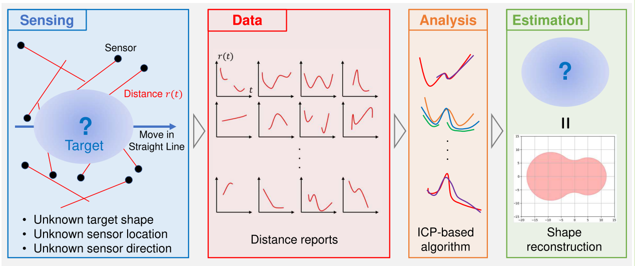
Shape Estimation Using Location-Unknown Distance Sensors: Iterative-Closest-Point-Based Approach

Hiroki Ikeuchi and Hiroshi Saito, *Fellow, IEEE*



***Abstract*—We discuss the shape estimation of moving target objects using distributed ranging sensors. Due to the inability to carefully design sensor locations or assign global positioning systems (GPS) to inexpensive sensors, it is often necessary to assume that the sensor locations and target object locations are unknown. Although methods have been developed that can be applied in such situations, the sensing results are assumed to include no noise and cannot be applied to practical situations in which sensing noise exists. We propose a method of estimating the whole shape of a moving target object in the presence of sensing noise. Our method analyzes continuous reports on the measured distance of a target object from distributed sensors and determines the sensing directions of sensors using a novel algorithm, which we developed inspired by the iterative closest point (ICP) algorithm. On the basis of the obtained sensing directions, the whole shape of the object can be estimated. We conducted extensive numerical simulations and an experiment using actual laser sensors to demonstrate the effectiveness and feasibility of our method.**

***Index Terms*—Distance sensor, distributed sensor, iterative closest point, random placement, shape estimation, sensor network, unknown location.**

I. Introduction

It is essential to collect information from a large number of distributed sensors and draw new insights from them. With the development of wireless communication technologies, such as low-power wide-area networks (LPWANs) [1]–[5], wide-area and long-distance communication is possible with low power consumption and low cost. In such a communication environment, sensors and Internet-of-Things (IoT) devices deployed like the scattering of dust [6] continuously transmit acquired data, which enables the controller to conduct integrated data analysis and extract useful information. Each sensor has only a simple function and does not necessarily have high performance [7]–[10]. However, by integrating sensors' reports, it is possible to extract the global properties of a target as collective knowledge, which cannot be obtained from local information alone. This type of sensing paradigm is in line with the smart city concept that has been expanding and expected to further develop.

We focus on the shape estimation of a moving target object using distributed sensors. Shape estimation by using sensors is one of the fundamental problems in wireless sensor networks and is important as a practical approach in constrained environments. For example, suppose we want to detect a particular type of vehicle entering a certain surveillance area on the basis of its shape. From the viewpoint of personal-information protection, it is sometimes necessary to avoid installing cameras that would enable the identification of human faces. In such cases, it is effective to place sensors that acquire only limited information rather than images, and use the shape estimated from such information to identify intruders. The availability of sensor-location information can also be an important constraint. With current methods, it is often assumed that sensor-location information is known or, if not known, can be identified by additional measurement mechanisms. In practice, however, this incurs very high labor costs to precisely design the locations of numerous sensors one by one and accurately place them in the designed locations. Even if it is possible to design the sensor location on a map, it may not be possible to place the sensor there due to various rea-

Manuscript received March 18, 2022; accepted April 13, 2022. Date of publication April 18, 2022; date of current version May 31, 2022.The associate editor coordinating the review of this article and approving it for publication was Prof. Chao TAN. *(Corresponding author: Hiroki Ikeuchi.)*

Hiroki Ikeuchi is with NTT Network Service Systems Laboratories, NTT Corporation, Tokyo 180-8585, Japan (e-mail: hiroki.ikeuchi.re@hco.ntt.co.jp).

Hiroshi Saito is with the Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-8656, Japan (e-mail: saito@g.ecc.u-tokyo.ac.jp).

Digital Object Identifier 10.1109/JSEN.2022.3168002

sons, such as geographical constraints or structural problems. In places where access is restricted, such as battlefields, it is difficult to implement additional measurement mechanisms and it may only be possible to scatter sensors from the air. For sensors with only simple measurement functions, global positioning systems (GPS) may not be able to be installed due to power-consumption limitations. In addition, if participatory sensing is considered, it may not be possible to obtain location information from the viewpoint of location privacy. Therefore, the following fundamental question arises: is it possible to estimate the shape of an object when sensor-location information is not available?

Our goal is to clarify the theoretical possibilities and limitations of shape estimation in situations where only limited information is available. Specifically, we consider a situation in which sensors are randomly placed and their positions are unknown. We also assume that the sensors have only a distance-measurement function. In reality, there may be cases in which more information can be accessed, but our problem formulation takes into account the worst-case conditions among various realistic situations. Therefore, the basic findings of estimability that we discuss will be useful not only for theoretical interests but also for thinking about more realistic problems.

In our previous studies, we showed from the theoretical perspective that shape estimation is possible for a wide class of objects, such as polygons and simple closed curves, even in an environment where position information is not available [11], [12]. With the methods proposed in these studies, an ideal situation is assumed in which the sensing data are not affected by noise. However, in reality, sensing data will always contain noise, and it is essential to present a formulation and estimation method that takes into account the existence of noise.

Taking the above problem into account, this research aims to reveal the possibility of shape estimation in the presence of noise in sensing data, in other words, a method that enables shape estimation when some or all of the following conditions are satisfied:

- Installation of a camera is not allowed from the viewpoint of personal information protection.
- For location privacy or power consumption, GPS cannot be installed on the sensor, and the location is unknown.
- Each sensor cannot handle large amount of data such as images due to the communication-bandwidth limitation and has only a ranging function.
- The measured value of the sensor contains random noise.

We propose a shape-estimation method that is robust against the sensing noise of distributed distance sensors. Similarly to our previous studies, sensor location is unknown and we do not use any side information or additional measurement mechanisms. With our previous method, we calculate the curvature at a single point of the target shape, which is quite sensitive to sensing noise. The proposed method avoids this process and takes an approach of, roughly speaking, searching a partial shape consistent with multiple sensing results by focusing on the outline of the shape instead of the curvature. Our method is based on an algorithm we developed, which is a variant of the iterative closest point (ICP) algorithm [13]–[15] commonly used for object recognition, to estimate the sensing direction of each sensor. To evaluate the effectiveness of the proposed method, which we call the ICP-based method, we conducted extensive numerical simulations to determine the impact of various parameters on the estimates as well as robustness against several types of noise. We also conducted an experiment using actual laser sensors, demonstrating that the ICP-based method works well in practical situations. The validation results of this method indicate that object-shape estimation is theoretically and practically feasible even when the sensor-location information is unknown and there is realistic noise in the sensing data.

The rest of this paper is organized as follows. In Section II, we introduce related work on shape estimation using sensors. In Section III, we define the mathematical model for discussing shape estimation. In Section IV, we briefly describe our previous method and issues with it, the technical details of which are presented in the Appendix, and then explain the ICP-based method. We present numerical examples of simulations in Section V and an experiment using actual laser sensors in Section VI. We conclude this paper in Section VII.

II. Related Work

A. Sensor Localization and Location-Aware Shape Estimation

Shape estimation is one of the most fundamental problems in sensor networks and has been extensively investigated. To estimate the shape of a target object using sensing data from distributed sensors, most research first attempted to determine the sensor location because it was believed that "the information gathered by such sensor nodes, in general, will be useless without determining the locations of these nodes" [16] or "the measurement data are meaningless without knowing the location from where the data are obtained" [17]. Thus, to solve the sensor localization problem [18], [19], they assumed having additional mechanisms or side information, such as locations of anchor sensors, angle-of-arrival measurements, training data and period, and distance-related measurements [20]–[26]

Since it is no longer difficult to estimate the shape of an object under the condition that the sensor-location information can be obtained or controlled, many studies focused on the efficiency and optimality of shape estimation. Wu *et al.* [27] proposed a method for capturing the whole shape of an object by combining many photos in which only a part of the object is visible. They aimed at efficient shape estimation by selecting photos on the basis of metadata including the position and orientation of the camera from the viewpoint of information-transmission constraints. Abraham *et al.* [28] proposed a method of controlling the dynamic trajectory of binary sensors to efficiently estimate the overall shape of multiple objects. The method is based on the theory of ergodic search and actively searches for the region that is most likely to return useful measurements.

Such approaches that rely on sensor localization and location-aware methods are effective because they are expected to provide highly accurate shape estimation. However, shape estimation becomes impossible in situations where additional measurement mechanisms or location information are not available, or where location privacy of the sensor is required.

B. Shape Estimation Without Location Information

We have studied the theoretical possibility of estimating the shape of an object while preserving the location privacy of sensors and and developed specific estimation methods. These studies can be roughly classified into two categories: those using binary sensors, which have a disk-shaped sensing area and return a binary value of 1/0 when detected or not detected, and those using distance sensors, which measure the shortest distance to an object or the distance in a specific direction. For binary sensors, a small number of parameters, such as the area and perimeter of the object, can be estimated [29]–[31], and several additional parameters can be estimated using a composite sensor consisting of multiple binary sensors lined up on a grid [32], [33]. However, it was also found that binary sensors alone could not obtain the entire shape of the object. On the other hand, when using distance sensors, it is possible to estimate the shape of polygonal objects [11], [34] and objects with smooth closed curves [12]. These studies showed that it is possible to achieve fairly accurate shape estimation using only distance sensors while preserving location privacy. However, since the method in [12] relies on curvature calculation, it can only be applied when the sensing noise is negligible. In the present study, we argue that shape estimation is possible even in the presence of realistic sensing noise while preserving location privacy.

III. Model

In this section, we formulate the model to be considered in shape estimation. Most of the model and the problem settings are the same as in our previous study [12], but with the present formulation, noise is assumed present in the sensing data, as described below. A single target object T enters and exits a monitored area $\mathbf{\Omega} \subset \mathbb{R}^2$, and its shape is time-invariant. It moves at a constant known speed $v > 0$ along an unknown reference directional line. In the remainder of this paper, we use this directional line as the x-axis and its direction as the reference direction. (We do not need to know the reference direction. This is used only to define direction.) The term $T(t) \subset \mathbb{R}^2$ denotes the set occupied by T at time t. The T is also defined as a region enclosed by a simple closed curve ∂T. We do not know T's shape, size, or location. We call a point the y coordinate of which is the minimum (maximum) value in T, the bottom (top) of T. Each point on ∂T is represented by an arc-length parameter s. The s of the bottom is set to $s = 0$ and increases along ∂T in the counter-clockwise direction. Let $\xi(s) \in [0, 2\pi)$ be the angle formed by the reference direction and tangent vector at s. We call the region below (above) a given line that is parallel to the x-axis and crosses T, the 'lower half-plane L' ('upper half-plane U'). The $\xi(s)$ is only used in the Appendix. The above definitions are illustrated in Figure 1.

There are n_s directional distance sensors deployed in $\mathbf{\Omega}$. Each sensor can continuously measure the distance to an object

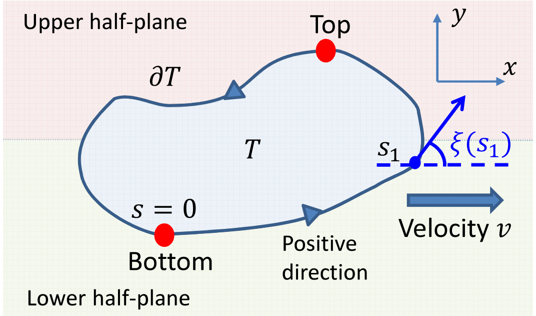


Fig. 1. Illustration of model.

lying in the sensing direction within the maximum range $r_{max} > 0$. For the i-th sensor ($1 \leq i \leq n_s$), let $\mathbf{x}_i \in \mathbf{\Omega}$ be its location, θ_i be its sensing direction, and $r_i(t)$ be the measured distance to T at t. Assume that we do not know $\mathbf{x}_i$ or θ_i for any i. We do not impose any assumptions on their distributions. We often remove the subscript and use $\mathbf{x}$, θ, $r(t)$, and ξ to simplify notation. When the location of a sensor is $\mathbf{x} = (x, y)^\mathsf{T}$ and its sensing direction from the reference direction is θ, the sensing area $S(\mathbf{x}, \theta)$ is $\{(x + s\cos\theta, y + s\sin\theta)^\mathsf{T} \mid 0 \leq \forall s \leq r_{max}\}$, where the superscript T denotes the transpose of the vector. The exact distance $\bar{r}(t)$ between the sensor and object at t is given as follows.

$$\bar{r}(t) = \begin{cases} \min_{(x+s\cos\theta,\, y+s\sin\theta)^\mathsf{T} \in T(t)} s, & \text{for} S(\mathbf{x}, \theta) \cap T(t) \neq \emptyset, \\ \emptyset, & \text{for} S(\mathbf{x}, \theta) \cap T(t) = \emptyset. \end{cases} \quad (1)$$

Specifically, $\bar{r}(t) = 0$ if $(x, y)^\mathsf{T} \in T(t)$. Here, $\emptyset$ denotes the empty set. Ideally, we expect $r(t)$ and $\bar{r}(t)$ to be equal, but in actual sensors, the measured value contains noise $\delta r(t)$. Therefore, it can be modeled as $r(t) = \bar{r}(t) + \delta r(t)$. The specific form of $\delta r(t)$ is not specified here, but it is assumed to be a noise caused by randomness such as white noise rather than systematic error. In what follows, we only use $\{r_i(t)\}_{i=1}^{N}$ ($N \leq n_s$) that satisfies $\forall t(r_i(t) > 0$ or $r_i(t) \neq \emptyset)$ and $\exists t(r_i(t) \neq \emptyset)$. That is, we do not use sensors that do not detect T even once or that collide with T.

We define an operator $\mathcal{H} : \{1, 2, \ldots, N\} \to \{L, U\}$ as $\mathcal{H}(i) = L$ (if $i \in L$) and $\mathcal{H}(i) = U$ (if $i \in U$), where $i \in L$ ($i \in U$) means the i-th sensor is in L (U). The operator $\mathcal{H}(\cdot)$ tells us the half-plane to which a given sensor belongs. By definition, $i \in \mathcal{H}(i)$ for $\forall i$.

To summarize, the shape estimation problem we consider in this paper is described as follows. After T has passed through $\mathbf{\Omega}$, we have only v and a set of time-series data $\{r_i(t)\}_i$ and do not have any other information such as $\mathbf{x}_i$, θ_i, or the line trajectory of T. Our goal is to estimate the shape of T by analyzing $\{r_i(t)\}_i$.

Remark: The velocity v of T may be unknown, in which case we need to assume the uniform distribution of $\{\mathbf{x}_i\}_i$ and estimate v by using our previous method [11].

IV. Shape Estimation Method

We now explain shape estimation in detail. Shape estimation is carried out in two parts. First, we estimate all θs for all sensors that detect T. We then estimate the whole shape of T

using the estimated θs. Once θs are estimated, the second part is rather easy to carry out as follows. We assume that we have already obtained $\{\hat{\theta}_i\}_{i\in L}$ ($\hat{\theta}$ indicates the estimated value of θ). Let $t^* \equiv \mathrm{argmin}_t \bar{r}(t)$ and the coordinate of the bottom (top) be ${}^t(0,0) \in \partial T$. Then, as can be easily seen from Figure 2, the following relation holds:

$$\begin{pmatrix} X(t) \\ Y(t) \end{pmatrix} \equiv (\bar{r}(t) - \bar{r}(t^*)) \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} - v \begin{pmatrix} t - t^* \\ 0 \end{pmatrix} \in \partial T, \quad (2)$$

where $t \in \{t \mid \bar{r}(t) \in \mathbb{R},\ 0 < \bar{r}(t) < r_{max}\}$. In accordance with (2), we know the x and y coordinate ${}^t(X(t), Y(t))$ of each point of ∂T. Thus, we can depict the shape of T as follows. For the lower-half plane, by using a chosen estimator $\hat{\theta}_{i_1}$ ($i_1 \in L$), we depict a partial shape of the lower part of T by sweeping t in (2). If the depicted shape does not cover the whole lower part, we use another estimator $\hat{\theta}_{i_2}$ ($i_2 \in L$), depict another shape, and connect them by focusing on the bottom of T, which is a common point in two partial shapes. We repeat this procedure until covering the whole lower part. We do the same for sensors that have $\theta \in [\pi, 2\pi)$ (i.e., sensors in U) and obtain the shape of the upper part of T. Finally, we properly connect the lower and upper parts of T (for example, by focusing on points of $\xi(s) = \pi/2, 3\pi/2$) and obtain the whole shape.

From the above discussion, we can attribute the shape estimation problem to θ estimation. In this section, we explain two types of θ-estimation methods. In Section IV-A, we briefly review our previous curvature-based method [12]. This method derives several curvature-related equations that θs must satisfy and obtains θs by solving them. Although this method is theoretically important in the sense that numerically exact θs can be obtained, calculations of higher-order derivatives of $r(t)$ are required, which makes it difficult to apply it to practical cases in which $r(t)$ has some noise $\delta r(t)$. To overcome this difficulty, we designed the ICP-based method to estimate θs in the presence of noise of $r(t)$ by using an algorithm inspired from the ICP algorithm [13]–[15]. The ICP-based method does not require calculation $r(t)$ derivatives and can be applied to practical cases.

A. Difficulties With Curvature-Based θ Estimation

We review our previous curvature-based method [12] by mainly focusing on θ estimation in a two-sensor case and describe the reason this method is sensitive to sensing noise. For detailed explanation of this method, see the Appendix and our original paper [12]. Since this method does not take noise into account, we consider $r(t) = \bar{r}(t)$. This method is based on the following property of $r(t)$:

$$t^* = \mathrm{argmin}_t r(t) \Leftrightarrow s(t^*) \text{ is the bottom or top of } T, \quad (3)$$

where the t of argmin runs in the domain in which $r(t)$ takes a real value, not $\emptyset$. The property in (3) obviously follows from Figure 2. Assume that we obtain two sensing results $r_1(t)$ and $r_2(t)$ from two sensors. In accordance with (3), if $\mathcal{H}(1) = \mathcal{H}(2)$, then the point that sensor 1 detects at $t_1^* = \mathrm{argmin}_t r_1(t)$ and the point that sensor 2 detects at $t_2^* = \mathrm{argmin}_t r_2(t)$ are the same, namely, the bottom or top of T. When the bottom (top) is smooth, we derive the following simultaneous equations regarding the curvature and its derivative at the bottom (top):

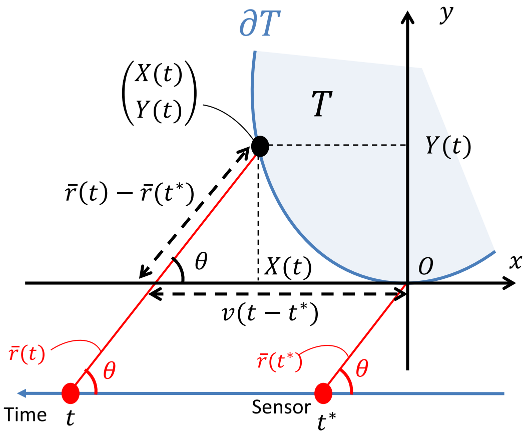


Fig. 2. Derivation of (2). In this figure, instead of moving the object in the positive x-direction, the sensor is moved in the negative x-direction. The red dots represent the positions of the sensor at time t^* and t, and the red line segments extending from them represent the sensing distance.

$$\ddot{r}_1(t_1^*)\sin\theta_1 = \ddot{r}_2(t_2^*)\sin\theta_2, \quad (4)$$

$$\begin{aligned} &\sin\theta_1(v\dddot{r}_1(t_1^*) + 3\ddot{r}_1(t_1^*)^2\cos\theta_1) \\ &= \sin\theta_2(v\dddot{r}_2(t_2^*) + 3\ddot{r}_2(t_2^*)^2\cos\theta_2). \end{aligned} \quad (5)$$

By solving these equations, we can obtain the numerically exact θ_1 and θ_2 in theory. In real situations, however, $r(t)$ is discretized and has noise $\delta r(t)$ at each t. Although we can smoothen $r(t)$ with regression methods, the calculated values of the higher order derivatives such as $\ddot{r}$ and $\dddot{r}$ still have non-negligible errors, making shape estimation practically impossible.

B. ICP-Based θ Estimation

1) Ideas and Strategies: Since the ICP-based method includes a mechanism to mitigate the effect of the measurement noise as described later, we consider an equation in which $\bar{r}$ in the partial-shape equation (2) is directly replaced with the measurement distance r. Now let us start by deriving another expression for (2). We assume that sensors 1 and 2 belong to the same half-plane: $\mathcal{H}(1) = \mathcal{H}(2)$. Let $t_i^* := \arg\min_t r_i(t)$, $\tilde{t}_i := t - t_i^*$, $\tilde{r}_i(\tilde{t}_i) := r_i(t) - r_i(t_i^*)$ for $i = 1, 2$. We can transform (2) to the following:

$$\begin{pmatrix} X_i \\ Y_i \end{pmatrix} = \begin{pmatrix} -1 & \cos\theta_i \\ 0 & \sin\theta_i \end{pmatrix} \begin{pmatrix} v\tilde{t}_i \\ \tilde{r}_i \end{pmatrix} =: A(\theta_i) \begin{pmatrix} v\tilde{t}_i \\ \tilde{r}_i \end{pmatrix}. \quad (6)$$

Hereafter, we simply rewrite $\tilde{t}_i$ and $\tilde{r}_i$ as t_i and r_i, respectively. We refer to the space in which T exists and the coordinate (X, Y) is given as the *real space* and the space to which the sensing results (vt_i, r_i) belong as the *measurement space* of sensor i. The ICP-based method first estimates pairwise θs from two sensing results, and then, by repeating this procedure for all pairs of θs, we obtain all θs. Thus, we first consider a two-sensor case and discuss the estimates of θ_1 and θ_2.

Our strategy is based on the following simple intuition. If θs are obtained correctly, the partial shapes reproduced from sensors belonging to the same half plane in accordance with (6) should overlap over a wide range. Conversely, if we can adjust θs so that the partial shapes obtained from (6) overlap over a wide range, the obtained θs are likely to be correct. Thus, the θ estimation problem comes down to "partial-shape matching." We carry out this process by using our variant of the ICP algorithm.

The ICP algorithm [13]–[15] is widely used in many fields, such as object recognition and robot sensing, to match two point clouds, one of which is called the reference cloud and the other the source cloud. In general, for each point belonging to the source cloud, we first find the closest point in the reference cloud and match them. The degree of matching of the reference and source clouds is expressed as an objective function, which is usually defined as a weighted summation of the distance between the matched point pairs. We determine the optimal parameters regarding translation and rotation, i.e., rigid-body transformation of the source cloud by solving the minimization problem of the given objective function. After transforming the source clouds with the obtained parameters, we repeat the above procedure from the closest-point-matching step until the objective function sufficiently converges.

To apply the ICP algorithm to our problem, we may consider two sensors' sensing results as the reference and source clouds. However, there are two challenges with our problem. One is that since the sensing results do not belong to the real space but measurement spaces, we need to consider another transformation to match the sensing results instead of rigid-body transformation. The other is that because two sensors having different θs generally detect different parts of T, the two partial shapes do not overlap completely even if θs are estimated correctly. This makes it difficult to achieve the global optimal solution. To overcome these challenges, we developed a variant of the ICP algorithm. The first problem is solved by finding a linear transformation that maps the measurement space of a sensor to that of another sensor. The second is addressed by gradually decreasing the threshold parameter d that determines which point pairs are involved in the objective function.

Algorithm 1 ICP-Based θ Estimation

```
Input $\{\boldsymbol{p}^i\}_i = \{(vt_1^i, r_1^i)\}_i, \{\boldsymbol{q}^j\}_j = \{(vt_2^j, r_2^j)\}_j, \theta_1^{ini}, \theta_2^{ini}, \epsilon$, $d$-decreasing-list
Output $\hat{\theta}_1, \hat{\theta}_2$
$\theta_1, \theta_2 \leftarrow \theta_1^{ini}, \theta_2^{ini}$
$m, M \leftarrow 10^5, 10^5$
// Decreasing $d$ gradually
for $d$ in $d$-decreasing-list do
  while $m > \epsilon$ do
    // Nearest point matching
    $\tilde{\boldsymbol{p}}^j \leftarrow \arg\min_{\boldsymbol{p} \in \{\boldsymbol{p}^i\}_i} \|R(\theta_1, \theta_2)\boldsymbol{q}^j - \boldsymbol{p}\|$ for each $j$
    // Minimization of objective function
    $(\theta_1, \theta_2) \leftarrow \arg\min_{\theta_1, \theta_2} \frac{1}{\tilde{N}} \sum_{j:\|\cdot\| \le d} \|R(\theta_1, \theta_2)\boldsymbol{q}^j - \tilde{\boldsymbol{p}}^j\|$
    $m \leftarrow M - \frac{1}{\tilde{N}} \sum_{j:\|\cdot\| \le d} \|R(\theta_1, \theta_2)\boldsymbol{q}^j - \tilde{\boldsymbol{p}}^j\|$
    $M \leftarrow \frac{1}{\tilde{N}} \sum_{j:\|\cdot\| \le d} \|R(\theta_1, \theta_2)\boldsymbol{q}^j - \tilde{\boldsymbol{p}}^j\|$
  end while
end for
$\hat{\theta}_1, \hat{\theta}_2 \leftarrow \theta_1, \theta_2$
```

2) Algorithm: We first explain the algorithm for estimating two sensors' directions θ_1 and θ_2. This is summarized in Algorithm 1. We assume having sensing results $\{\boldsymbol{p}^i\}_i = \{(vt_1^i, r_1^i)\}_i$ of sensor 1 and those $\{\boldsymbol{q}^j\}_j = \{(vt_2^j, r_2^j)\}_j$ of sensor 2. Thus far, we treated $r(t)$ as a continuous function of t, but in practice, sensors have finite time resolution of sensing and provide discretized finite plots. The subscripts i, j are indexes to distinguish each measured value of sensing results (Note that they are not indexes to distinguish sensors). We first determine the initial values of θ_1 and θ_2. It may be a good idea to carry out grid search for the objective function explained later and initialize θ_1 and θ_2. Next, we transform each measured point $\boldsymbol{q}^j$ to $R(\theta_1, \theta_2)\boldsymbol{q}^j$, where $R(\theta_1, \theta_2) := A(\theta_1)^{-1}A(\theta_2)$. Thereby, each measured point of sensor 2 is mapped into the measurement space of sensor 1. Since this is not rigid-body transformation (since $\{R(\theta_1, \theta_2)\}_{\theta_1, \theta_2}$ does not form a group under multiplication), we cannot multiply $\boldsymbol{q}^j$ by $R(\theta_1, \theta_2)$ iteratively as is usually done with the original ICP algorithm. We then find the closest point to $R(\theta_1, \theta_2)\boldsymbol{q}^j$ in the sensing results $\{\boldsymbol{p}^i\}_i = \{(vt_1^i, r_1^i)\}_i$ of sensor 1 and match them. We may use the Euclid distance as a metric of "closest." The matching process can be executed quickly by using space-searching methods, such as kd-tree. After that, we minimize the objective function, i.e., the error function $\frac{1}{\tilde{N}} \sum_{j:\|\cdot\| \le d} \|R(\theta_1, \theta_2)\boldsymbol{q}^j - \tilde{\boldsymbol{p}}^j\|$ with respect to θ_1 and θ_2, where $\tilde{\boldsymbol{p}}^j \in \{\boldsymbol{p}^i\}_i$ is the closest point to $R(\theta_1, \theta_2)\boldsymbol{q}^j$ and the summation is carried out only with respect to j that satisfies $\|R(\theta_1, \theta_2)\boldsymbol{q}^j - \tilde{\boldsymbol{p}}^j\| \le d$ (a given threshold) and $\tilde{N}$ is the number of such js. The j not satisfying $\|R(\theta_1, \theta_2)\boldsymbol{q}^j - \tilde{\boldsymbol{p}}^j\| \le d$ is not added to the sum because it is considered a non-overlapping sensing area of sensors 1 and 2. Using the obtained new θ_1 and θ_2, we iterate the above procedure from the nearest point matching until the objective function converges sufficiently.

How to set the threshold d with our algorithm is crucial to achieve convergence to the optimal solution. In the first half of iterations, since we need to make two outlines of sensing results close fast, we set a large d and make a large number of points contribute to the minimization. In the latter half of iterations, however, two outlines of the sensing results have similar forms and require fine tuning of θ_1 and θ_2. Thus, we should set a small d and reject the contribution of the matching pairs between distant points that may negatively affect the convergence. Taking such consideration into account, we gradually decrease d as the algorithm proceeds.

Remark: One may wonder why we carry out point matching in the measurement space not in the real space. Of course, we can formulate point matching and minimization problem in the real space as follows. We map $\{\boldsymbol{p}^i\}_i$ and $\{\boldsymbol{q}^j\}_j$ to the real space as $A(\theta_1)\boldsymbol{p}^i$ and $A(\theta_2)\boldsymbol{q}^j$. We then find the closest pairs and consider an objective function $\frac{1}{\tilde{N}} \sum_{k:\|\cdot\| \le d} \|A(\theta_1)\boldsymbol{p}^{i_k} - A(\theta_2)\boldsymbol{q}^{j_k}\|$, where $A(\theta_1)\boldsymbol{p}^{i_k}$ and $A(\theta_2)\boldsymbol{q}^{j_k}$ are the closest to each other in the real space. Although this seems equivalent to Algorithm 1, we found that this formulation yields a little worse performance than ours as a result of preliminary numerical calculations. This is probably because $\hat{\theta}$ tends to fall into a suboptimal in which the points of sensing results are arranged in a single line, i.e., $\theta \approx 0$ or π.

We now are in a position to discuss θ estimation in a general case. The basic idea of θ estimation is as follows. We obtain $N-1$ sets of estimators of θ_1 by applying the above algorithm for $2(N-1)$ pairs $(r_1, r_2), (r_1, r_3), \ldots, (r_1, r_N), (r_2, r_1), (r_3, r_1), \ldots, (r_{N-1}, r_1)$ and (r_N, r_1). If $\mathcal{H}(1) = \mathcal{H}(i)$ $(i \in \{2, 3, \ldots, N\})$, then the derived set of estimators using $\{(r_1, r_i)\}_{i=2}^{N}$ and $\{(r_i, r_1)\}_{i=2}^{N}$ pairs include the same (correct) value of $\hat{\theta}_1$. Otherwise, the pairs may not provide a solution or provide meaningless values. Therefore, pairs (r_1, r_i) providing the same $\hat{\theta}_1$ can be considered belonging to the same half-plane: $\mathcal{H}(1) = \mathcal{H}(i)$. Otherwise, r_1 and r_i are in a different half-plane: $\mathcal{H}(1) \neq \mathcal{H}(i)$. Similarly, we apply this algorithm to the pair (r_j, r_k), where $j > 1, \forall k \in \mathcal{H}(j)$ to determine $\hat{\theta}_j$. To alleviate the fluctuation stemming from sensing noise and numerical error, we conduct the above procedure more systematically as follows.

For $i \neq j$, let $\hat{\theta}_{i,1}(j)$ and $\hat{\theta}_{j,2}(i)$ be the θ_i and θ_j determined from our algorithm with (r_i, r_j) input, respectively, and $\hat{\theta}_{i,2}(j)$ and $\hat{\theta}_{j,1}(i)$ be the θ_i and θ_j determined from our algorithm with (r_j, r_i) input, respectively. Since Algorithm 1 is not a symmetric operation with respect to $\{\boldsymbol{p}^i\}_i$ and $\{\boldsymbol{q}^j\}_j$, $\hat{\theta}_{i,1}(j)$ and $\hat{\theta}_{i,2}(j)$ $(\hat{\theta}_{j,1}(i)$ and $\hat{\theta}_{j,2}(i))$ are not exactly equal due to measurement and numerical errors. Also, if $\mathcal{H}(i) \neq \mathcal{H}(j)$, then $\hat{\theta}_{i,1}(j)$ $(\hat{\theta}_{i,2}(j)$, $\hat{\theta}_{j,1}(i)$, $\hat{\theta}_{j,2}(i))$ should be meaningless and must be discarded. Therefore, in the following, we describe systematically obtaining the estimate of θ_i, which is denoted as $\hat{\theta}_i$, from $\{\hat{\theta}_{i,1}(j)\}_{j\neq i}$ and $\{\hat{\theta}_{i,2}(j)\}_{j\neq i}$. First, omit $\hat{\theta}_{i,k}$ such that $\hat{\theta}_{i,k} \simeq 0$ or π because $R(\theta_i, \theta_j)$ in our algorithm is unstable around these values and solutions around 0 and π are unreliable. We can also see from geometric considerations that most sensors with the direction around 0 or π cannot detect the target object (Note that we removed sensors that collide with the target object in advance, as explained in Section III.) Thus, this heuristic process does not negatively affect the estimation results. Second, from set $\{\hat{\theta}_{i,k}(j)\}_{j\neq i, k\in\{1,2\}}$, find a subset E_i such that all the elements in it have approximately the same value. In principle, $\{\hat{\theta}_{i,k}(j)\}_{j\in A(i), k\in\{1,2\}}$ should be almost the same values where $A(i) = \{j \in \{1, 2, \ldots, N\}\backslash\{i\} \mid \mathcal{H}(i) = \mathcal{H}(j)\}$, whereas $\{\hat{\theta}_{i,k}(j)\}_{j\in A(i)^c, k\in\{1,2\}}$ take meaningless random values where $A(i)^c = \{j \in \{1, 2, \ldots, N\}\backslash\{i\} \mid \mathcal{H}(i) \neq \mathcal{H}(j)\}$. Thus, we expect E_i to be $\{\hat{\theta}_{i,k}(j)\}_{j\in A(i), k\in\{1,2\}}$. If the objective function does not sufficiently converge, we can consider such sensors to belong to the latter subset $A(i)^c$. Finally, we obtain the estimates $\hat{\theta}_i$ as the median of E_i. The median is generally more robust against outliers than the mean.

Remark: If the target object is linearly symmetric with respect to the reference direction, then E_i obtained above is almost equal to the set of all sensors. In other words, we cannot divide the sensors into those belonging to the upper half-plane and those belonging to the lower half-plane. Given this fact, if the sensor set cannot be clearly classified into two sets, we assume that the object is symmetrical and estimate the object shape by determining the shape of the half-planes. We do not know to which half-plane each sensor belongs, but we can estimate the shape of the target object (under the assumption that it is symmetric).

The remaining task is to describe the shape of the object on the basis of (2) using the obtained $\hat{\theta}_i$, but it is not necessary to use all the sensing data if the whole shape of the target is covered. Since the accuracy of shape estimation can be considered higher if we use sensing data $r_i(t)$ with high θ_i estimation accuracy, we use $r_i(t)$ in the order of increasing quartile range of E_i to cover the whole target.

V. Numerical Simulations

We conducted extensive numerical simulations to evaluate the effectiveness of the ICP-based method, especially in the presence of sensing noise.

A. Default Conditions and Simulation Method

In this section, we use the following conditions unless explicitly mentioned otherwise. The $\boldsymbol{\Omega}$ is a rectangular area of 200×150, the longer edge of which is along the x-axis, and T is moving on the centerline parallel to the x-axis of $\boldsymbol{\Omega}$ at $v = 1$. We set $r_{max} = 100$ and $n_s = 100$. The sensors, the θs of which are randomly distributed in $[0, 2\pi)$, are uniformly distributed in $\boldsymbol{\Omega}$. Sensors reported $r(t)$ at $\Delta t = 0.05$ intervals during the simulations. At each t, $r(t)$ includes the true distance to T, $\bar{r}(t)$, and a random value $\delta r(t)$ that obeys an independent and identically distributed Gaussian noise with a mean of 0 and standard deviation σ of 0.05. Before applying the ICP-based method, however, we smoothed the sensing results around the minimum of $r(t)$ by polynomial regression. Specifically, we conducted polynomial regression from three to seven degrees and used the degree that minimizes the Akaike information criterion (AIC). We used two figures as T, which are (A) an ellipse (Ellipse): major diameter of 30 and minor diameter of 15, and (B) gourd shape (Gourd), as shown in Figure 3. Note that Gourd has a non-convex shape. To present a realistic interpretation of these examples, we may consider one unit of length to be 0.5 m and one unit of time to be 0.5 s. We evaluated estimation accuracy, which is defined as $R^2/PQ \in [0, 1]$, where P is the area of the actual target shape, Q the area of the estimated shape, and R the area of the intersection of the actual target and estimated shape. To define the intersection uniquely, we overlaid the bottom of the estimated shape with that of the actual shape. In the θ-estimation algorithm, we set $\epsilon = 10^{-7}$ and d-decreasing-list = [20, 10, 5, 3, 1]. We repeated each simulation five times by changing random seeds and averaged their results.

B. Impact of σ

We first investigated the impact of noise on estimation accuracy. Figure 4 shows the relation between σ and estimation accuracy derived from the ICP-based method (Proposed) and curvature-based method (CB) [12]. The error bars show the standard deviation of five trials. The accuracy of CB remained high under $\sigma < 10^{-7}$. As σ exceeded this value, however, it started to drop, and over $\sigma > 10^{-4}$, shape estimation was almost impossible. The fragility against noise is attributed to

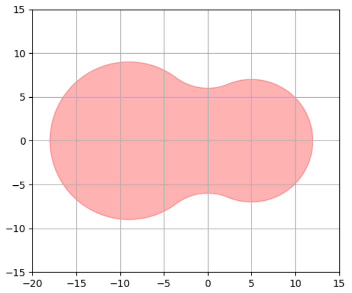


Fig. 3. Gourd, a gourd-shaped object used in the simulation. Unlike Ellipse, it has a non-convex shape. In the simulation, it moved in the positive *x*-direction (to the right) while maintaining the orientation shown in the figure.

the fluctuation in $A_l(i, j)$, which is defined in the Appendix, and essentially caused by the higher-order derivatives of $r(t)$, as discussed in our previous study [12]. On the other hand, the ICP-based method provided high estimation accuracy even in the range of $\sigma > 10^{-4}$. Since it does not rely on the curvature of $r(t)$ but the outline of shape, its noise resilience is higher than that of CB. Figure 5 plots the estimated θs and their errors for $\sigma = 10^{-4}, 0.05$ and 0.5 The vertical axis shows the estimated θs, the true values of which are given on the horizontal axis. The figure shows a box plot of E_i, which means that red signal indicates the median $Q_{2/4}$, edges of box indicate the lower quartile $Q_{1/4}$ and upper quartile $Q_{3/4}$, and closed interval indicates

$$[\max\{Q_{1/4} - 1.5\text{IQR}, \hat{\theta}_{min}\},\ \min\{Q_{3/4} + 1.5\text{IQR}, \hat{\theta}_{max}\}],$$

where IQR $= Q_{3/4} - Q_{1/4}$. If θs are correctly estimated, the red signals should line up on the blue dotted line. For $\sigma = 10^{-4}$, almost all θs were correctly estimated, and the errors were quite small. For $\sigma = 0.05$, some θs had relatively large fluctuation around the estimated values, which are still along the blue dotted line. This is thanks to taking the median of estimated $\{\hat{\theta}_{i,k}(j)\}$ as $\hat{\theta}_i$. The errors became larger for $\sigma = 0.5$ and some estimated values deviated from the true ones. However, for θ that has a small quartile range, the estimated values represented with the red signals, i.e., the median $\hat{\theta}_i$, are roughly along the blue dotted line. As described in the previous section, we used the sensors in order of increasing quartile range of E_i, so we were able to achieve high accuracy in shape estimation even with a large noise level such as $\sigma = 0.5$.

From these results, we can conclude that, in the presence of realistic noise, the ICP-based method outperforms CB and works well enough for shape estimation. In the following simulations, we imposed $\sigma = 0.05$ noise, which is unacceptable with CB. We present the results of the ICP-based method only.

C. Impact of n_s

As we can see in the previous results, the accuracy of the estimated θs largely relies on the statistical processing of $\{\hat{\theta}_{i,k}(j)\}$. Thus, we should investigate the impact of n_s on shape and θ estimation. Figure 6 shows the shape-estimation accuracy by varying n_s from 25 to 200. We found that the ICP-based method still provides a good result even for $n_s = 25$. This is consistent with the θ-estimation shown in Figure 7, which shows that the estimated values denoted with red signals line up on the blue dotted line. From this observation, even if we make n_s smaller than 25, θ estimation would be possible. However, the number of sensors may be too small to sense the whole target object; thus, shape estimation would fail.

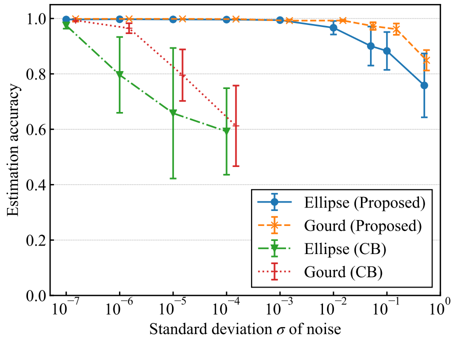
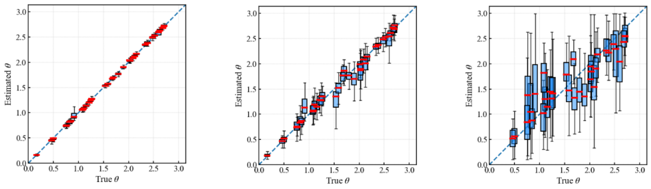


Fig. 4. Impact of σ on shape estimation.



(a) $\sigma = 10^{-4}$ (b) $\sigma = 0.05$ (c) $\sigma = 0.5$

Fig. 5. Impact of σ on θ estimation (Ellipse).

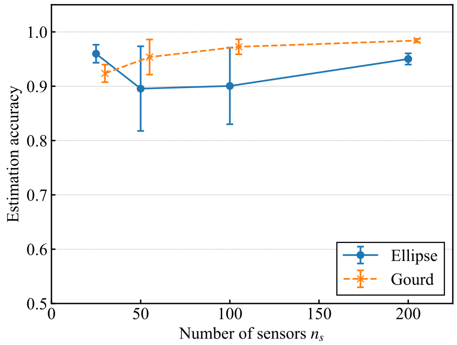
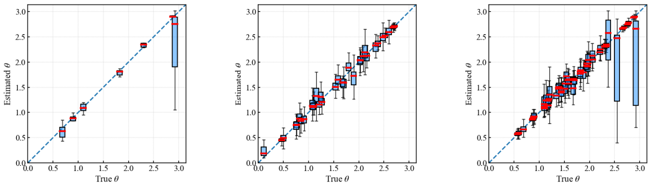


Fig. 6. Impact of n_s on shape estimation.



(a) $n_s = 25$ (b) $n_s = 100$ (c) $n_s = 200$

Fig. 7. Impact of n_s on θ estimation (Gourd).

D. Impact of v

We then investigated the relation between v and shape-estimation accuracy. Figure 8 shows the results. For Ellipse, the shape-estimation accuracy was quite robust against the increase in v. The θ estimation was not much affected by the change in v, as shown in Figure 9. However, we found that the shape-estimation accuracy of Gourd gradually worsened as v increased. The reason for this can be considered as

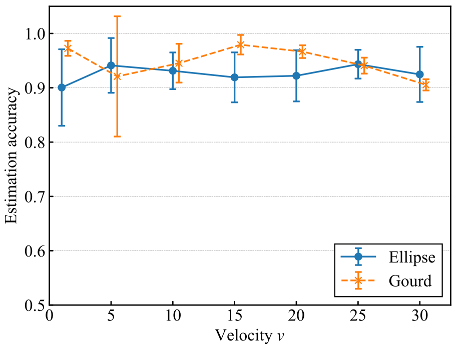
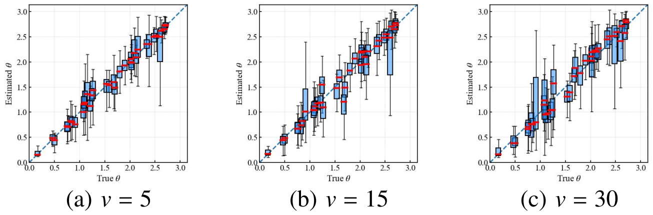


Fig. 8. Impact of v on shape estimation.



(a) $v = 5$ (b) $v = 15$ (c) $v = 30$

Fig. 9. Impact of v on θ estimation (Ellipse).

follows. Since Gourd had a more complex shape than Ellipse, it required more sensing points to represent the outline of its shape. With the increase in v, however, the interval of sensing points becomes wider and the number of sensing points lessens. This leads to inaccurate point matching when applying the ICP algorithm. For essentially the same reason, we obtained similar results when changing the size of T.

E. Impact of v Inaccuracy

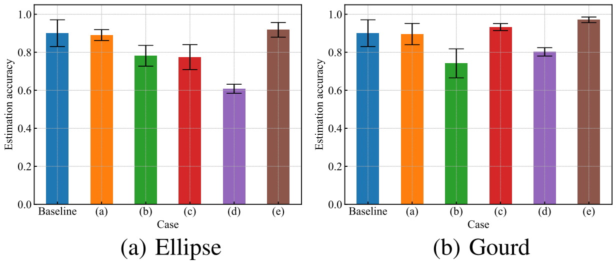
As another aspect of the impact of v, we should consider the impact of v inaccuracy. Although we assume that v is constant with the ICP-based method, a T, such as a car, may gradually change v or road conditions may make fluctuate v. Thus, considering five specific cases: (a)/(b) Acceleration, (c)/(d) Deceleration, and (e) Noise of v, we investigated their impact on estimation accuracy. Note that we applied the θ-estimation algorithm assuming v to be 1.0 taking the position that we do not know the change or fluctuation in v.

The settings of all cases are shown in Table I. For cases (a)–(d), we set the initial v to v_i and the acceleration rate along the same axis as the velocity to a, in which case the final v of T when T exited Ω is v_f. Cases (a) and (b) are when T accelerates slowly and quickly, respectively. Cases (c) and (d) are when T decelerates slowly and quickly, respectively. For case (e), we imposed Gaussian noise with a mean of 0 and standard deviation (SD) of 0.05 on $v = 1$ at each t.

Figure 10 summarizes the results of shape-estimation accuracy, and Figure 11 shows those of θ estimation for Ellipse in each case. Baseline shows the case in which T moves at a constant velocity $v = 1$. When the acceleration rate was small, the ICP-based method was robust, even when v was not accurate. With the increase in the acceleration rate, accuracy decreased although shape estimation was still possible. The ICP-based method was also robust against noisy velocity changes. These results indicate that the ICP-based method is robust with respect to random errors in v, but less accurate in the presence of systematic errors in v.

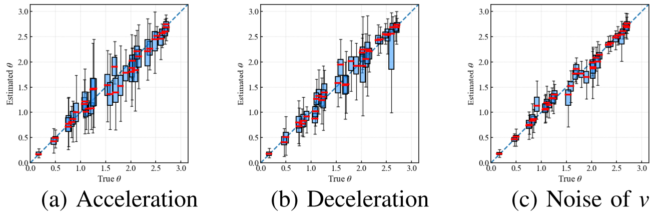
TABLE I
Simulation Settings

	case	v_i	v_f	a
Acceleration	(a) (b)	0.95 0.90	1.05 1.10	+0.0005 +0.0010
Deceleration	(c) (d)	1.05 1.10	0.95 0.90	−0.0005 −0.0010
Noise of v	(e)	$v = 1$ with noise SD of which is 0.05		



(a) Ellipse (b) Gourd

Fig. 10. Estimation accuracy for inaccurate v.



(a) Acceleration (b) Deceleration (c) Noise of v

Fig. 11. Impact of v inaccuracy on θ estimation (Ellipse).

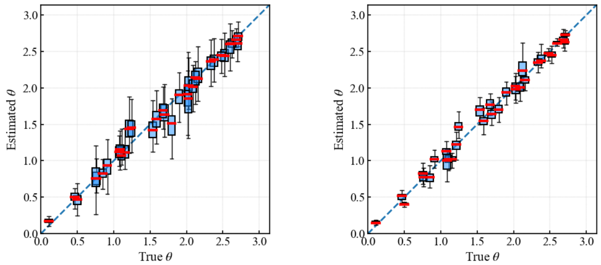
F. Non-Straight Line Moving

Finally, we considered the case in which T moves in a non-straight line. Although with the ICP-based method it is assumed that T moves in a straight line, real routes of a T, such as a car, could move in a curve. We set v (along the x axis) to 1.0, initial velocity along the y axis to 0, and acceleration along the y axis to 0.0005. That is, T moved on a parabola curve and its projection on the x axis moved with a constant velocity. The head of T kept facing in the direction of the tangent of the trajectory at each t. Note that we applied the θ-estimation algorithm assuming the trajectory of T to be straight and taking the position that we do not know the curve of the trajectory.

The shape-estimation accuracies were 0.973 and 0.978 for Ellipse and Gourd, respectively. Figure 12 shows the θ-estimation accuracy for both target objects. As a whole, we can conclude that the ICP-based method is also robust against a gentle curve of a T's trajectory.

VI. Experiment

We conducted an experiment with an actual optical sensor to evaluate the feasibility of the ICP-based method in practical situations.



(a) Ellipse (b) Gourd

Fig. 12. Impact of curved trajectory on θ estimation.

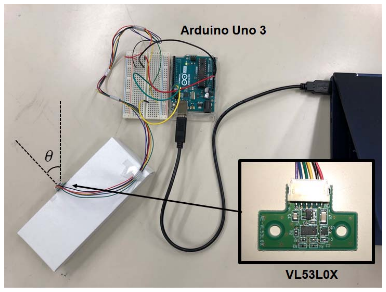


Fig. 13. Photograph of devices used in the experiment.

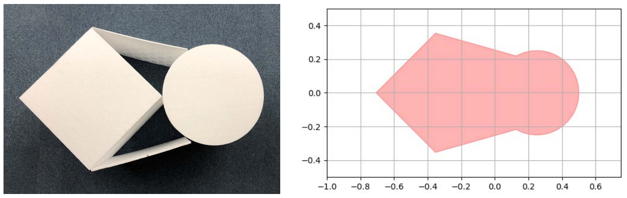
A. Experimental Settings

We used the commercial laser-ranging sensor module VL53L0X developed by STMicroelectronics NV. It can measure distance with a few-millimeter error at an interval of about $\Delta t = 0.025$ s. We obtained the time-series data through the open-source microcontroller board Arduino Uno R3, as shown in Figure 13. We prepared the non-convex shaped object shown in Figure 14a the size (in meters) of which is shown in Figure 14b. Instead of distributing many sensors, we repeated measurements using a single sensor. Specifically, we fixed the sensor's position and direction in accordance with random numbers generated in advance and moved the target object at a constant speed of about $v = 0.1$ m/s. By repeating this procedure, we obtained $\{r_i(t)\}_i$. Note that the obtained data here are equivalent to data obtained simultaneously with multiple sensors because the ICP-based method does not require $\{r_i(t)\}_i$ to be time-synchronized. We collected 30 sensing results of $r(t)$ and determined θs from them. As mentioned in Section IV-B.2, we assumed the object to be symmetric.

Although the main purpose of this experiment is to clarify whether object-shape estimation is possible even in the presence of naturally occurring sensing noise and fluctuations in v, we can find the following concrete application aspects for this experiment. The shape estimation using distance sensors discussed in this paper does not require the precision of identifying a human face but the granularity of classifying what the category of the object is (e.g., human, family car, truck). Therefore, in this experimental setup, we can consider that we are discussing whether we can estimate the intrusion of an object of the size of a wheelbarrow, for example, by capturing its shape characteristics, such as convexity.

B. Results

Figure 15 plots the results of θ estimation. We found that θs can be estimated rather well. Figure 16 shows the results of shape estimation. The red area denotes the actual shape of T, and the blue curves indicate the estimated shape of T. When we depicted this shape, we used the linear symmetry of T and overlaid the bottom (top) of the estimated shape with that of the actual shape. Figures 16a and 16b show the estimated shape using 5 and 15 sensors, respectively. The 5 or 15 sensors were chosen in order of increasing quartile range E_i, as described in Section IV-B.2. Even though θs were accurately estimated, $\{r_i(t)\}_i$ were affected by sensing noise and v noise. Thus, the estimated shape was reflected by which sensors were used in estimating the shape. The shape estimated with only 5 sensors were partially lacking, especially around the head of T. When using 15 sensors, the whole shape was almost covered. Although there were differences in the estimated shapes at the front and rear of the object depending on the sensor used, the concavity of the sides and the convexity of the obtuse angle, which are the characteristics of the shape, could be captured, indicating that the object can be discriminated adequately.



(a) Photograph (b) Size in meters

Fig. 14. Target object used in the experiment.

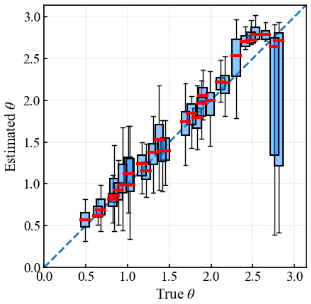
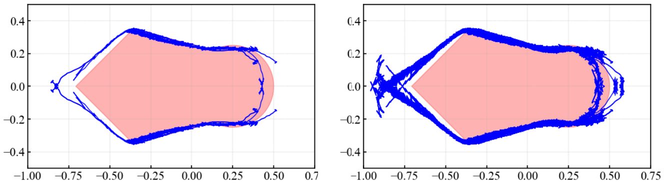


Fig. 15. Estimated θs and their uncertainty obtained from the experiment.



(a) 5 sensors (b) 15 sensors

Fig. 16. Actual shape (red area) and estimated shape (blue curves).

VII. Conclusion

We proposed a method for estimating the shape of a moving target object from the sensing results of distributed distance sensors. We considered a situation in which we do not have any prior knowledge on sensor location, direction, or distribution. Although our previous study showed that shape estimation would be theoretically possible under such an information-lacking condition, our curvature-based method is too sensitive to sensing noise to be applied in practical situations. The proposed method can be applied to such realistic situations. It can successfully obtain the directions of sensors through our variant of the ICP-based algorithm by avoiding the calculation of higher order derivatives, which is sensitive to noise. To evaluate the effectiveness of our method, we conducted extensive numerical simulations, demonstrating that it is quite robust against noise of sensing results and velocity. Although it is assumed with our method that the target object is moving on a straight line and its velocity is constant, we found that it performs well even when the assumption is not exactly satisfied. We also conducted an experiment using an actual laser sensor, showing that shape estimation almost succeeds in the presence of real noise. Therefore, this study, in conjunction with our previous studies, supports the hypothesis that shape estimation is theoretically possible and also practical to some extent even without imposing elaborate conditions of distributed sensors. We believe that this study opens up a new direction in sensor network research, such as research on what prior conditions and information are necessary in making shape estimation possible or more accurate.

The following are also worth addressing for further study. (1) We need to develop a method applicable to multiple target objects. (2) In realistic situations, there are obstacles in the monitored area. If obstacles do not move, the proposed method can be applied only using time-variant $r(t)$. Therefore, it needs to be extended to cases in which moving obstacles exist. (3) The accuracy of the proposed method may be improved by considering the noise characteristics of the sensors. For example, since the variance of the measurement noise depends on the angle of incidence of the laser to the target, it cannot be assumed that all sensors share the same noise characteristics. Therefore, weighting the measurements in accordance with the estimated angle may lead to more accurate shape estimation.

Appendix

A. Curvature-Based θ Estimation

In Appendix, we assume that $r(t)$ does not include noise, i.e., $r(t) = \bar{r}(t)$. First, we introduce important relations in preparation for the θ estimation. See our previous study [12] for derivations and detailed discussion. By considering the linear motion of the target object within an infinitesimal time interval, we obtain the slope $\xi(s)$, curvature $\kappa(s) \equiv d\xi/ds$, and its derivative $d\kappa/ds$ using $r(t)$ as follows:

$$\tan\xi = \frac{\dot{r}\sin\theta}{\dot{r}\cos\theta - v}, \tag{7}$$

$$\kappa(s) = \frac{\pm v}{(1+\tan^2\xi)^{3/2}} \frac{\ddot{r}\sin\theta}{(\dot{r}\cos\theta - v)^3}, \tag{8}$$

$$\frac{d\kappa}{ds} = \frac{1}{(1+\tan^2\xi)^3} \times \frac{v\sin\theta\left(\dot{r}(\dddot{r}\dot{r} - 2\dddot{r}v\cos\theta - 3\ddot{r}^2) + \dddot{r}v^2 + 3\ddot{r}^2 v\cos\theta\right)}{(\dot{r}\cos\theta - v)|\dot{r}\cos\theta - v|^5}. \tag{9}$$

where the dots over r (such as $\dot{r}, \ddot{r}$) denote the derivatives with respect to t, and $\pm$ corresponds to $\cos\xi \lessgtr 0$.

The following property of $r(t)$ is also important:

$$t^* = \mathrm{argmin}_t r(t) \Leftrightarrow s(t^*) \text{ is the bottom or top of } T, \tag{10}$$

where the t of argmin runs in the domain where $r(t)$ takes a real value, not $\emptyset$.

We then consider a two-sensor case before discussing θ estimation in the general case. Assume that we obtain two sensing results $r_1(t)$ and $r_2(t)$ from two sensors. In accordance with (10), if $\mathcal{H}(1) = \mathcal{H}(2)$, then the point that sensor 1 detects at $t_1^* = \mathrm{argmin}_t r_1(t)$ and the point that sensor 2 detects at $t_2^* = \mathrm{argmin}_t r_2(t)$ are the same, namely, the bottom or top of T. When the bottom (top) is smooth, we can determine θ_1 and θ_2 by solving the following simultaneous equations obtained by substituting $\dot{r}_1(t_1^*) = \dot{r}_2(t_2^*) = 0$ into (8) and (9):

$$\ddot{r}_1(t_1^*)\sin\theta_1 = \ddot{r}_2(t_2^*)\sin\theta_2, \tag{11}$$

$$\sin\theta_1(v\dddot{r}_1(t_1^*) + 3\ddot{r}_1(t_1^*)^2\cos\theta_1) = \sin\theta_2(v\dddot{r}_2(t_2^*) + 3\ddot{r}_2(t_2^*)^2\cos\theta_2). \tag{12}$$

We can derive similar expressions when the bottom (top) is not smooth.

Now, we are in a position to discuss θ estimation in the general case. Since this method relies on the calculation of higher-order derivatives of $r(t)$, the estimated θs tend to fluctuate greatly. Thus, we need to alleviate the fluctuation in solutions by the following statistical processing.

1) Estimating Values of {(11), (12)}: Assume that the bottom (top) is smooth. Let $A_1(i, j)$ and $A_2(i, j)$ be the values of (11) and (12) for a pair of sensors (i, j), respectively. To be precise, obtain $\hat{\theta}_i$ and $\hat{\theta}_j$ by solving (11) and (12) with $(r_i(t_i^*), r_j(t_j^*))$ and let $A_1(i, j) \equiv \ddot{r}_i(t_i^*)\sin\hat{\theta}_i$ and $A_2(i, j) \equiv \sin\hat{\theta}_i(v\dddot{r}_i(t_i^*) + 3\ddot{r}_i(t_i^*)^2\cos\hat{\theta}_i)$.

We first obtain $A_1(i, j)$ and $A_2(i, j)$ for all pairs of sensors $(i, j = 1, 2, \ldots N)$ by solving (11) and (12). If there is no noise or error, we have $A_l(i, j) \equiv C_l^L$ $(const.)$ for $\forall i, j \in L$ $(l = 1, 2)$ and $A_l(i, j) \equiv C_l^U$ $(const.)$ for $\forall i, j \in U$ $(l = 1, 2)$. Thus, we can find two distinct subsets in $\{A_l(i, j)\}_{i,j}$. The $\{A_{l,1}(i, j)\}_{i,j\in L}$ is a subset corresponding to L, and $\{A_{l,2}(i, j)\}_{i,j\in U}$ is the other subset corresponding to U $(l = 1, 2)$. All the elements in each have almost the same values. If there seems to be fluctuations in $A_l(i, j)$'s because of noise or error, it is a good idea to apply a certain classification tool to obtain two subsets.

We then obtain $\hat{A}_{l,m}$ defined as the median of $\{A_{l,m}(i, j)\}_{i,j}(l, m = 1, 2)$. The $\hat{A}_{l,m}$ can be considered the estimated values of (11) and (12). Note that median is more robust against outliers than mean.

2) Determining θ in Ascending Order of SEM: By solving $\{(11), (12)\}$, we have already obtained θs. However, because the values of $\{A_{l,m}(i, j)\}_{i,j}$ fluctuate, the obtained θs also have errors. Thus, we find θs in ascending order of the square error from median (SEM) in the following manner.

For each $m = 1, 2$, determine (sufficiently many) n sensor pairs (i, j) in ascending order of $(\hat{A}_{1,m} - A_{1,m}(i, j))^2 + (\hat{A}_{2,m} - A_{2,m}(i, j))^2$, which means the SEM of $\{A_{l,m}(i, j)\}_{i,j}$. Let $(\tilde{i}_k, \tilde{j}_k)$ be the

k-th smallest SEM sensor pairs $(1 \le k \le n)$ and $\{\hat{\theta}(p,m)\}_p = \{\hat{\theta}(\tilde{i}_1,m), \hat{\theta}(\tilde{j}_1,m), \hat{\theta}(\tilde{i}_2,m), \hat{\theta}(\tilde{j}_2,m), \ldots, \hat{\theta}(\tilde{i}_n,m), \hat{\theta}(\tilde{j}_n,m)\}$ be their θs, where $\hat{\theta}(\tilde{i}_k,m), \hat{\theta}(\tilde{j}_k,m)$ are derived from $A_{1,m}(\tilde{i}_k,\tilde{j}_k), A_{2,m}(\tilde{i}_k,\tilde{j}_k)(1 \le k \le n)$. Because we believe that $\hat{\theta}$ derived from a smaller SEM is more accurate, we use $\hat{\theta}$ in order of $\{\hat{\theta}(p,m)\}_p$ in estimating the whole shape of T.

References

[1] *LoRa Alliance*. Accessed: Jan. 21, 2022. [Online]. Available: https://www.lora-alliance.org/

[2] *NarrowBand IOT*. Accessed: Jan. 21, 2022. [Online]. Available: https://www.3gpp.org/news-events/3gpp-news/1733-niot

[3] H. Saito, O. Kagami, M. Umehira, and Y. Kado, "Wide area ubiquitous network: The network operator's view of a sensor network," *IEEE Commun. Mag.*, vol. 46, no. 12, pp. 112–120, Dec. 2008.

[4] M. Umehira, H. Saito, O. Kagami, T. Fujita, and Y. Fujino, "Concept and feasibility study of wide area ubiquitous network for sensors and actuators," in *Proc. IEEE 65th Veh. Technol. Conf. VTC-Spring*, Apr. 2007, pp. 165–169.

[5] *Sigfox - The Global Communications Service Provider for the Internet of Things (IoT)*. Accessed: Jan. 21, 2022. [Online]. Available: https://www.sigfox.com/

[6] *SMART DUST*. Accessed: Jan. 21, 2022. [Online]. Available: https://people.eecs.berkeley.edu/~pister/SmartDust/

[7] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Commun. ACM*, vol. 43, no. 5, pp. 51–58, 2000.

[8] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102–114, Aug. 2002.

[9] B. W. Cook, S. Lanzisera, and K. S. Pister, "SoC issues for RF smart dust," *Proc. IEEE*, vol. 94, no. 6, pp. 1177–1196, Jun. 2006.

[10] C. Zhu, C. Zheng, L. Shu, and G. Han, "A survey on coverage and connectivity issues in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 35, no. 2, pp. 619–632, Mar. 2012.

[11] H. Saito and H. Ikeuchi, "Estimating shape of target object moving on unknown trajectory by using location-unknown distance sensors: Theoretical framework," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2018, pp. 4118–4125.

[12] H. Ikeuchi and H. Saito, "Shape estimation using location-unknown distance sensors: A curvature based approach," in *Proc. 15th Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, May 2019, pp. 90–97.

[13] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 1991, pp. 2724–2795.

[14] P. J. Besl and N. D. Mckay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 2002.

[15] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. 3rd Int. Conf. 3-D Digit. Imag. Modeling*, May 2001, pp. 145–152.

[16] A. A. Kannan, B. Fidan, and G. Mao, "Analysis of flip ambiguities for robust sensor network localization," *IEEE Trans. Veh. Technol.*, vol. 59, no. 4, pp. 2057–2070, May 2010.

[17] G. Mao, B. Fidan, and B. D. O. Anderson, "Wireless sensor network localization techniques," *Comput. Netw.*, vol. 51, no. 10, pp. 2529–2553, 2007.

[18] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal, "Locating the nodes: Cooperative localization in wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 54–69, Jul. 2005.

[19] A. K. Paul and T. Sato, "Localization in wireless sensor networks: A survey on algorithms, measurement techniques, applications and challenges," *J. Sens. Actuator Netw.*, vol. 6, no. 4, p. 24, 2017.

[20] F. Gustafsson and F. Gunnarsson, "Mobile positioning using wireless networks: Possibilities and fundamental limitations based on available wireless network measurements," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 41–53, Jul. 2005.

[21] A. H. Sayed, A. Tarighat, and N. Khajehnouri, "Network-based wireless location: Challenges faced in developing techniques for accurate wireless location information," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 24–40, Jul. 2005.

[22] G. Han, J. Jiang, C. Zhang, T. Q. Duong, M. Guizani, and G. K. Karagiannidis, "A survey on mobile anchor node assisted localization in wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2220–2243, 3rd Quart., 2016.

[23] J. C. Chen, R. E. Hudson, and K. Yao, "Maximum-likelihood source localization and unknown sensor location estimation for wideband signals in the near-field," *IEEE Trans. Signal Process.*, vol. 50, no. 8, pp. 1843–1854, Aug. 2002.

[24] X. Nguyen, M. I. Jordan, and B. Sinopoli, "A kernel-based learning approach to ad hoc sensor network localization," *ACM Trans. Sensor Netw.*, vol. 1, no. 1, pp. 134–152, 2005.

[25] C.-H. Ou, "A localization scheme for wireless sensor networks using mobile anchors with directional antennas," *IEEE Sensors J.*, vol. 11, no. 7, pp. 1607–1616, Jul. 2011.

[26] Y. Ahmadi, N. Neda, and R. Ghazizadeh, "Range free localization in wireless sensor networks for homogeneous and non-homogeneous environment," *IEEE Sensors J.*, vol. 16, no. 22, pp. 8018–8026, Nov. 2016.

[27] Y. Wu, Y. Wang, and G. Cao, "Photo crowdsourcing for area coverage in resource constrained environments," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, May 2017, pp. 1–9.

[28] I. Abraham, A. Prabhakar, M. J. Z. Hartmann, and T. D. Murphey, "Ergodic exploration using binary sensing for nonparametric shape estimation," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 827–834, Apr. 2017.

[29] H. Saito and H. Honda, "Geometric analysis of estimability of target object shape using location-unknown distance sensors," *IEEE Trans. Control Netw. Syst.*, vol. 6, no. 1, pp. 94–103, Mar. 2019.

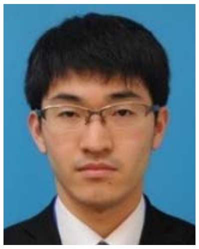
[30] H. Saito, S. Shimogawa, S. Shioda, and J. Harada, "Shape estimation using networked binary sensors," in *Proc. IEEE INFOCOM 28th Conf. Comput. Commun.*, Apr. 2009, pp. 2901–2905.

[31] H. Saito, "Local information, observable parameters, and global view," *IEICE Trans. Commun.*, vol. E96.B, no. 12, pp. 3017–3027, 2013.

[32] H. Saito, S. Tanaka, and S. Shioda, "Stochastic geometric filter and its application to shape estimation for target objects," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4971–4984, Oct. 2011.

[33] H. Saito, S. Shimogawa, S. Tanaka, and S. Shioda, "Estimating parameters of multiple heterogeneous target objects using composite sensor nodes," *IEEE Trans. Mobile Comput.*, vol. 11, no. 1, pp. 125–138, Jan. 2012.

[34] H. Saito and T. Kimura, "Theoretical framework for estimating target-object shape by using location-unknown mobile distance sensors," *IEEE Trans. Mobile Comput.*, vol. 19, no. 5, pp. 1233–1246, May 2020.



Hiroki Ikeuchi received the B.S. and M.S. degrees in physics from the University of Tokyo in 2014 and 2016, respectively. He joined NTT Laboratories, Musashino, Japan, in 2016. His research interest includes wireless sensor network and automation of network operation. He is a member of IEICE and the Physical Society of Japan (JPS). He received the Network Systems Research Award (IEICE Technical Committee on Network Systems) and the Young Researcher's Award (IEICE) in Japan, in 2021.

Hiroshi Saito (Fellow, IEEE) received the B.E. degree in mathematical engineering in 1981, the M.E. degree in control engineering in 1983, and the Dr.Eng. degree in teletraffic engineering from the University of Tokyo in 1992. He joined NTT in 1983. Since 2018, he has been a Professor with The University of Tokyo. His research interests include traffic technologies of communications systems, network architecture, and applied mathematics in communications systems. He is a Fellow of IEICE and ORSJ and a member of IFIP WG 7.3. He received the Young Engineer Award from the Institute of Electronics, Information and Communication Engineers (IEICE) in 1990, the Telecommunication Advancement Institute Award in 1995 and 2010, the Excellent Papers Award of the Operations Research Society of Japan (ORSJ) in 1998, the ACM MSWiM Conference Best Paper Award in 2016, and the Arne Jensen Lifetime Achievement Award in 2020. He has served as an Editor and a Guest Editor of technical journals such as *Performance Evaluation*, *Computer Networks*, IEEE JOURNAL OF SELECTED AREAS IN COMMUNICATIONS, and IEICE *Transactions on Communications*. He is the organizing Committee chairperson and the Program Committee chairperson of several international conferences. He is a TPC Member of more than 40 international conferences and the Director of the Journals and Transactions of IEICE. More information can be found at http://www9.plala.or.jp/hslab.