose is denoted by $\hat{\mathbf{X}}^{(k+1)}(t)$.

Step 3. State prediction in LiDAR observation period $\Delta\tau$

Using self-poses $\hat{\mathbf{X}}^{(k)}(t)$ and $\hat{\mathbf{X}}^{(k+1)}(t)$, which are estimated at $t\tau + k\tau_{IMU}$ and $t\tau + (k+1)\tau_{IMU}$, respectively, self-pose $\hat{\mathbf{X}}^{(k)}(t, j)$ at $t\tau + k\tau_{IMU} + j\Delta\tau$ ($j = 1-21$) is given by the following interpolation formula:

$$\hat{\mathbf{X}}^{(k)}(t, j) = \hat{\mathbf{X}}^{(k)}(t) + \frac{\hat{\mathbf{X}}^{(k+1)}(t) - \hat{\mathbf{X}}^{(k)}(t)}{\tau_{IMU}} j \Delta\tau \quad (18)$$

Step 4. Coordinate transformation of LiDAR point cloud data

The coordinates of the i -th measurement of LiDAR point cloud data obtained at $t\tau + k\tau_{IMU} + j\Delta\tau$ are denoted by $\mathbf{p}_{hi}^{(k)}(t, j)$ in the helmet coordinate system and by $\mathbf{p}_i^{(k)}(t, j)$ in the world coordinate system. $\mathbf{p}_{hi}^{(k)}(t, j)$ can be transformed into $\mathbf{p}_i^{(k)}(t, j)$ based on $\hat{\mathbf{X}}^{(k)}(t, j)$ and (1) as follows:

$$\begin{pmatrix} \mathbf{p}_i^{(k)}(t, j) \\ 1 \end{pmatrix} = \mathbf{T}(\hat{\mathbf{X}}^{(k)}(t, j)) \begin{pmatrix} \mathbf{p}_{hi}^{(k)}(t, j) \\ 1 \end{pmatrix} \quad (19)$$

Based on helmet self-pose $\hat{\mathbf{X}}^{(12)}(t)$ obtained at time $(t+1)\tau$ ($= t\tau + 12\tau_{IMU}$), $\mathbf{p}_i^{(k)}(t, j)$ is transformed into $\mathbf{p}_{hi}^*(t+1)$ at $(t+1)\tau$ as follows:

$$\begin{pmatrix} \mathbf{p}_{hi}^*(t+1) \\ 1 \end{pmatrix} = \mathbf{T}(\hat{\mathbf{X}}^{(12)}(t))^{-1} \begin{pmatrix} \mathbf{p}_i^{(k)}(t, j) \\ 1 \end{pmatrix} \quad (20)$$

The above equation means that the coordinates of LiDAR point cloud data obtained between times $t\tau$ and $(t+1)\tau$ can be transformed into those obtained at time $(t+1)\tau$.

Step 5. State estimate using self-pose by NDT scan matching in LiDAR sampling period τ

LiDAR point cloud data corrected in step 4 are used as current point cloud data at $(t+1)\tau$, and helmet self-pose \mathbf{z}_{NDT} is calculated using NDT scan matching. Based on (16), state estimate $\hat{\xi}^{(12)}(t)$ and its error covariance $\Xi^{(12)}(t)$ at time $(t+1)\tau$ are obtained using IMU information. The state estimate and its error covariance are considered as a priori information, and the helmet state is estimated using pose measurement \mathbf{z}_{NDT} at time $(t+1)\tau$.

First, 27 sigma points are obtained as follows:

$$\left. \begin{aligned} \chi_{NDT_0}(t+1) &= \hat{\xi}^{(12)}(t) \\ \chi_{NDT_i}(t+1) &= \hat{\xi}^{(12)}(t) + \sqrt{13+\lambda} \left(\sqrt{\Xi^{(12)}(t)} \right)_i \quad (i=1, 2, \dots, 13) \\ \chi_{NDT_i}(t+1) &= \hat{\xi}^{(12)}(t) - \sqrt{13+\lambda} \left(\sqrt{\Xi^{(12)}(t)} \right)_{i-13} \quad (i=14, 15, \dots, 26) \end{aligned} \right\} \quad (21)$$

Then, the measurement prediction at time $(t+1)\tau$ for the sigma points in (21) is calculated by:

$$\zeta_{NDT_i}(t+1) = \mathbf{h}_{NDT}[\chi_{NDT_i}(t+1)] \quad (22)$$

The measurement prediction and its error covariance at time $(t+1)\tau$ are given by

$$\left. \begin{aligned} \zeta_{NDT}(t+1) &= \sum_{i=0}^{26} \mu_i \zeta_{NDT_i}(t+1) \\ \mathbf{Z}_{NDT}(t+1) &= \sum_{i=0}^{26} \mu_i \left[\zeta_{NDT_i}(t+1) - \zeta_{NDT}(t+1) \right] \left[\bullet \right]^T + \mathbf{R}_{NDT} \end{aligned} \right\} \quad (23)$$

where $\mu_0 = \lambda / (13 + \lambda)$ and $\mu_i = \lambda / (26 + 2\lambda)$ ($i \neq 0$). \mathbf{R}_{NDT} is the covariance of the measurement error $\Delta\mathbf{z}_{NDT}$.

The state estimate and its error covariance are then given by

$$\left. \begin{aligned} \hat{\xi}^{(0)}(t+1) &= \hat{\xi}^{(12)}(t) + \mathbf{K}(t+1) \left[\mathbf{z}_{NDT}(t+1) - \zeta_{NDT}(t+1) \right] \\ \Xi^{(0)}(t+1) &= \Xi^{(12)}(t) - \mathbf{K}(t+1) \mathbf{Z}_{NDT}(t+1) \mathbf{K}(t+1)^T \end{aligned} \right\} \quad (24)$$

where Kalman gain \mathbf{K} is expressed as

$$\mathbf{K}(t+1) = \sum_{i=0}^{26} \mu_i \left[\chi_{NDT_i}(t+1) - \hat{\xi}^{(12)}(t) \right] \left[\zeta_{NDT_i}(t+1) - \zeta_{NDT}(t+1) \right]^T (\Xi^{(12)}(t))^{-1} \quad (25)$$

V. CLASSIFICATION OF LiDAR POINT CLOUD DATA

The current LiDAR point cloud data contain various measurements related to road surfaces, road obstacles, stationary objects, and moving objects. Therefore, they are classified, and the measurements related to stationary objects and road obstacles are used to build an environment map. The measurements related to moving objects are used for MOT.

First, the current LiDAR point cloud data are classified into measurements related to the road surface, objects, and road obstacles, such as curbs and falling objects, using a ground-plane fitting method [21].

In the helmet coordinate system, a 2D polar grid map is set, as shown in Figure 6. LiDAR point cloud data are mapped onto the grid map. The cell size in the grid map depends on the distance from the LiDAR, such that the number of LiDAR point cloud datapoints occupied in each cell is comparable.

In each cell, 20 LiDAR measurements with the lowest heights are extracted as candidate measurements related to road surfaces. Then, by applying principal component analysis to the candidate measurements, the plane represented by the following equation is estimated:

$$A(x - x_g) + b(y - y_g) + c(z - z_g) = 0 \quad (26)$$

where (a, b, c) is the eigenvector of the third principal

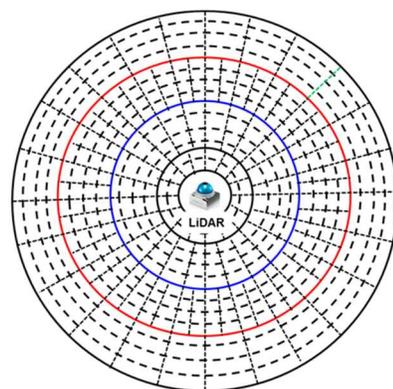


Figure 6. 2D polar grid map.

component for the candidate measurements in each cell, and (x_g, y_g, z_g) is the geometrical center of the candidate measurements.

The normal distance L of each LiDAR point cloud datapoint to the estimated plane is calculated, and LiDAR point cloud data are classified as follows:

- $L < 0.1$ m: LiDAR measurements related to road surfaces,
- $0.1 \text{ m} \leq L < 0.25$ m: LiDAR measurements related to road obstacles,
- $L \geq 0.25$ m: LiDAR measurements related to objects.

Then, the LiDAR measurements related to road obstacles are used to build a road obstacle map.

LiDAR measurements related to objects extracted above comprise measurements related to stationary and moving objects. Therefore, the occupancy grid method is used to further classify the measurements related to objects into those of stationary and moving objects.

A 2D orthogonal grid map (elevation map) with a cell size of 0.3 m per side is set in the world coordinate system. LiDAR measurements related to objects are mapped onto the elevation map. LiDAR measurements related to moving objects occupy the same cells for a short time, while those related to stationary objects occupy the same cells for a long time. Therefore, LiDAR measurements related to stationary and moving objects can be classified by measuring the cell occupancy time [18]. In this study, the threshold of occupancy time is set to 0.8 s.

Then, LiDAR point cloud data related to stationary and moving objects are used for SLAM and MOT, respectively.

VI. FUNDAMENTAL EXPERIMENTS

An environment map is built by driving a micromobility (bicycle) on a roadway at our university campus, as shown in Figure 7. The distance traveled of the micromobility is 450 m, and its maximum speed is 15 km/h. At the locations indicated by the blue and yellow circles in Figure 7 (a), the rider moves

his head in the right-left and rearward directions, respectively. At the location indicated by the green circle, the rider lowers his head to pick up an object placed on the road. In the experiments, LiDAR point cloud data are recorded, and SLAMMOT is executed offline on a laptop computer.

Figure 8 shows the mapping results. The environment map is properly built the proposed method. To evaluate the mapping performance, experiments in the following three conditions are conducted.

- Condition 1: Mapping using quaternion-UKF-based distortion correction (proposed method),
- Condition 2: Mapping using Euler angle-UKF-based distortion correction (previous method in [15]),
- Condition 3: Mapping without distortion correction.

The performance of SLAM-based mapping is equivalent to that of self-pose estimation. Therefore, the error in the helmet self-position estimate at the goal position is obtained when the micromobility is driven. The micromobility was driven three times under each condition. Table I lists the results. The proposed method (condition 1) can build an environment map more accurately than the methods evaluated in conditions 2 and 3.

The micromobility was moved six times along the path shown in Figure 7 (a). Then, 219 moving objects (209 pedestrians and 10 cars) were tracked. Table II shows the tracking result: the number of correct and incorrect tracking. From the results, our proposed method (condition 1) achieves the highest MOT accuracy. The reason for the false tracking is that the safety confirmation by the rider causes a large posture change in his head, which prevents accurate mapping of stationary point cloud data. Untracked objects are all people. This is due to the inability to both distinguish between people in close proximity and recognize pedestrians due to occlusions by trees and shrubbery, as shown in Figure 7 (b).

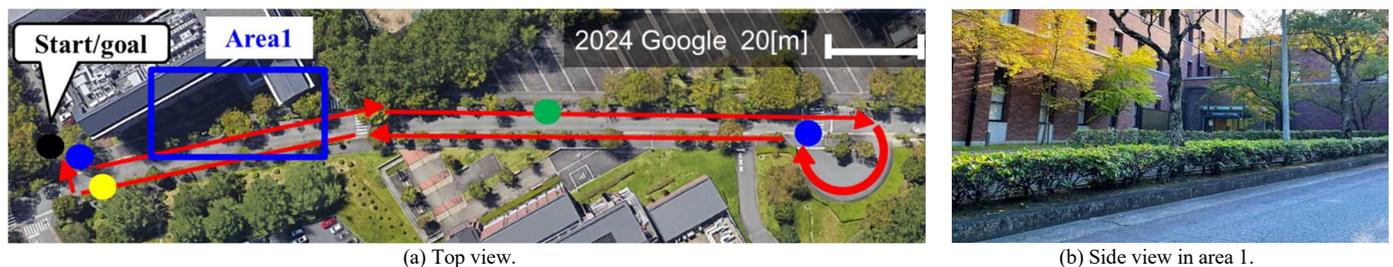


Figure 7. Photo of experimental environment. In (a), the red line indicates movement path of micromobility in roadway.

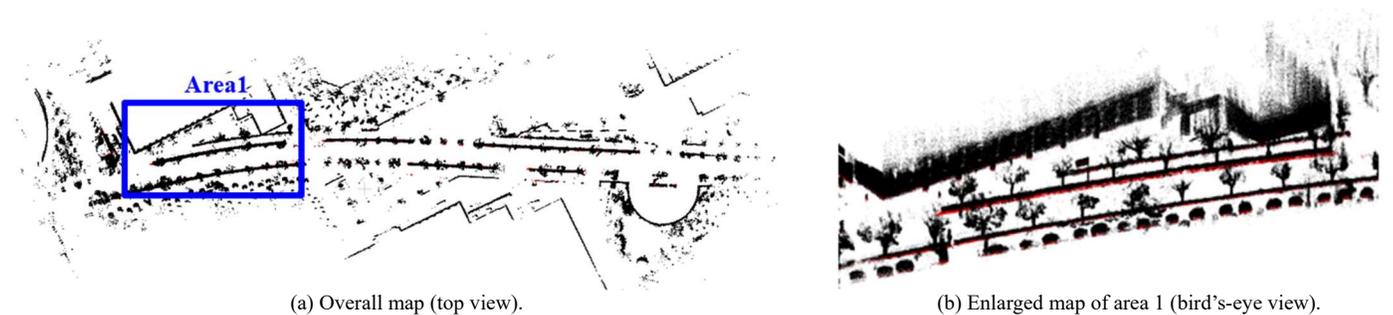


Figure 8. Mapping results. The black and red dots indicate LiDAR point cloud data related to stationary objects and road obstacles, respectively.

TABLE I. ERROR OF POSITION ESTIMATE OF HELMET AT GOAL POSITION

	Condition 1	Condition 2	Condition 3
Run 1	1.57 m	2.32 m	3.01 m
Run 2	0.43 m	0.91 m	1.52 m
Run 3	2.16 m	2.19 m	23.8 m

TABLE II. NUMBER OF CORRECT AND INCORRECT TRACKING

	Condition 1	Condition 2	Condition 3
Correct tracking	198	167	161
False tracking	41	58	73
Untracking	21	52	58

VII. CONCLUSION AND FUTURE WORK

This paper presented a SLAMMOT method using a small and lightweight solid-state LiDAR attached to the rider helmet of a micromobility. To improve the performance of SLAMMOT during motion of micromobility and rider's head, the distortion in LiDAR point cloud data was corrected using a quaternion-UKF-based method. Fundamental experiments conducted at our university campus confirmed the effectiveness of the proposed distortion correction method compared to the conventional Euler angle-UKF-based method.

In this paper, the SLAMMOT experiments were confined to a controlled environment. Future studies will evaluate SLAMMOT accuracy under varying intensities of rider's head motions and in more diverse urban environments with higher traffic. Since a single motion model (i.e., constant velocity model) of target objects was assumed in MOT, tracking performance degrades when object motion suddenly change, such as during sudden starts or stops. To improve the MOT performance, an interacting multimodel estimator will be implemented. In addition, improving the mapping accuracy will be considered based on the fusion of SLAM-based environment maps built by many micromobilities.

In the experiments, LiDAR point cloud data were recorded, and SLAMMOT was executed offline on a laptop computer. Since micromobility applications require energy- and processing-efficient solutions, the computational cost (e.g., processing time and energy consumption) of SLAMMOT should be considered to assess feasibility in embedded systems.

ACKNOWLEDGMENT

This study was partially supported by the KAKENHI Grant #23K03781, the Japan Society for the Promotion of Science (JSPS).

REFERENCES

- [1] Y. F. Payalan and M. A. Guvensan, "Towards Next-Generation Vehicles Featuring the Vehicle Intelligence," *IEEE Trans. on Intelligent Transportation Systems*, vol. 21, pp. 30–46, 2020.
- [2] E. Arnold, et al., "A Survey on 3D Object Detection Methods for Autonomous Driving Applications," *IEEE Trans. on Intelligent Transportation Systems*, vol. 20, pp. 3782–3795, 2019.
- [3] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, "A Survey of Autonomous Driving: Common Practices and Emerging Technologies," *IEEE Access*, vol. 8, pp. 58443–58469, 2020.
- [4] G. Bresson, Z. Alsayed, L. Yu, and S. Glaser, "Simultaneous Localization And Mapping: A Survey of Current Trends in Autonomous Driving," *IEEE Trans. on Intelligent Transportation Systems*, vol. 2, pp. 194–220, 2017.
- [5] B. Madapur, S. Madangopal, and M. N. Chandrashekar, "Micro-Mobility Infrastructure for Redefining Urban Mobility," *European J. of Engineering Science and Technology*, vol. 3, pp. 71–85, 2020.
- [6] I. Yoshida, A. Yoshida, M. Hashimoto, and K. Takahashi, "Environmental Map Building and Moving Object Tracking Using Helmet-Mounted LiDAR and IMU for Micromobility," *Int. J. on Advances in Systems and Measurements*, vol. 16, pp. 40–52, 2023.
- [7] T. Raj, F. H. Hashim, A. B. Huddin, M. F. Ibrahim, and A. Hussain, "A Survey on LiDAR Scanning Mechanisms," *Electronics* 9, 741, 2020.
- [8] K. Li, M. Li, and U. D. Hanebeck, "Towards High-Performance Solid-State-LiDAR-Inertial Odometry and Mapping," *IEEE Robotics and Automation Letters*, vol. 6, pp. 5167–5174, 2021.
- [9] V. Kumar, S. C. Subramanian, and R. Rajamani, "A Novel Algorithm to Track Closely Spaced Road Vehicles Using a Low Density Flash Lidar," *Signal Processing* 191, pp. 1–11, 2022.
- [10] J. Li, et al., "WHU-Helmet: A Helmet-Based Multisensor SLAM Dataset for the Evaluation of Real-Time 3-D Mapping in Large-Scale GNSS-Denied Environments," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 61, pp. 1–16, 2023.
- [11] Z. Peng, Z. Xiong, Y. Zhao, and L. Zhang, "3-D Objects Detection and Tracking Using Solid-State LiDAR and RGB Camera," *IEEE Sensors Journal*, vol. 23, pp. 14795–14808, 2023.
- [12] S. Hong, H. Ko, and J. Kim, "VICP: Velocity Updating Iterative Closest Point Algorithm," *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp. 1893–1898, 2010.
- [13] P. Zhou, X. Guo, X. Pei, and C. Chen, "T-LOAM: Truncated Least Squares LiDAR-only Odometry and Mapping in Real Time," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 60, pp. 1–13, 2022.
- [14] I. Yoshida, A. Yoshida, M. Hashimoto, and K. Takahashi, "Map Building Using Helmet-Mounted LiDAR for Micromobility," *Artificial Life and Robotics*, vol. 28, pp. 471–482, 2023.
- [15] R. Nakamura, I. Inaga, M. Hashimoto, and K. Takahashi, "SLAM-Based Mapping Using Micromobility-Mounted Solid-State LiDAR," *Proc. the 11th IIAE Int. Conf. on Intelligent Systems and Image Processing*, pp. 93–100, 2024.
- [16] E. A. Wan and R. van der Merwe, "The Unscented Kalman Filter: Kalman Filtering and Neural Networks," S. Haykin, Eds, Wiley Publishing, 2001.
- [17] P. Biber and W. Strasser, "The Normal Distributions Transform: A New Approach to Laser Scan Matching," *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2743–2748, 2003.
- [18] M. Hashimoto, R. Izumi, Y. Tamura, and K. Takahashi, "Laser-based Tracking of People and Vehicles by Multiple Mobile Robots," *Proc. the 11th Int. Conf. on Informatics in Control, Automation and Robotics*, pp. 522–527, 2014.
- [19] G. Toussaint, "Solving Geometric Problems with the Rotating Calipers," *Proc. IEEE Mediterranean Electrotechnical Conf.*'83, pp. 1–8, 1983.
- [20] Y. B. Jia, "Quaternions and Rotations," Available from: <https://www.fuchs-braun.com/media/56347a973e30256ffff802effffff1.pdf>.
- [21] H. Lim, M. Oh, and H. Myung, "Patchwork: Concentric Zone-based Region-wise Ground Segmentation with Ground Likelihood Estimation Using a 3D LiDAR Sensor," *IEEE Robotics and Automation Letters*, vol. 6, pp. 6458–6465, 2021.