IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

State Estimation and Environment Recognition for Articulated Structures via Proximity Sensors Distributed Over the Whole Body

Kengo Iwao, Hikaru Arita, *Member, IEEE*, and Kenji Tahara, *Member, IEEE*

***Abstract*—For robots with low rigidity, determining the robot's state based solely on kinematics is challenging. This is particularly crucial for a robot whose entire body is in contact with the environment, as accurate state estimation is essential for environmental interaction. We propose a method for simultaneous articulated robot posture estimation and environmental mapping by integrating data from proximity sensors distributed over the whole body. Our method extends the discrete-time model, typically used for state estimation, to the spatial direction of the articulated structure. The simulations demonstrate that this approach significantly reduces estimation errors.**

***Index Terms*—SLAM, sensor fusion, modeling, control, and learning for soft robots.**

I. Introduction

THE posture of an articulated robot can generally be determined from the joint angles via kinematics. This is true for robots in which each link is rigid and each joint angle measurement is accurate, such as industrial manipulators. However, not all modern robots have such characteristics. For example, the number of robots that can perform detailed tasks by learning with inexpensive hardware [1], [2], [3] and lightweight arms designed to be mounted on mobile robots [2], [4] have increased in recent years. The lightweight and inexpensive features of such robots mean that the rigidity of each link and the accuracy of joint angle measurements tend to be lower than in previous robots. When the deformation of these less rigid links and the angular errors of the joints are considered, kinematics alone cannot accurately estimate the posture, which can be an important problem in situations that require detailed work.

Research has been done on methods of ensuring accurate end-effector positioning, including end-effector position correction through marker observation [5], [6] and arm tracking through depth images from cameras mounted separately from the joints [7]. However, there are situations in which tracking an end-effector is not sufficient. A robot that moves within the environment and has an articulated structure, such as in [2], [4], needs information of external information regarding its entire body and whole-body state to the environment because the entire body may come into contact with the environment. These issues are equally applicable to snake robots. Snake robots navigate in unknown environments by maintaining full-body contact, making it crucial for them to ascertain their own postures relative to the environment. Furthermore, when traversing uneven terrain, these robots often lift parts of their bodies while maintaining contact with the environment. Consequently, lightweight links are frequently employed, many of which are prone to deformation. For situations described above, which involve uncertainties in a robot's pose and require precise relative pose estimation within unknown environments, Simultaneous Localization and Mapping (SLAM) is known as one of the effective approaches.

The SLAM method is generally used for mobile robots, but several studies have applied SLAM to robot arms to address the uncertainty of the state of a robot due to factors such as gear backlash and nonrigid deformation. For example, ARM-SLAM [8] uses a depth camera attached to the end-effector to perform SLAM, which reduces uncertainty and simultaneously yields information on the external environment. In addition, a method has been proposed to attach RGB-D cameras to multiple joints of a soft robot and perform SLAM to estimate the robot's configuration [9]. While these studies have successfully reduced angular errors in the arms, these cameras cannot obtain information at close range because their field of view is completely obstructed when the camera is too close to the environment, and is not suitable for the situations involving contacts. Moreover, acquiring information about the whole-body surroundings is difficult with a camera due to its limitations in focus, angle of view, and mounting position.

As proximity perception information for motions involving contact, proximity sensor data are utilized as visual information [10]. Because these sensors are small and lightweight, they can be placed on the robot's entire body to acquire the environmental information surrounding the entire body. There is increasing research on attaching proximity sensors to link surfaces and using external information from the robot surface. For example, information obtained from proximity sensors that cover the surface of a link has been used to perform collision avoidance for a robot [11], [12], [13]. Some studies have also

Received 18 September 2024; accepted 22 January 2025. Date of publication 5 February 2025; date of current version 18 February 2025. This article was recommended for publication by Associate Editor W. Shan and Editor Y.-L. Park upon evaluation of the reviewers' comments. This work was supported by JSPS KAKENHI under Grant JP23K20923 and Grant JP24H00726. *(Corresponding author: Hikaru Arita.)*

The authors are with the Department of Mechanical Engineering, Kyushu University, Fukuoka 819-0395, Japan (e-mail: iwao@hcr.mech.kyushu-u.ac.jp; arita@ieee.org; tahara@ieee.org).

This article has supplementary downloadable material available at https://doi.org/10.1109/LRA.2025.3539117, provided by the authors.

Digital Object Identifier 10.1109/LRA.2025.3539117

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

used proximity sensors attached to each joint of a snake robot to detect planes in the area of the entire body [14]. Utilizing the idea of these research, we focus on obtaining observation data for whole-body SLAM from optical proximity sensors distributed across the entire body of the robot.

By distributing sensors over the whole body, each link can have its own external information, allowing SLAM to be performed individually for each link. Using this feature, we propose extending the discrete-time model of SLAM in the spatial direction by recursive estimation of the whole body. In general, SLAM uses the idea of the Bayesian filter, which discretizes the continuous motion of the robot, for estimating the robot's motion. We focus on applying this discrete model to articulated robot structures. By recursively describing the state of each articulated link at the same time, state variables and their uncertainties can be propagated along the spatial direction. This enables the cumulative errors that occur with each successive link to be reduced.

In summary, the statement of our problem and the corresponding proposal to solve it are as follows:

- Problem Statement
  - For articulated robots and soft robots that are constructed with nonrigid components for the purpose of weight reduction and simplification, it is difficult to determine their states solely through kinematics.
  - Moreover, as these robots are often utilized in complex environments where full-body contact with their surroundings is likely, it is essential to determine the posture of the whole body relative to the environment.
- Proposed Approach
  - To address these issues, we propose a method for estimating a robot's state relative to its environment by distributing proximity sensors across the entire body of the robot and performing SLAM on observations from the full body.
- Key Innovation
  - We reduce the accumulation of errors by extending the structure of the discrete-time model used in SLAM to the spatial direction along the links.

We first explain our proposed method in Section II. Simulations for validating the proposed method are described in Section III. Finally, the advantages of the proposed method obtained from the simulations, as well as its applicability, are discussed in Section IV, and Section V concludes this paper.

TABLE I
SYMBOL DESCRIPTION

$\mathbf{x}_{i,k}$	state of the ith link at the kth step. $i = 0$ is the state of the root.
$\bar{\mathbf{x}}$	final estimated state
$\tilde{\mathbf{x}}$	error state with respect to the true state
$\hat{\mathbf{x}}^{\kappa}$	state obtained in the κth iterated Kalman filter. $\hat{\mathbf{x}}^{0}$ prior estimated state.
$\tilde{\mathbf{x}}^{\kappa}$	error state between $\hat{\mathbf{x}}^{\kappa}$ and $\hat{\mathbf{x}}^{\kappa+1}$
$\tilde{\mathbf{x}}^{0,\kappa}$	error state of $\hat{\mathbf{x}}^{0}$ with respect to $\hat{\mathbf{x}}^{\kappa}$.
$\mathbf{p}_j$	jth point measurement from the sensor for a single estimation.
$\mathbf{q}_j$	point on the map corresponding to $\mathbf{p}_j$
L, W	link frame and world frame

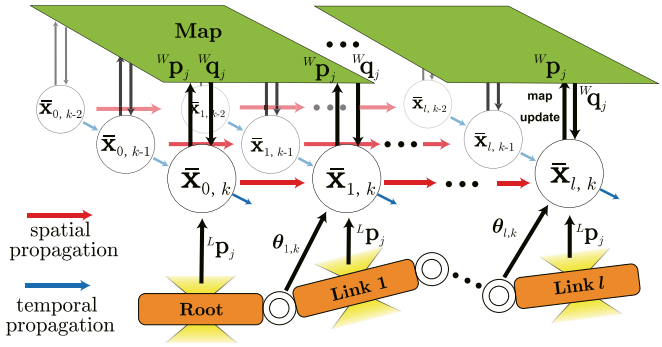

Fig. 1. Overview of the method. θ is the angle obtained from the encoder.

II. PROPOSED METHOD

A. Mathematical Notation

To describe the estimation methods, this paper uses several symbols, as shown in Table I. Furthermore, we define a single full-body estimation process at a given time as a "step".

B. Problem Statement

For simplicity, a common and straightforward model of a articulated structure is considered in this paper, as illustrated in Fig. 1. The root is the reference link of an articulated structure, such as the base of a manipulator or the head of a snake-like robot. The rotational axis of each joint can be defined arbitrarily, allowing the entire structure to perform three-dimensional motion. The origin of each link frame is defined as the center of the joint on the root side where the link is connected. The joint angle can be obtained from the encoders at each joint, which includes bias, and external information data can be obtained from the proximity sensors covering each link, which includes white noise.

To consider a feasible method, we assume a situation that uses multiple VL53L5CX [15] as an example of the existing ToF-type proximity sensors that can be mounted on a link. The VL53L5CX is small (6.4 mm $\times$ 3.0 mm $\times$ 1.5 mm) and can be distributed on links. We assume that these sensors are placed along the circumference of the link. The sensor uses multiple light beams emitted from a single unit to acquire distance data from target objects. While the sensor has relatively low responsiveness compared with other proximity sensors, a single unit is capable of obtaining environmental information at a frequency comparable to those of the LiDAR systems commonly employed in SLAM applications. This configuration enables the comprehensive environmental data surrounding the entire body to be treated as point cloud data derived from proximity sensors. Notably, our method is not limited to VL53L5CX. The fundamental requirement of our method is the installation of noncontact sensors on each robot link that are capable of acquiring point cloud data. This concept is discussed in detail in the next section.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

C. Foundational Method

Point cloud-based SLAM methods commonly extract features, such as planes and edges, from sensor point cloud data to reduce the number of computations [16]. However, the method of feature extraction changes according to the method of gathering points [17]. In our assumed situation, where many sensors are installed on the link surface, the sensor model must be constructed every time the sensor arrangement changes. To avoid this, we applied the technique of FAST-LIO2 [17] to our state estimation method. One of the advantages of FAST-LIO2 is its ability to use raw point cloud data for estimation while maintaining a low computational load; it can do this because it uses a point cloud management method involving *ikd-Tree* [17] and an error-state iterated Kalman filter, which has a computational load that depends on the state dimension instead of the measurement dimension [18].

We developed a method for whole-body SLAM of articulated structures by applying FAST-LIO2 to our two key ideas: the acquisition of external environment information by proximity sensors and the information propagation in the spatial direction using the articulated link structure.

D. System Overview

The system overview is shown in Fig. 1. The proposed method is divided into two stages at each step: the estimation of the root state $\mathbf{x}_{0,k}$ and the subsequent estimation of the state $\mathbf{x}_{i,k}$ $(i > 0)$ of each link. The root estimation stage is performed through a conventional discrete-time model. By combining the prediction based on the root state $\bar{\mathbf{x}}_{0,k-1}$ obtained from the previous step with the current observations $\mathbf{p}_j$, we can perform state estimation of the root and acquire information on the environment surrounding the root.

Once the estimation of the root state is completed, for the subsequent links, we can construct a model that utilizes the constraint that all links are connected at the same time. In other words, after the root estimation, we make a prediction by recursively describing the state of each link along the link direction and then combine this prediction with observations to perform estimation. In this process, estimation proceeds from the root to the end link in the same time series, which we call the spatial direction. However, not all state variables can be propagated spatially. Some state variables, such as the biases of the joints, change in unique ways for each joint, regardless of the link constraints. Therefore, as with the root, these variables are predicted on the basis of their past states. In this process, estimation proceeds in the temporal direction. Thus, the states of the links after the root are predicted by combining the state variables propagated spatially and those propagated temporally. These predictions are then integrated with observations to estimate the state relative to the environment and acquire environmental information.

Once the estimation of all the links, i.e., the estimation of the full-body posture, is complete, the estimation for the next step begins again from the root. The point cloud data obtained from the proximity sensors are converted to the world coordinate system after the state of each link is estimated and are added to the map as information by *on-tree downsampling* [17]. This allows the amount of available map information to increase with each successive estimation of the link, even within the same step.

Integration with observations is performed via an iterated extended Kalman filter. The details of this process, including the model, are described in Sections II-E to II-G.

E. Articulated Structure Model

The root state $\mathbf{x}_{0,k}$ includes the position $\mathbf{p}_{0,k} \in \mathbb{R}^3$ and orientation $\mathbf{R}_{0,k} \in \mathrm{SO}(3)$. The state of the root at the kth step can be expressed as follows.

$$\mathbf{p}_{0,k} = \mathbf{p}_{0,k-1} + \boldsymbol{\omega}_{\mathbf{p}_k} \tag{1}$$

$$\mathbf{R}_{0,k} = \mathbf{R}_{0,k-1}\mathrm{Exp}(\boldsymbol{\omega}_{\mathbf{R}_k}) \tag{2}$$

In this work, temporal changes in position and orientation with respect to the previous step are modeled by Gaussian noise as a random walk process, where $\boldsymbol{\omega}_{\mathbf{p}_k} \in \mathbb{R}^3$ is the amount of change in position; $\boldsymbol{\omega}_{\mathbf{R}_k} \in \mathbb{R}^3$ is a vector of rotation axes. In the case of snake robots or mobile manipulators, the root often includes IMUs or wheel odometry. It is therefore possible to construct a root model that incorporates the available odometry. $\mathrm{Exp}(\mathbf{n}) \in \mathrm{SO}(3)$ is the matrix exponential expressed by the *Rodrigues rotation formula* as follows:

$$\mathrm{Exp}(\mathbf{n}) = \mathbf{I} + \sin\|\mathbf{n}\| \left[\frac{\mathbf{n}}{\|\mathbf{n}\|}\right]_\times + (1 - \cos\|\mathbf{n}\|) \left[\frac{\mathbf{n}}{\|\mathbf{n}\|}\right]_\times^2 \tag{3}$$

where $\mathbf{I}$ represents the identity matrix and where $[\cdot]_\times$ denotes an operator that transforms an $\mathbb{R}^3$ vector to a skew-symmetric $\mathbb{R}^{3\times3}$ matrix.

The state of the link following the root $\mathbf{x}_{i,k}$ $(i > 0)$ includes not only $\mathbf{p}_{i,k}$ and $\mathbf{R}_{i,k}$ but also the angular bias $b_{i,k} \in \mathbb{R}$ of the joint. The state of the ith link at the kth step can be described as follows.

$$b_{i,k} = b_{i,k-1} + \omega_{b_{i,k}} \tag{4}$$

$$\mathbf{p}_{i,k} = \mathbf{p}_{i-1,k} + \mathbf{R}_{i-1,k}\,{}^{i-1}\mathbf{p}_i \tag{5}$$

$$\mathbf{R}_{i,k} = \mathbf{R}_{i-1,k}\mathrm{Exp}\left\{\boldsymbol{\theta}_{i,k} - \left(b_{i,k-1} - \omega_{\theta_{i,k}}\right)\frac{\boldsymbol{\theta}_{i,k}}{\|\boldsymbol{\theta}_{i,k}\|}\right\} \tag{6}$$

The position $\mathbf{p}_{i,k}$ and orientation $\mathbf{R}_{i,k}$ are determined by the state of the $i-1$th link propagated in the spatial direction from the same step, while the bias $b_{i,k}$ is determined by the state of the ith estimated in the past step. The amount of change in the bias at each step is modeled by the Gaussian noise $\omega_{b_{i,k}}$ as a random walk process. The relative position vector ${}^{i-1}\mathbf{p}_i \in \mathbb{R}^3$ between the links is determined by the shape of each link. The change in posture between links is represented by a rotation axis vector $\boldsymbol{\theta}_{i,k}$ derived from the joint angles considering the bias $b_{i,k-1}$ and measurement noise $\omega_{\theta_{i,k}}$. Our method recursively represents the full-body state by considering localized interlink connections without representing the entire body of the robot in a single state space. This approach enables the estimation process to be performed via spatial propagation.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

F. Measurement Model

The sensors on each link acquire a point cloud $\{\mathbf{p}_j \mid j = 1, 2, \ldots, m\}$ once per step. The sensor model employed in this study is fundamentally similar to that described in [17]. For a detailed derivation of this sensor model, readers are directed to [17].

The acquired point cloud is converted from a link frame to a global frame according to the predicted position and orientation of each sensor and then projected onto the map. Assuming that the projected points should be in the local plane on the map, as in [17], the implicit sensor model is constructed as follows:

$$\mathbf{n}_j^\top \left[\mathbf{R}_{i,k} \left\{ \mathbf{R}_{S_i}^{L_i} \left(\mathbf{p}_j - \boldsymbol{\upsilon}_j \right) + \mathbf{p}_{S_i}^{L_i} \right\} + \mathbf{p}_{i,k} - \mathbf{q}_j \right] = 0 \tag{7}$$

where $\mathbf{n}_j$ is the normal vector of the local plane formed by the neighborhood points of the map that include $\mathbf{q}_j$ when the points $\mathbf{p}_j$ are projected onto the map; $\mathbf{p}_{S_i}^{L_i}$ and $\mathbf{R}_{L_i}^{S_i}$ are the position vector and orientation matrix of the sensor in the ith link frame, respectively; and $\boldsymbol{\upsilon}_j$ is the measurement noise.

G. Iterated Extended Kalman Filter

The state estimation is performed via an iterated extended Kalman filter using an error state model as in [18], [19]. The use of error states allows all state quantities, including attitudes, to be expressed in $\mathbb{R}^3$, which is a minimum representation [20]. The error state for each state is defined as follows:

$$\tilde{\mathbf{a}} = \mathbf{a} - \bar{\mathbf{a}} \qquad \mathbf{a} \in \mathbb{R}^3 \tag{8}$$

$$\tilde{\mathbf{A}} = \mathrm{Log}\left(\bar{\mathbf{A}}^\top \mathbf{A} \right) \qquad \mathbf{A} \in \mathrm{SO}(3) \tag{9}$$

where $\mathrm{Log}(\cdot) \in \mathbb{R}^3$ is the inverse function of (3); its normalized vector represents the rotation axis of the posture error, and its magnitude represents the rotation angle of the posture error. With the introduction of error states, each state can be represented by a single vector as follows ($i > 0$):

$$\tilde{\mathbf{x}}_{0,k} = \begin{bmatrix} \tilde{\mathbf{p}}_{0,k}^\top & \tilde{\mathbf{R}}_{0,k}^\top \end{bmatrix}^\top \tag{10}$$

$$\tilde{\mathbf{x}}_{i,k} = \begin{bmatrix} \tilde{\mathbf{p}}_{i,k}^\top & \tilde{\mathbf{R}}_{i,k}^\top & \tilde{b}_{i,k} \end{bmatrix}^\top \tag{11}$$

$$\boldsymbol{\omega}_{0,k} = \begin{bmatrix} \boldsymbol{\omega}_{\mathbf{p}_k}^\top & \boldsymbol{\omega}_{\mathbf{R}_k}^\top \end{bmatrix}^\top, \boldsymbol{\omega}_{i,k} = \begin{bmatrix} \omega_{b_{i,k}} & \omega_{\theta_{i,k}} \end{bmatrix}^\top \tag{12}$$

where $\boldsymbol{\omega}_{0,k}$ and $\boldsymbol{\omega}_{i,k}$ represent the process noise vectors.

On the basis of (1)–(6), the estimated values obtained from the model's prediction are as follows ($i > 0$):

$$\hat{\mathbf{p}}_{0,k}^0 = \bar{\mathbf{p}}_{0,k-1}, \quad \hat{\mathbf{R}}_{0,k}^0 = \bar{\mathbf{R}}_{0,k-1} \tag{13}$$

$$\hat{\mathbf{p}}_{i,k}^0 = \bar{\mathbf{p}}_{i-1,k} + \bar{\mathbf{R}}_{i-1,k}{}^{i-1}\mathbf{p}_i \tag{14}$$

$$\hat{\mathbf{R}}_{i,k}^0 = \bar{\mathbf{R}}_{i-1,k} \mathrm{Exp}\left(\boldsymbol{\theta}_{i,k} - b_{i,k-1} \frac{\boldsymbol{\theta}_{i,k}}{\|\boldsymbol{\theta}_{i,k}\|} \right) \tag{15}$$

$$\hat{b}_{i,k}^0 = \bar{b}_{i,k-1} \tag{16}$$

Using (8) and (9), the true values on the left-hand sides of (1), (2), and (4)–(6) can be expressed in terms of the error state $\tilde{\mathbf{x}}$ relative to the predicted value and the predicted value itself $\hat{\mathbf{x}}^0$. Similarly, the true values on the right-hand side can be expressed via the error state $\tilde{\mathbf{x}}$ relative to the estimated value and the estimated value itself $\bar{\mathbf{x}}$. These allow us to rewrite the articulated structure model as an equation for the transition of the error state. The derived equation can be linearized in the area where the error and noise approach zero and can be expressed as follows ($i > 0$):

$$\tilde{\mathbf{x}}_{0,k} \simeq \mathbf{F}_{\tilde{\mathbf{x}}_{0,k-1}} \tilde{\mathbf{x}}_{0,k-1} + \mathbf{F}_{\boldsymbol{\omega}_{0,k}} \boldsymbol{\omega}_{0,k} \tag{17}$$

$$\tilde{\mathbf{x}}_{i,k} \simeq \mathbf{F}_{\tilde{\mathbf{x}}_{i,k-1}} \tilde{\mathbf{x}}_{i,k-1} + \mathbf{F}_{\tilde{\mathbf{x}}_{i-1,k}} \tilde{\mathbf{x}}_{i-1,k} + \mathbf{F}_{\boldsymbol{\omega}_{i,k}} \boldsymbol{\omega}_{i,k} \tag{18}$$

where each $\mathbf{F}$ is a Jacobian for linearization and can be derived as in [18]. With (17) and (18), the uncertainty of the error state is propagated as follows ($i > 0$):

$$\hat{\mathbf{P}}_{0,k} = \mathbf{F}_{\tilde{\mathbf{x}}_{0,k-1}} \bar{\mathbf{P}}_{0,k-1} \mathbf{F}_{\tilde{\mathbf{x}}_{0,k-1}}^\top + \mathbf{F}_{\boldsymbol{\omega}_{0,k}} \mathbf{Q}_{0,k} \mathbf{F}_{\boldsymbol{\omega}_{0,k}}^\top \tag{19}$$

$$\begin{aligned} \hat{\mathbf{P}}_{i,k} = {} & \mathbf{F}_{\tilde{\mathbf{x}}_{i,k-1}} \bar{\mathbf{P}}_{i,k-1} \mathbf{F}_{\tilde{\mathbf{x}}_{i,k-1}}^\top + \mathbf{F}_{\tilde{\mathbf{x}}_{i-1,k}} \bar{\mathbf{P}}_{i-1,k} \mathbf{F}_{\tilde{\mathbf{x}}_{i-1,k}}^\top \\ & + \mathbf{F}_{\boldsymbol{\omega}_{i,k}} \mathbf{Q}_{i,k} \mathbf{F}_{\boldsymbol{\omega}_{i,k}}^\top \end{aligned} \tag{20}$$

where $\hat{\mathbf{P}}_{0,k}$ and $\hat{\mathbf{P}}_{i,k}$ are propagated covariance matrices of $\tilde{\mathbf{x}}_{0,k}$ and $\tilde{\mathbf{x}}_{i,k}$; $\bar{\mathbf{P}}_{0,k-1}$, $\bar{\mathbf{P}}_{i,k-1}$ and $\bar{\mathbf{P}}_{i-1,k}$ are covariance matrices of $\tilde{\mathbf{x}}_{0,k-1}$, $\tilde{\mathbf{x}}_{i,k-1}$ and $\tilde{\mathbf{x}}_{i-1,k}$, respectively; and $\mathbf{Q}_{0,k}$ and $\mathbf{Q}_{i,k}$ are the noise covariances of $\boldsymbol{\omega}_{0,k}$ and $\boldsymbol{\omega}_{i,k}$, which are set manually.

As with the articulated structure model, the measurement model in (7) can be rewritten using the error state and can be linearized as follows:

$$0 \simeq z_j^\kappa + \mathbf{H}_j^\kappa \tilde{\mathbf{x}}_{i,k}^\kappa + \upsilon_j \tag{21}$$

where z_j^κ represents the actual observed measurement obtained by substituting $\mathbf{p}_{i,k} = \hat{\mathbf{p}}_{i,k}^\kappa$, $\mathbf{R}_{i,k} = \hat{\mathbf{R}}_{i,k}^\kappa$ and $\mathbf{v}_j = \mathbf{0}$ into (7) and serves as the basis for linearization. $\mathbf{H}_j^\kappa$ is a Jacobian for linearization, corresponding to $\tilde{\mathbf{x}}_{i,k}^\kappa$. $\upsilon_j = -\mathbf{n}_j \hat{\mathbf{R}}_{i,k}^\kappa \mathbf{R}_{S_i}^{L_i} \boldsymbol{\upsilon}_{i,k}^j$ includes measurement noise whose covariance varies with each estimation, but it is shown in [17] that setting this variable to a constant value works well. As κ is included in (21), the observation model is computed for each iteration.

The iterated Kalman filter estimates the increment $\tilde{\mathbf{x}}^\kappa$ with respect to the current error state vector $\tilde{\mathbf{x}}_{i,k}^{0,\kappa}$ to minimize the following weight square sum:

$$\min_{\tilde{\mathbf{x}}^\kappa} \left(\left\| \tilde{\mathbf{x}}_{i,k}^{0,\kappa} + \mathbf{J}_{i,k}^\kappa \tilde{\mathbf{x}}_{i,k}^\kappa \right\| + \sum_{j=1}^{m} \left\| z_j^\kappa + \mathbf{H}_j^\kappa \tilde{\mathbf{x}}_{i,k}^\kappa \right\| \right) \tag{22}$$

where $\mathbf{J}_{i,k}^\kappa$ is a square matrix that eliminates the nonlinearity associated with the computation of the attitude error vector [17], [18] and where m is the number of point measurements.

With (22), we can estimate $\tilde{\mathbf{x}}_{i,k}^\kappa$ as follows [17], [18]:

$$\mathbf{z}^\kappa = \begin{bmatrix} z_1^\kappa & \ldots & z_m^\kappa \end{bmatrix}^\top, \quad \mathbf{H}^\kappa = \begin{bmatrix} \mathbf{H}_1^{\kappa\top} & \ldots & \mathbf{H}_m^{\kappa\top} \end{bmatrix}^\top \tag{23}$$

$$\mathbf{K}_{i,k}^\kappa = \left(\mathbf{H}_\kappa^\top \mathbf{R}^{-1} \mathbf{H}_\kappa + \left(\mathbf{J}_{i,k}^\kappa \right)^\top \hat{\mathbf{P}}_{i,k}^{-1} \mathbf{J}_{i,k}^\kappa \right)^{-1} \mathbf{H}_\kappa^\top \mathbf{R}^{-1} \tag{24}$$

$$\tilde{\mathbf{x}}_{i,k}^\kappa = \mathbf{K}_{i,k}^\kappa \left(-\mathbf{z}_{i,k}^\kappa + \mathbf{H}^\kappa \left(\mathbf{J}_{i,k}^\kappa \right)^{-1} \tilde{\mathbf{x}}_{i,k}^{0,\kappa} \right) - \left(\mathbf{J}_{i,k}^\kappa \right)^{-1} \tilde{\mathbf{x}}_{i,k}^{0,\kappa} \tag{25}$$

where $\mathbf{R}$ represents the diagonal covariance matrix of υ_1 to υ_j. In (24), we obtain a variant of the formula for the general

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Algorithm 1: Estimation Process.

for k **do**
  for $0 \leq i \leq$ number of links **do**
    if $i = 0$ **then**
      Input: $\bar{\mathbf{x}}_{0,k-1}$, $\bar{\mathbf{P}}_{0,k-1}$, $\mathbf{p}_j$;
      Calculate $\hat{\mathbf{P}}_{0,k}$ by (19);
    end
    else
      Input: $\bar{\mathbf{x}}_{i,k-1}$, $\bar{\mathbf{P}}_{i,k-1}$, $\bar{\mathbf{x}}_{i-1,k}$, $\bar{\mathbf{P}}_{i-1,k}$;
      Input: $\mathbf{p}_j$, and $\theta_{i,k}$;
      Calculate $\hat{\mathbf{x}}^0_{i,k}$, $\hat{\mathbf{P}}_{i,k}$ by (20);
    end
    $\kappa \leftarrow 0$;
    repeat
      Compute $\mathbf{H}^\kappa$, $\mathbf{z}^\kappa$, $\mathbf{J}^\kappa$, and $\tilde{\mathbf{x}}^{0,\kappa}$;
      Compute $\tilde{\mathbf{x}}^\kappa$ and $\hat{\mathbf{x}}^{\kappa+1}$ via (24)-(28);
      $\kappa \leftarrow \kappa + 1$;
    until $\tilde{\mathbf{x}}^\kappa_{i,k}$ converges;
    Output: $\bar{\mathbf{x}}_{i,k}$, $\bar{\mathbf{P}}_{i,k}$ via (29), (30);
    Add the measured points to the map according to $\bar{\mathbf{x}}_{i,k}$ via (31);
  end
end

Kalman gain $\mathbf{K}^\kappa_{i,k}$ by using an inverse matrix lemma, allowing the calculation to be performed in the state dimension rather than the measurement dimension [17]. From the estimate $\tilde{\mathbf{x}}^\kappa_{i,k}$, each state is updated as follows.

$$\hat{\mathbf{p}}^{\kappa+1}_{i,k} = \hat{\mathbf{p}}^\kappa_{i,k} + \tilde{\mathbf{p}}^\kappa_{i,k} \tag{26}$$

$$\hat{\mathbf{R}}^{\kappa+1}_{i,k} = \hat{\mathbf{R}}^\kappa_{i,k}\mathrm{Exp}\left(\tilde{\mathbf{R}}^\kappa_{i,k}\right) \tag{27}$$

$$\hat{b}^{\kappa+1}_{i,k} = \hat{b}^\kappa_{i,k} + \tilde{b}^\kappa_{i,k} \tag{28}$$

Using $\hat{\mathbf{x}}^{\kappa+1}$, (21)–(28) are repeated until $\tilde{\mathbf{x}}^\kappa_{i,k}$ falls below the threshold and converges. After convergence, the final estimated state $\bar{\mathbf{x}}_{i,k}$ and the covariance matrix of its error state $\bar{\mathbf{P}}_{i,k}$ are determined as follows:

$$\bar{\mathbf{x}}_{i,k} = \hat{\mathbf{x}}^{\kappa+1}_{i,k} \tag{29}$$

$$\bar{\mathbf{P}}_{i,k} = \left(\mathbf{I} - \mathbf{K}^\kappa\mathbf{H}^\kappa\right)\left(\mathbf{J}^\kappa\right)^{-1}\hat{\mathbf{P}}_{i,k}\left(\mathbf{J}^\kappa\right)^{-\top} \tag{30}$$

The estimated state and covariance are propagated in the spatial and temporal directions and are used for each estimation. The points acquired by the proximity sensor $\mathbf{p}^j_{i,k}$ are transformed into the world frame ${}^G\mathbf{p}^j_{i,k}$ on the basis of the estimated state of the link and added to the map by *on-tree downsampling* [17] after each link estimation step.

$${}^G\mathbf{p}^j_{i,k} = \bar{\mathbf{R}}_{i,k}\left(\mathbf{R}^{L_i}_{S_i}\mathbf{p}^j_{i,k} + \mathbf{p}^{L_i}_{S_i}\right) + \bar{\mathbf{p}}_{i,k} \tag{31}$$

In summary, the estimation process of this method is shown in Algorithm 1.

III. Simulation

To verify the effectiveness of the proposed method, we conducted simulations in Gazebo under multiple environments. The multijoint structure model in the simulation consists of links connected in series by one-DOF joints. In Fig. 1, we define the x-axis along the length of the links, the y-axis in the depth direction, and the z-axis in the vertical direction. For this simulation, we assume a model in which odd-numbered joints rotate around the z-axis and even-numbered joints rotate around the y-axis. Eight proximity sensors, modeled after the VL53L5CX, are positioned circumferentially at the midpoint of each link's length with a radius of 5 cm. Each sensor acquires the data of 64 points, resulting in a total of 512 points per link. The detection range of each sensor is 5 cm to 4 m. Furthermore, we consider two sources of uncertainty: a bias of 0.05 rad in the angle measurement for each joint and white noise in the distance measurements obtained from proximity sensors. The covariance σ of the white noise is set to $2.7\sigma = \alpha r$, where r is the measured distance and α is the noise level. In each simulation, arbitrary sine waves are given as position command values to each joint of the simulation model to generate motion. We then compared the state of the articulated structure relative to the environment, which is obtained via the proposed method, with that derived solely from the kinematic model to verify the effectiveness of the proposed method.

As described in Section II-D, the proposed method is divided into the steps of estimating the root and estimating each link following the root. Since the method of root estimation is the same process as in the general state estimation method, it is not the primary focus of this paper. The most crucial point of our proposed method is the estimation of each link following the root, which combines state variables that propagate spatially and temporally. Therefore, in Section III-A, we first present the results of simulations in which the root is fixed. Additionally, in Section III-B, to verify the adaptability of the proposed method to situations where the root position is ambiguous or the root is in motion, we conduct simulations with an unconstrained root.

Simulations are conducted on a computer equipped with a 16-core Intel i7-13700 CPU and 32 GB of RAM.

A. Effects of Spatial Direction Estimation

Fig. 2(a-1) to (a-3) present the results of the simulations with the 5-link structure, whereas (b-1) to (b-3) present those for the 20-link structure in $\alpha = 1\%$. The 5-link structure is composed of links with 37 cm length, which assumed to represent a robot arm, while the 20-link structure consists of links with 17 cm length, which assumed to represent a robot with many small-scale links, such as a snake robot. (a-2) and (a-3), as well as (b-2) and (b-3), are captured with the same viewing angle relative to the root. While the use of the kinematic model alone results in significantly distorted pose estimates due to biases, leading to misaligned environmental sensor data and an imprecise map, our proposed method successfully compensates for these biases and correctly maps the environment.

In addition, Table II presents the results of simulations performed by individually varying several parameters based on the

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

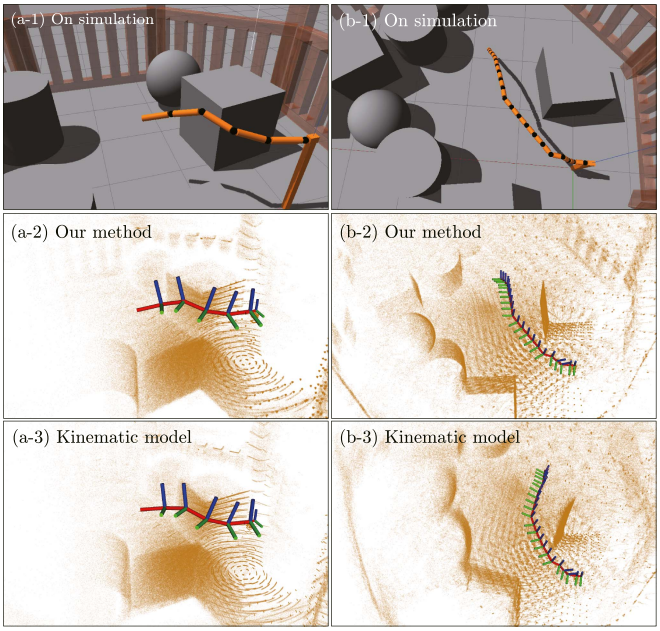

Fig. 2. Snapshot of the simulation. (a-1), (b-1) show the actual structure in the simulation environment. (a-2), (b-2) show the states and acquisition environment obtained via the proposed method. (a-3), (b-3) show the states and acquisition environment obtained via the kinematic model. The position and orientation of the robot are represented in coordinate systems. The origin of each system is located at the center of the adjacent joint on the root side of each link. In these coordinate systems, the x-axis is denoted by red, the y-axis by green, and the z-axis by blue.

TABLE II
MEAN AND STANDART DEVIATION OF THE ABUSOLUTE POSITION ERROR IN 100 SIMULATIONS

		$\mathbf{R}$	Link 5 (cm)	Link 10 (cm)	Link 20 (cm)
Link number	20-link	0.01	5.0±0.8	4.3±0.9	5.3±1.6
	10-link	0.01	5.9±1.5	6.9±2.5	-
	5-link	0.01	4.0±3.2	-	-
Point number	256	0.01	4.0±1.3	4.9±1.9	8.6±4.0
	128	0.01	3.1±1.3	4.5±1.6	9.8±5.1
Noise level	2%	0.02	3.4±0.9	4.5±1.4	7.8±3.2
	5%	0.05	5.2±2.2	6.7±4.6	11.5±11.0
	7%	0.07	6.5±2.6	8.4±3.7	20.5±13.2
Bias level (°)	2	0.02	4.3±1.3	4.3±1.7	6.4±3.2
	3	0.05	4.4±1.1	4.2±1.2	4.0±2.1
	5	0.07	3.9±1.1	4.3±1.7	7.6±7.5
Kinematic model		20-link	5.4±0.0	22.8±0.2	90.0±5.6
		10-link	5.9±1.5	33.9±1.7	-
		5-link	12.5±0.5	-	-

simulation conditions of the 20-link structure illustrated in Fig. 2(b-1) to (b-3). Specifically, we varied the number of links, the number of acquired points (sensor count), the noise levels α, and the sensor position biases. Respect to the number of links, the link length was set to 37 cm for the 5-link, 27 cm for the 10-link configuration. The variation in the number of acquired points was implemented by reducing the spatial resolution of the sensors. The sensor position biases represent sensor placement uncertainties, specifically considering scenarios where each sensor were misaligned by an angle with respect to an arbitrary axis. The rotation axis was randomized for each simulation and each sensor, with deviation angle randomly selected within a predefined maximum range. We evaluated the induced performance variations on the basis of this maximum deviation angle, which we call the "bias level". The one-minute simulations were conducted 100 times under each condition. In Table II, the mean absolute errors and standard deviations between the estimated and true values are represented. The values for the fifth link, tenth link, and twentieth link from the root are shown as representative in the table.

Regarding the link count, our method demonstrated consistent accuracy across structures with various numbers of links. When the spatial resolution of the sensor was reduced from 512 points to 128 points per link (a quarter of the original quantity), the mean error increased, but no significant performance degradation was observed. Increasing the sensor position bias occasionally led to estimation failures due to point cloud misalignment. However, by enlarging the observation covariance matrix $\mathbf{R}$ of the Kalman filter to tolerate a certain degree of sensor misalignment, a successful estimation effect was achieved. As the noise level of the sensor α increased, both the error and variance grew due to the impacts of sensor noise, with these errors accumulating toward the endpoint. In the one-minute simulations, the mean absolute error at the end link reached 20 cm. On the other hand, in a kinematic model including biases, the mean position error at the endpoint was approximately 90 cm for the 20-link configuration. Even under the most challenging noise condition ($\alpha = 7\%$), our method was able to compensate for the cumulative error induced by the kinematic-only approach, reducing the endpoint error by 77%.

Through simulations with a fixed root, the results demonstrate the effectiveness of spatial propagation in estimation, which is the most crucial aspect of our proposed method. Furthermore, by conducting simulations under various conditions with a fixed root, we validated the difference of the peformance of our method across different link lengths, acquired point counts, and sources of uncertainty. A detailed discussion is provided in Section IV-A.

B. Robustness to Root Uncertainty

As shown in Fig. 3, we placed the 20-link articulated structure on the ground, similar to the simulation in Section III-A, but did not fix it in place. The motion of the structure is generated from the sine wave applied to each joint, and no control is implemented. The biases in each joint were maintained at the same levels as those in the fixed-root simulation in Section III-A, and the noise level α of each sensor was set to 1%. In this study, the IMU is not used to estimate the root, and the model prediction equation is shown in (13). Hence, it is impossible to understand the movement of the root in relation to the environment without environmental information, and a comparison with the results obtained using the kinematic model alone is not given here. Fig. 4 depicts the trajectory of the root's position and orientation, along with the corresponding ground-truth trajectory.

In the simulations, we applied the same trajectory 100 times to the root link. We represent the mean of the estimated root values as solid lines, the ground-truth values as dashed lines, and the standard deviations as shaded bands in Fig. 4. Additionally,

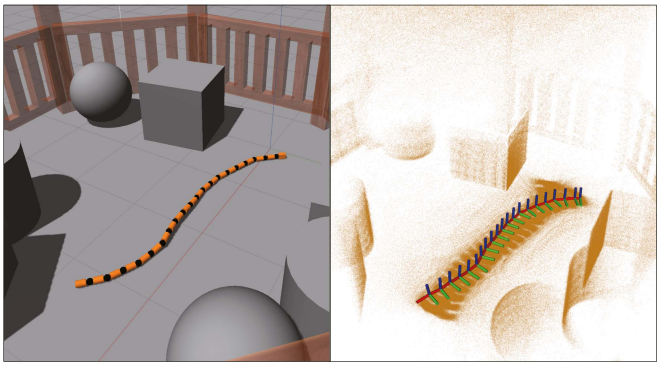

Fig. 3. Simulation with an unconstrained root. The left image shows the actual state of the articulated structure in the simulation, whereas the right image displays the estimated state of the structure and the acquired surrounding environment.

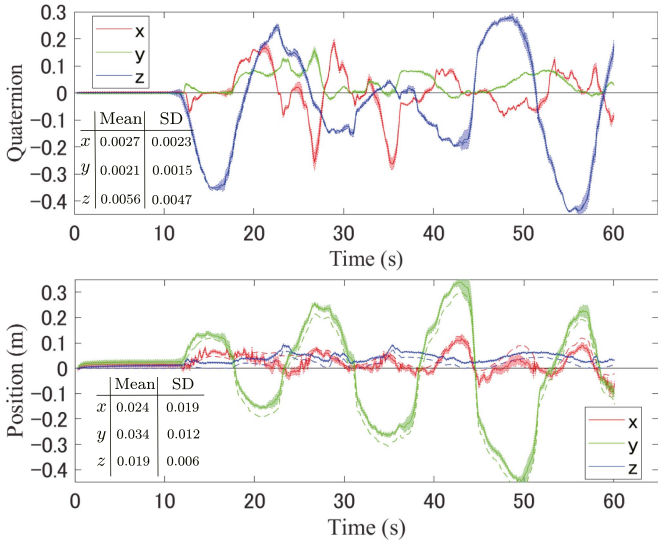

Fig. 4. Plot of the root link states acquired via the proposed method and the ground truth. The upper plot shows the x, y, and z components of the quaternion, and the lower plot displays the position. We represent the mean of each root's estimated value as solid lines, each ground truth values as dashed lines, and each standard deviations as shaded bands. The means and standard deviations of absolute errors between estimated and true values for each element are presented in the bottom-left corner of each plot.

the means and standard deviations of absolute errors between estimated and true values for each element are presented in the bottom-left corner of each plot. From 10 s onward, the position and orientation of the root undergo displacement due to the whole-body motion of the structure. Despite this movement, the mean norm of the position error vector was within 0.046 m. Each quaternion element closely tracked the true value with a minimal error. Utilizing external information derived from whole-body proximity sensors, our proposed method successfully estimated the state of the root.

However, the estimation is unsuccessful in cases where the root undergoes rapid movements, such as when the entire structure collapses in a rolling motion. This failure can be attributed to the relatively low temporal frequency of sensor data acquisition compared with the speed of movement, resulting in temporally sparse information.

In simulations with unfixed roots, we were able to demonstrate that the proposed method can be effectively applied even when the entire structure moves relative to the environment and the state of the root is unclear, provided that the movement of the root is slow.

IV. Discussion

A. Advantages of Estimation in the Spatial Direction

Distributing proximity sensors over an entire structure not only enables the acquisition of environmental information surrounding the entire structure but also allows the individual estimation of each link by providing unique external information to each joint. This estimation process enables the spatial propagation of the estimated state quantities. In the simulation of the 20-link articulated structure described in Section III-A, we considered cases where small errors accumulate toward the final link, potentially resulting in significant endpoint deviations. The ability of the proposed method to accurately estimate the states in such scenarios can be attributed to its approach of propagating the estimation in the spatial direction and correcting errors from the root. Furthermore, as the estimation of each link is followed by the sequential incorporation of the acquired environmental information into the map, the increased availability of environmental data to the end-effector links contributes significantly to improving the estimation accuracy of these terminal links.

The benefits of propagating this spatial information are due to the fact that all links are estimated simultaneously. Therefore, when designing the hardware system, the ideal design would be one in which all the proximity sensors across the body can acquire data as close to simultaneously as possible. Furthermore, as a future development of the proposed method, we are considering the possibility of developing a technique that can accommodate temporal misalignment of sensor data acquisition in cases where simultaneous data collection is difficult.

In our approach, the estimated state variables are always associated with a single link. Consequently, the state dimensionality of the Kalman filter does not change with the number of links, and the computational complexity of the Kalman filter itself remains constant even as links are added. However, since our method recursively estimates the entire body, the number of estimation steps increases with each additional link and the time required for computation will scale with the increase in the link count. In the simulations, the time required to estimate the whole body was 17 ms for the 5-link, 22 ms for the 10-link, and 32 ms for the 20-link on average, with a CPU utilization of 5% and a memory utilization of 0.6%.

B. Contribution to the Estimation of Roots

In Section III-B, we verified that when the movement per unit time is small, it is possible to estimate the root's motion using only external environmental information, even if the structure is not fixed relative to the environment. The map utilized for estimation in the proposed method incorporates environmental

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

information acquired by proximity sensors on links other than the one being estimated. This comprehensive approach enables the estimation process to leverage all available environmental data. The utilization of a map covering a broader area than that captured by the root's own sensors is likely a contributing factor to successful estimation.

C. Application

Our proposed method can adapt to more various situations by obtaining higher-frequency information about the root's state. In Section III-B, the estimation was performed under the assumption that the temporal changes in the root were unknown, which made it difficult to estimate rapid movements. However, if we can obtain the state between sampling periods of proximity sensors via devices such as IMUs or wheel odometry, we can improve the accuracy of the estimation. By incorporating information from these sensors into the model equations (1) and (2), we can expect to achieve whole-body state estimation relative to the environment for articulated mobile robots involving large displacements and rotations.

In the simulations, we considered a basic model in which joint angles were directly obtained from encoders. However, since our model represents relative poses via $\mathrm{SO}(3)$, we believe that this approach can be applied to robots with diverse degrees of freedom, such as soft robots, as long as their structures can be spatially discretized and the relative relationships between different links can be described by a geometric model. In future work, we aim to implement this method on various articulated structures or soft robots to verify its effectiveness.

V. Conclusion

We propose a method for whole-body state estimation and environmental information acquisition for articulated structures that are challenging to assess via conventional kinematic models due to link deformations or that require comprehensive external environmental information. This method involves deploying proximity sensors throughout the body. The proximity sensors distributed across the entire structure not only enable the acquisition of environmental information surrounding the whole structure but also facilitate individual joint estimation, allowing estimation to progress spatially. This spatial progression of estimation enables the correction of cumulative biases along the length of the structure from the root, thus allowing for accurate whole-body state estimation even when the state significantly deviates from the kinematic model. Our method was validated through simulations, which demonstrated its ability to correct individual joint biases in a model with inherent biases and to achieve accurate posture estimation relative to the actual environment. The proposed method is intended to be applied to various articulated structures and soft robots, and we would like to implement it on nonrigid robots or articulated mobile robots to verify its effectiveness in future work.

References

[1] P. Wu, Y. Shentu, Z. Yi, X. Lin, and P. Abbeel, "GELLO: A general, low-cost, and intuitive teleoperation framework for robot manipulators," in *Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, 2024, pp. 12156–12163. [Online]. Available: https://openreview.net/forum?id=FO6tePGRZj

[2] Z. Fu, T. Z. Zhao, and C. Finn, "Mobile ALOHA: Learning bimanual mobile manipulation using low-cost whole-body teleoperation," in *Proc. 8th Annu. Conf. Robot Learn.*, 2024.

[3] H. Fang et al., "AirExo: Low-cost exoskeletons for learning whole-arm manipulation in the wild," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2024, pp. 15031–15038.

[4] "TsukuArmRobotics," (n.d.). [Online]. Available: https://tsukarm.co.jp/

[5] J. De Smet, G. Borghesan, and E. Vander Poorten, "Accurate pose estimation for comanipulation robotic surgery," in *Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, 2022, pp. 8064–8071.

[6] L. Meyer, K. H. Strobl, and R. Triebel, "The probabilistic robot kinematics model and its application to sensor fusion," in *Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, 2022, pp. 3263–3270.

[7] C. Garcia Cifuentes, J. Issac, M. Wüthrich, S. Schaal, and J. Bohg, "Probabilistic articulated real-time tracking for robot manipulation," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 577–584, Apr. 2017.

[8] M. Klingensmith, S. S. Sirinivasa, and M. Kaess, "Articulated robot motion for simultaneous localization and mapping (ARM-SLAM)," *IEEE Robot. Autom. Lett.*, vol. 1, no. 2, pp. 1156–1163, Jul. 2016.

[9] C. Sorensen, P. Hyatt, M. Ricks, S. Nielsen, and M. D. Killpack, "Soft robot configuration estimation and control using simultaneous localization and mapping," in *Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, 2021, pp. 616–623.

[10] S. E. Navarro et al., "Proximity perception in human-centered robotics: A survey on sensing systems and applications," *IEEE Trans. Robot.*, vol. 38, no. 3, pp. 1599–1620, Jun. 2022.

[11] H. Arita, "A fast optical proximity sensor skin that contains an analog computing circuit and can cover an entire link," *Adv. Robot.*, vol. 37, no. 17, pp. 1083–1099, 2023.

[12] S. J. Moon, J. Kim, H. Yim, Y. Kim, and H. R. Choi, "Real-time obstacle avoidance using dual-type proximity sensor for safe human-robot interaction," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 8021–8028, Oct. 2021.

[13] Y. Ding, F. Wilhelm, L. Faulhammer, and U. Thomas, "With proximity servoing towards safe human-robot-interaction," in *Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, 2019, pp. 4907–4912.

[14] S. Suyama, M. Nakajima, H. Arita, and M. Tanaka, "Control of a snake robot with proximity sensors to adapt for two variable planes," *IEEE Access*, vol. 12, pp. 46864–46880, 2024.

[15] STMicroelectronics, "Time-of-Flight (ToF) 8x8 multizone ranging sensor with wide field of view," (n.d.). [Online]. Available: https://www.st.com/en/imaging-and-photonics-solutions/vl53l5cx.html

[16] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robot. Sci. Syst.*, 2014, no. 9, pp. 1–9.

[17] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct liDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.

[18] W. Xu and F. Zhang, "FAST-LIO: A fast, robust liDAR-inertial odometry package by tightly-coupled iterated Kalman filter," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 3317–3324, Apr. 2021.

[19] J. Huai and X. Gao, "A quick guide for the iterated extended Kalman filter on manifolds," 2023, *arXiv:2307.09237*.

[20] V. Madyastha, V. Ravindra, S. Mallikarjunan, and A. Goyal, "Extended Kalman filter vs. error state Kalman filter for aircraft attitude estimation," in *Proc. AIAA Guid. Navigat. Control Conf.*, 2011, Art. no. 6615.