IEICE TRANS. FUNDAMENTALS, VOL.E105–A, NO.6 JUNE 2022

PAPER

Machine-Learning Approach for Solving Inverse Problems in Magnetic-Field-Based Positioning

Ai-ichiro SASAKI[†a)], *Member and* **Ken FUKUSHIMA**[††], *Nonmember*

SUMMARY Magnetic fields are often utilized for position sensing of mobile devices. In typical sensing systems, multiple sensors are used to detect magnetic fields generated by target devices. To determine the positions of the devices, magnetic-field data detected by the sensors must be converted to device-position data. The data conversion is not trivial because it is a nonlinear inverse problem. In this study, we propose a machine-learning approach suitable for data conversion required in the magnetic-field-based position sensing of target devices. In our approach, two different sets of training data are used. One of the training datasets is composed of raw data of magnetic fields to be detected by sensors. The other set is composed of logarithmically represented data of the fields. We can obtain two different predictor functions by learning with these training datasets. Results show that the prediction accuracy of the target position improves when the two different predictor functions are used. Based on our simulation, the error of the target position estimated with the predictor functions is within 10 cm in a 2 m × 2 m × 2 m cubic space for 87% of all the cases of the target device states. The computational time required for predicting the positions of the target device is 4 ms. As the prediction method is accurate and rapid, it can be utilized for the real-time tracking of moving objects and people.

key words: *inverse problems, machine learning, magnetic fields, nearest neighbor searches, optimization*

1. Introduction

The importance of position sensing technologies will increase in the Internet of Things era. Therefore, position sensing technologies are currently being investigated by many researchers using various approaches [1]–[5].

Position sensing technologies can be categorized into two main streams. The first mainstream is technologies using electromagnetic waves [3]–[5]. Position sensing systems that use electromagnetic waves are suitable for covering wide areas. However, it is difficult to establish accurate position sensing systems with electromagnetic waves because they are easily reflected by obstacles, walls, and human bodies [1]. As it is difficult to completely predict the behaviors of the reflected waves, the prediction accuracy of target positions is inevitably degraded.

The other main stream is position sensing technologies with magnetic fields, which are further divided into two approaches. One of the approaches utilizes geomagnetic fields, which are static fields [1], [2]. This type of systems usually adopt fingerprinting techniques and they are suitable for covering relatively large areas. In these systems, geomagnetic-field patterns are used as the fingerprint of the area where the position sensing systems are to be located. A main limitation of the geomagnetic-field approach is that the field patterns must be measured beforehand for obtaining the fingerprint. The high-resolution field patterns are required for accurate position sensing. However, it is a quite cumbersome task to measure the field patterns.

The other approach is based on artificially generated magnetic fields, which are usually low-frequency (LF) magnetic fields [5]–[24]. In this paper, we treat this type of position sensing systems. From a viewpoint of position sensing accuracy in non-line-of-sight (NLOS) environments, the approach with artificially generated magnetic fields is considered to be best [2]. Furthermore, the cumbersome task for obtaining the fingerprint is not required with this approach. Although the artificially generated magnetic fields cannot cover wide areas, they offer accurate position sensing of devices existing in a relatively small area. This is because the interaction between magnetic fields and objects is generally significantly weaker than that between electromagnetic waves and objects. Consequently, magnetic-field patterns are minimally affected by surrounding objects and obstacles. Technologies for extending the cover area of magnetic-field-based sensing systems has been investigated [25], [26]. It is also valid for extending the cover area to combine electromagnetic waves with the magnetic-field-based approach [3], [4].

In typical position sensing systems, magnetic fields generated from a target device are detected by multiple sensors that are located at different positions inside a target area. Since our purpose is sensing positions of the device, the magnetic-field data detected by the sensors must be processed for obtaining the position data of the target device. The conversion from sensor data to position data is not trivial because it is a nonlinear inverse problem. The subject treated in this paper is to improve the method for solving the inverse problem. Several approaches have been proposed to solve the inverse problem.

One of them involves the direct calculation of position data from sensor data using closed-form formulae [8]–[10]. An advantage of the closed-form approach is the speed of data conversion. This approach enables the real-time tracking of moving objects. However, closed-form formu-

Manuscript received June 16, 2021.
Manuscript revised October 28, 2021.
Manuscript publicized December 13, 2021.

[†]The author is with the Dept. of Electronic Engineering and Computer Science, Kindai University, Higashihiroshima-shi, 739-2116 Japan.
[††]The author is with the Kansai Transmission and Distribution, Inc., Osaka-shi, 530-0005 Japan.
a) E-mail: aisasaki@hiro.kindai.ac.jp

DOI: 10.1587/transfun.2021EAP1063

lae are obtained only for specific sensor configurations such as numbers, positions, and angles of sensor coils. Therefore, this approach is applicable only to systems of specific sensor configurations. This prevents us from designing position sensing systems flexibly.

The second approach is to solve an optimization problem that is reduced from the inverse problem. The position of target devices can be predicted from the sensor data by minimizing objective functions [11]–[13]. As the optimization approach is applicable to any sensor configurations, it is superior to the closed-form approach in terms of flexibility in system design. However, the optimization approach requires a relatively longer computational time for solving optimization problems, which may complicate certain applications such as the real-time tracking of moving devices. Furthermore, it is often difficult for beginners to obtain appropriate solutions because the optimization approach requires certain skill. We often obtain inappropriate solutions caused by the problem of local minima.

It is now common knowledge that machine learning is effective in various fields [27]–[30]. It was also applied to position sensing technologies that utilize electromagnetic waves [31]–[35] and geomagnetic fields [36]–[42]. Since various software tools for machine learning appear recent years, even beginners can effectively apply machine learning for their own purposes. However, there had not been any attempts to apply machine learning for position sensing that uses artificially generated magnetic fields.

Recently, we first proposed a machine-learning method applicable to position sensing with artificially generated magnetic fields [43]. It was demonstrated that the position of mobile devices can be predicted more accurately within a shorter computational time through machine learning in comparison with the conventional optimization approach. Furthermore, the machine-learning approach is also applicable to any sensor configurations. Therefore, the approach offers considerable flexibility in system design. The only weakness of our machine-learning approach is that the prediction accuracy is degraded when the target device exists in the vicinity of one of the sensors [43].

In this study, we propose methods to overcome the weakness of the machine-learning approach. In Sect. 2, we briefly review the basic concepts of position prediction with machine learning [43]. Our original proposals and results are described in Sects. 3, 4, and 5. In Sect. 3, a new method to create training data is explained. A primary aspect of the method is to logarithmically represent magnetic-field data that are used as training data. In Sect. 4, we demonstrate that the prediction accuracy in the vicinity of the sensors improves after learning with the proposed training data. In Sect. 5, a method for further improving the prediction accuracy is detailed. The central concept of the method is learning with both the conventional training data and logarithmically represented training data. Section 6 gives conclusion. With our new method, the degradation of the prediction accuracy observed in our previous machine-learning approach [43] can be resolved without significantly increasing computational time. Our proposed method may be valid for high-precision real-time tracking of moving objects.

2. Basics of Position Sensing with Machine Learning

In this section, we briefly review the basic theory of position sensing with machine learning [43].

Conceptual images of a position sensing system are shown in Figs. 1 and 2. A target device equipped with a transmitter (TX) can freely move inside the cubic space. Magnetic fields are generated from the coil attached to the TX. Magnetic-field sensors (S1–S4) are placed at four corners of a room. The sensors can detect all components of magnetic-field vectors B_x, B_y, and B_z. Our purpose is to predict the position of the TX from the magnetic-field data detected by the sensors.

Evidently, the magnetic fields detected by the sensors depend on the position and direction of the TX coil. In equation form, the magnetic-field components detected by the k-th sensor ($k = 1, 2, 3, 4$) are expressed as

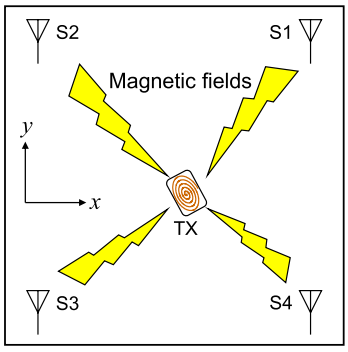


Fig. 1 Top view of a room equipped with four sensors (S1–S4) at the corners. The sensors detect magnetic fields generated by a TX, which is a target device.

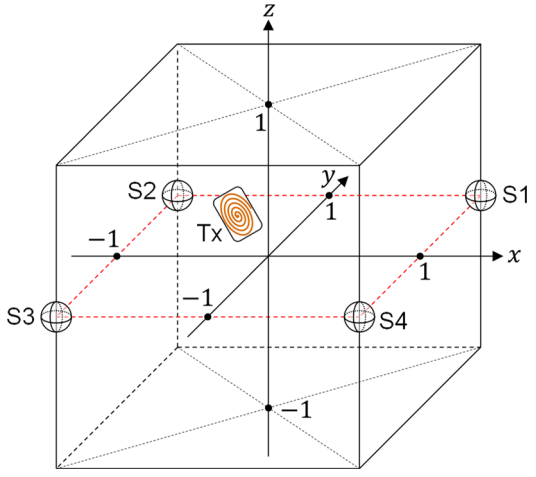


Fig. 2 Concept of the conditions for our simulation. A TX can move freely within a 2 m × 2 m × 2 m cubic space. Four sensors (S1, S2, S3, and S4) fixed at each corner of the x-y plane detect magnetic-field vectors generated by the TX. Based on the detected data, the TX position is predicted.

$$\mathbf{B}^{(k)} = \begin{pmatrix} B_x^{(k)}\left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \\ B_y^{(k)}\left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \\ B_z^{(k)}\left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \end{pmatrix}, \tag{1}$$

where $\left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}\right)$ and $\left(\theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right)$ represent the positions and angles of the TX, respectively [43]. Hereinafter, $\left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right)$ is referred to as "TX-state parameters" because they represent the physical state of the TX.

The magnetic fields generated at an arbitrary point (x, y, z) by a TX coil located at a coordinate origin can be written as [43]

$$\begin{aligned} &\begin{pmatrix} B_x\left(x, y, z, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \\ B_y\left(x, y, z, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \\ B_z\left(x, y, z, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \end{pmatrix} = \\ &\frac{\mu_0}{4\pi}\frac{m}{\left(x^2+y^2+z^2\right)^{5/2}} \times \left[\cos\theta^{(\mathrm{t})}\begin{pmatrix} 3xz \\ 3yz \\ 2z^2-x^2-y^2 \end{pmatrix}\right. \\ &\left. + \sin\theta^{(\mathrm{t})}\begin{pmatrix} 3xy\sin\varphi^{(\mathrm{t})} + \left(2x^2-y^2-z^2\right)\cos\varphi^{(\mathrm{t})} \\ 3xy\cos\varphi^{(\mathrm{t})} + \left(2y^2-x^2-z^2\right)\sin\varphi^{(\mathrm{t})} \\ 3z\left(x\cos\varphi^{(\mathrm{t})} + y\sin\varphi^{(\mathrm{t})}\right) \end{pmatrix}\right], \end{aligned} \tag{2}$$

where μ_0 and m are the vacuum permeability and dipole moment associated with the TX-coil and current, respectively. It is noteworthy that the magnetic-field components detected by the k-th sensor can be expressed in terms of the left-hand side of (2)

$$\begin{aligned} \mathbf{B}^{(k)} &= \begin{pmatrix} B_x^{(k)}\left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \\ B_y^{(k)}\left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \\ B_z^{(k)}\left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \end{pmatrix} \\ &= \begin{pmatrix} B_x\left(x^{(k)}-x^{(\mathrm{t})}, y^{(k)}-y^{(\mathrm{t})}, z^{(k)}-z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \\ B_y\left(x^{(k)}-x^{(\mathrm{t})}, y^{(k)}-y^{(\mathrm{t})}, z^{(k)}-z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \\ B_z\left(x^{(k)}-x^{(\mathrm{t})}, y^{(k)}-y^{(\mathrm{t})}, z^{(k)}-z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \end{pmatrix}, \end{aligned} \tag{3}$$

where $(x^{(k)}, y^{(k)}, z^{(k)})$ denotes the position of the k-th sensor. Finally, by substituting (2) into (3), we can obtain closed-form expressions to calculate the magnetic-field components detected by the sensors for the TX under arbitrary states. Because the closed-form expression is complicated, we omit them herein.

The field components detected by the sensors can be calculated easily using the closed-form expressions. This is a simple forward problem. However, what we must do is to calculate the TX-state parameters $\left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right)$ from the sensor data $\mathbf{B}^{(k)}$. As it is a nonlinear inverse problem, it is difficult to directly calculate the TX-state parameters.

One of the promising methods to solve the inverse problem is machine learning. Our purpose is to obtain the predictor function $\mathbf{P}$ which can predict the TX position from the sensor data $\mathbf{B}^{(k)}$. In equation form, the role of the predictor function is written as

$$\mathbf{x}^{(\mathrm{p})} = \mathbf{P}\left(\mathbf{B}^{(1)}, \mathbf{B}^{(2)}, \mathbf{B}^{(3)}, \mathbf{B}^{(4)}\right), \tag{4}$$

where $\mathbf{x}^{(\mathrm{p})}$ is the predicted TX-position vector. The superscript (p) means that $\mathbf{x}^{(\mathrm{p})}$ is a predicted quantity. Therefore, $\mathbf{x}^{(\mathrm{p})}$ is considered an approximation of the actual TX-position vector $\mathbf{x}^{(\mathrm{t})}$. In component representation, (4) can be written as

$$\begin{aligned} \begin{pmatrix} x^{(\mathrm{p})} \\ y^{(\mathrm{p})} \\ z^{(\mathrm{p})} \end{pmatrix} &= \mathbf{P}\left(B_x^{(1)}, B_y^{(1)}, B_z^{(1)}, \cdots, B_x^{(4)}, B_y^{(4)}, B_z^{(4)}\right) \\ &= \begin{pmatrix} P_x\left(B_x^{(1)}, B_y^{(1)}, B_z^{(1)}, \cdots, B_x^{(4)}, B_y^{(4)}, B_z^{(4)}\right) \\ P_y\left(B_x^{(1)}, B_y^{(1)}, B_z^{(1)}, \cdots, B_x^{(4)}, B_y^{(4)}, B_z^{(4)}\right) \\ P_z\left(B_x^{(1)}, B_y^{(1)}, B_z^{(1)}, \cdots, B_x^{(4)}, B_y^{(4)}, B_z^{(4)}\right) \end{pmatrix}. \end{aligned} \tag{5}$$

An advantage of using machine learning is that the predictor function can be generated from the training data. In this case, the training sample can be written as

$$\left(\mathbf{B}^{(1)}, \mathbf{B}^{(2)}, \mathbf{B}^{(3)}, \mathbf{B}^{(4)}\right) \longmapsto \mathbf{x}^{(\mathrm{t})}, \tag{6}$$

where the left- and right-hand sides are the input and desired output of the predictor function, respectively. Fairly reliable predictor functions can be generated by learning with sufficient numbers of training samples, which can be calculated easily using (2) and (3) for various TX-state parameters. This calculation is a simple forward problem. Unlike several applications of machine learning, training samples from the real world need not be gathered. In this sense, it is reasonable to apply machine learning to position sensing based on artificially-generated magnetic fields.

Because $\mathbf{B}^{(k)}$ is a function of the TX-state parameters, the predictor function can be regarded also as a function of those parameters. Therefore, we introduce a new predictor function $\widetilde{\mathbf{P}}$ as

$$\begin{aligned} \mathbf{x}^{(\mathrm{p})} &= \mathbf{P}\left(\mathbf{B}^{(1)}\left(\mathbf{x}^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right), \cdots, \mathbf{B}^{(4)}\left(\mathbf{x}^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right)\right) \\ &= \widetilde{\mathbf{P}}\left(\mathbf{x}^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right). \end{aligned} \tag{7}$$

The accuracy of the predictor function can be evaluated by the distance between $\mathbf{x}^{(\mathrm{t})}$ and $\mathbf{x}^{(\mathrm{p})}$. Subsequently, we introduce the error-distance function (EDF) as follows:

$$d\left(\mathbf{x}^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) \triangleq \left\|\mathbf{x}^{(\mathrm{t})} - \widetilde{\mathbf{P}}\left(\mathbf{x}^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right)\right\|. \tag{8}$$

The aforementioned EDF represents the prediction errors. Therefore, the values of d decrease with improvements in the predictor functions.

3. Logarithmic Representation of Training Data

In this section, we discuss the procedure to obtain training data that are more suitable for magnetic-field-based position sensing.

When $\theta^{(\mathrm{t})} = 0$, (2) can be simply written in the spherical coordinates as follows:

$$\begin{pmatrix} B_r(r,\theta,\varphi) \\ B_\theta(r,\theta,\varphi) \\ B_\varphi(r,\theta,\varphi) \end{pmatrix} = \frac{\mu_0}{4\pi}\frac{m}{r^3}\begin{pmatrix} 2\cos\theta \\ \sin\theta \\ 0 \end{pmatrix}. \tag{9}$$

Note that θ and φ are not TX angles but parameters of the spherical coordinate. Equation (9) shows that the amplitude of the detected signals is inversely proportional to the cube of r, which is the distance between the TX and the sensors. By considering that the amplitude of the electromagnetic waves is inversely proportional to r, it can be understood that the spatial-attenuation ratio of the magnetic fields is much sharper than that of the electromagnetic waves. Therefore, when the TX moves around the neighborhood of one of the sensors, the signal level detected by the sensor becomes significantly large and the level varies considerably. However, when the distances between the TX and sensors increase, the signal levels and their variations become substantially small. It is assumed that the difference in the behavior of the signal levels will affect smooth learning processes and degrade the prediction accuracy.

A feasible method to resolve the problem is to logarithmically represent magnetic-field levels in the training data. This is equivalent to expressing the signal levels in decibels. The differences in the signal behaviors mentioned above can be reduced by using the logarithmic representation. We denote the logarithmically represented field levels as follows:

$$\bar{B}_i^{(k)} \triangleq \log\left|B_i^{(k)}\right|, \tag{10}$$

where i can be x, y, or z. To demonstrate the effectiveness of using the logarithmic representation, B and $\bar{B}$ are plotted as a function of r in Figs. 3(a) and (b), respectively. It is confirmed that the extreme behavior of the signal level shown in Fig. 3(a) is reduced in Fig. 3(b).

Based on (10), a disadvantage of the logarithmic representation is that the positive/negative information of the field is inevitably lost. To compensate for the lost information, we introduce the sign parameters of the magnetic fields, as shown below:

$$S_i^{(k)} \triangleq \operatorname{sign}\left(B_i^{(k)}\right). \tag{11}$$

In the conventional method, training samples are expressed as in (6), which can be written in the component notation as follows:

$$\left(B_x^{(1)}, B_y^{(1)}, B_z^{(1)}, \cdots, B_x^{(4)}, B_y^{(4)}, B_z^{(4)}\right) \longmapsto \left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}\right). \tag{12}$$

It is noteworthy that the numbers of the input and desired output parameters are 12 and 3, respectively. We name the representation of (12) for a training sample as a "linear representation." A straightforward method to write training samples in the logarithmic representation is as follows:

$$\left(\bar{B}_x^{(1)}, \cdots, \bar{B}_z^{(4)}; S_x^{(1)}, \cdots, S_z^{(4)}\right) \longmapsto \left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}\right). \tag{13}$$

It is noteworthy that the numbers of input parameters are increased to 24. We name the abovementioned representation of a training sample as "naïve-log representation." It

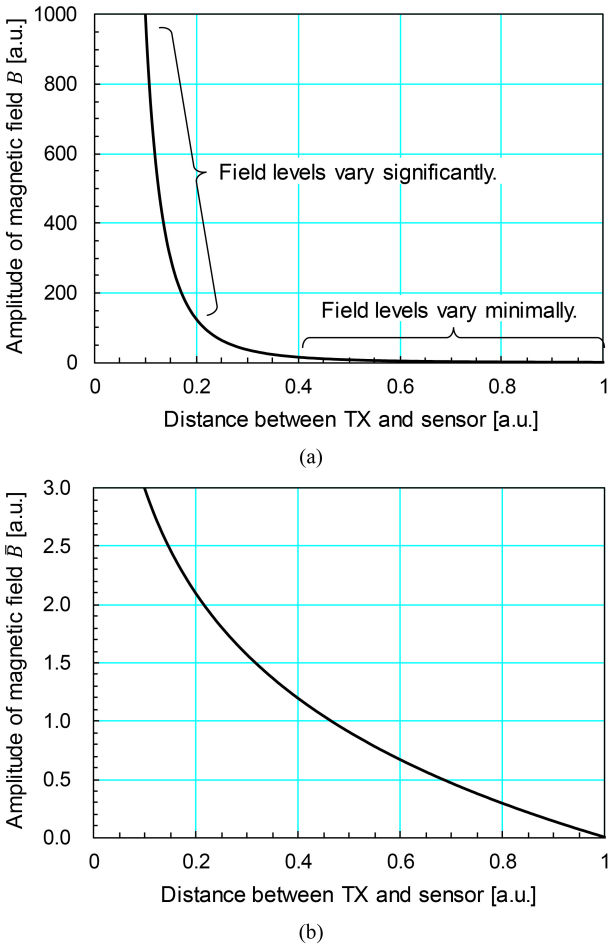


Fig. 3 Relation between magnetic-field levels and distances. (a) Linear representation. (b) Logarithmic representation.

results in an increase in the time required for both learning and prediction because of the increased numbers of the input parameters. In particular, prediction is a severe problem because it negates the advantage of using machine learning in position sensing, i.e., the computational speed of prediction. We denote the predictor function obtained from naïve-log representation by $\mathbf{L}$. With this predictor function, $\mathbf{x}^{(\mathrm{p})}$ is written as

$$\mathbf{x}^{(\mathrm{p})} = \mathbf{L}\left(\bar{B}_x^{(1)}, \cdots, \bar{B}_z^{(4)}; S_x^{(1)}, \cdots, S_z^{(4)}\right) \tag{14}$$

One of the key ideas for reducing the input parameters is encoding the information of $S_i^{(k)}$ on $\bar{B}_i^{(k)}$. The simplest method to achieve this is by multiplying $\bar{B}_i^{(k)}$ with $S_i^{(k)}$. In equation form, the transformation of the magnetic-field levels can be written as

$$\check{B}_i^{(k)} \triangleq \operatorname{sign}\left(B_i^{(k)}\right)\log\left|B_i^{(k)}\right|. \tag{15}$$

An example of the relation between $B_i^{(k)}$ and $\check{B}_i^{(k)}$ is plotted in Fig. 4(a). The possible maximum signal amplitude detected by the sensors is denoted by $|B|_{\max}$ and it is expressed as

$$|B|_{\max} \triangleq \max_{(\text{all training data})}\left\{\left|B_i^{(k)}\right|\right\}. \tag{16}$$

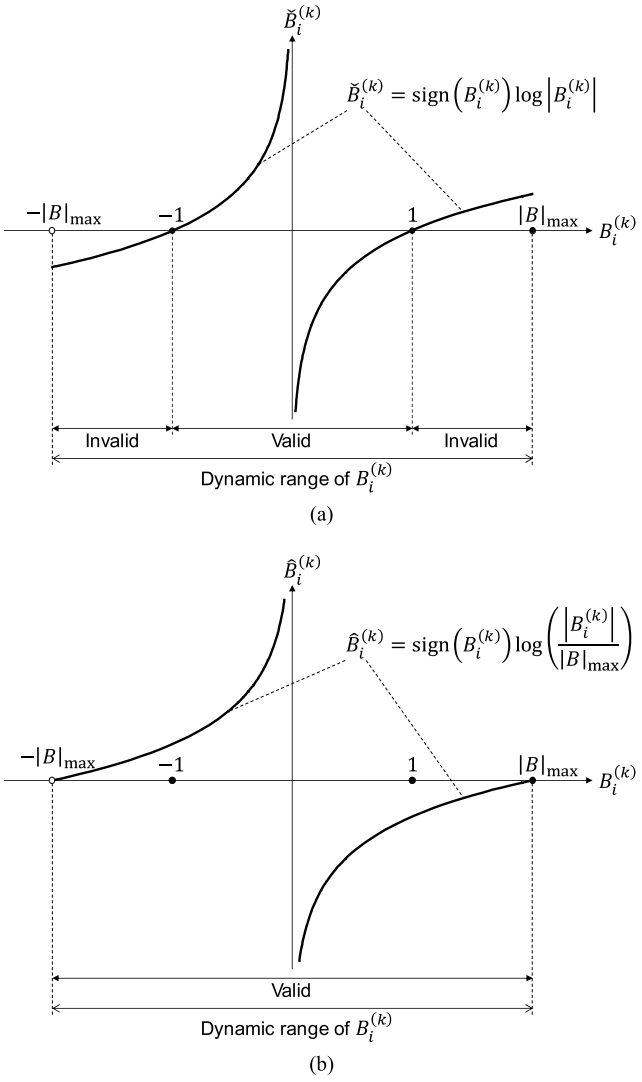


Fig. 4 Relation between the true value of the received signal $B_i^{(k)}$ and transformed values. (a) Relation between $B_i^{(k)}$ and $\check{B}_i^{(k)}$, which is defined in (15). In this case, one-to-one correspondence is not satisfied. (b) Relation between $B_i^{(k)}$ and $\hat{B}_i^{(k)}$, which is defined in (17). In this case, one-to-one correspondence is satisfied.

The dynamic range of the received signal level becomes $2|B|_{\max}$ by considering the positive/negative information of the magnetic fields, as shown in Fig. 4(a). The magnetic-field signals detected by the receivers are within the dynamic range. In this case, the transformation $B_i^{(k)} \longmapsto \check{B}_i^{(k)}$ does not satisfy a one-to-one correspondence. To use the transformed data for the training samples, the transformation must satisfy a one-to-one correspondence.

To avoid the difficulty, we introduce the normalized signed-log (NSL) transformation, expressed as follows:

$$\hat{B}_i^{(k)} \triangleq \mathrm{sign}\left(B_i^{(k)}\right)\log\left(\frac{\left|B_i^{(k)}\right|}{|B|_{\max}}\right). \tag{17}$$

It is noteworthy that the argument of the logarithm is normalized. In other words, the argument is restricted within the following range:

$$0 < \frac{\left|B_i^{(k)}\right|}{|B|_{\max}} \le 1. \tag{18}$$

The relation between $B_i^{(k)}$ and $\hat{B}_i^{(k)}$ is plotted in Fig. 4(b). The transformation $B_i^{(k)} \longmapsto \hat{B}_i^{(k)}$ holds a one-to-one correspondence owing to the normalization. Moreover, the positive/negative information of $B_i^{(k)}$ is encoded on $\hat{B}_i^{(k)}$. Therefore, with the NSL transformation provided in (17), the number of input parameters in the training samples is maintained as 12, as shown below:

$$\left(\hat{B}_x^{(1)}, \hat{B}_y^{(1)}, \hat{B}_z^{(1)}, \cdots, \hat{B}_x^{(4)}, \hat{B}_y^{(4)}, \hat{B}_z^{(4)}\right) \longmapsto \left(x^{(\mathrm{t})}, y^{(\mathrm{t})}, z^{(\mathrm{t})}\right). \tag{19}$$

We name this representation as "NSL representation." Equations (4) and (7) are written as

$$\begin{aligned}\mathbf{x}^{(\mathrm{p})} &= \mathbf{Q}\left(\hat{B}_x^{(1)}, \hat{B}_y^{(1)}, \hat{B}_z^{(1)}, \cdots, \hat{B}_x^{(4)}, \hat{B}_y^{(4)}, \hat{B}_z^{(4)}\right)\\ &= \mathbf{Q}\left(\hat{\mathbf{B}}^{(1)}, \hat{\mathbf{B}}^{(2)}, \hat{\mathbf{B}}^{(3)}, \hat{\mathbf{B}}^{(4)}\right)\\ &= \mathbf{Q}\left(\hat{\mathbf{B}}^{(1)}\left(\mathbf{x}^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right), \cdots, \hat{\mathbf{B}}^{(4)}\left(\mathbf{x}^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right)\right)\\ &= \widetilde{\mathbf{Q}}\left(\mathbf{x}^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right),\end{aligned} \tag{20}$$

where $\mathbf{Q}$ and $\widetilde{\mathbf{Q}}$ are predictor functions obtained with the training data in NSL representation.

4. Performance Analysis of Machine Learning with Various Training-Data Representations

We numerically investigated the effectiveness of using training data in logarithmic representations. The scenarios and conditions for our simulations are illustrated in Figs. 2 and 5. For our simulation, the dimensions of the target space were set to 2 m × 2 m × 2 m. However, the dimensions were not essential because the scaling law holds in magnetic-field-based position sensing. Therefore, all results obtained for the 2 m × 2 m × 2 m cubic space are theoretically applicable to cubic spaces of arbitrary dimensions [43].

To obtain training samples such as those expressed in (12), (13), and (19), we calculated $\mathbf{B}^{(k)}$, i.e., the signal levels detected by the k-th sensor, for a number of different TX-state parameters $\left(\mathbf{x}^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right)$. The TX positions used to calculate the training samples are indicated as black dots in Figs. 5(a) and (b), which show the x-y and x-z planes of the 2 m × 2 m × 2 m cubic space, respectively. The units for the numerical values shown in Fig. 5 is meter. Therefore, the distance between the two nearest dots is 10 cm ($\delta x = \delta y = \delta z = 0.1$ m). In this case, the number of TX positions for the training samples becomes $19^3 = 6,859$. Because $\mathbf{B}^{(k)}$ depends on the TX angles, the training samples must be obtained for various different TX angles. Therefore, we should calculate $\mathbf{B}^{(k)}$ for a number of TX-angle sets $\left(\theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right)$ for each TX position. In our calculation, $\delta\theta^{(\mathrm{t})} = \delta\varphi^{(\mathrm{t})} = 11.25°$ was adopted. Therefore, the number of angle sets was 482. Hence, the total number of TX states used for the training data reached 6,859 × 482 = 3,306,038.

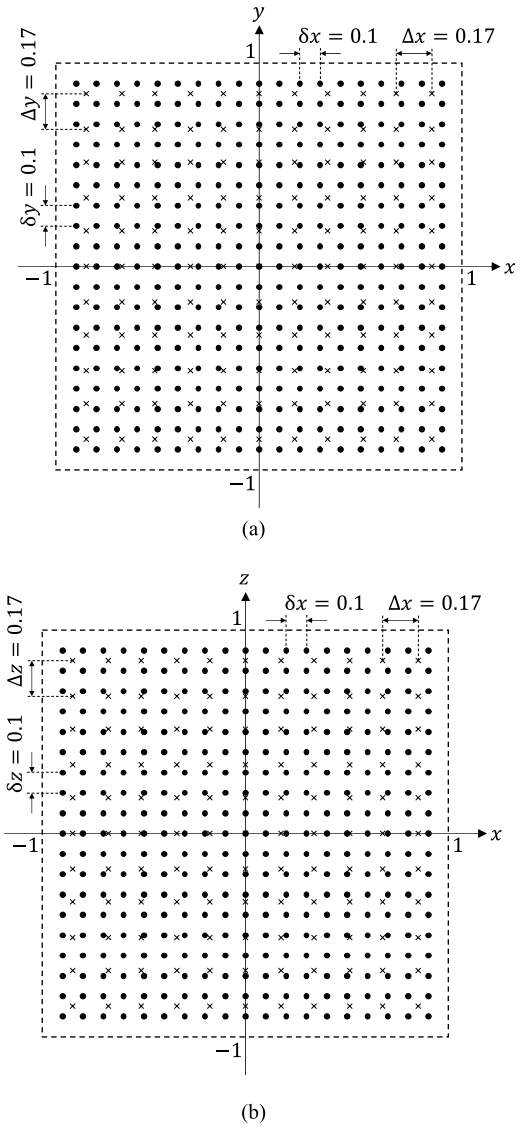


Fig. 5 (a) x-y plane and (b) x-z plane sliced in the 2 m × 2 m × 2 m cubic space. TX positions used for calculating training data are indicated with black dots. The distance between the dots is 10 cm. TX positions used for calculating EDF are indicated with crosses. The distance between the crosses is 17 cm. The positions of the dots and crosses do not coincide except for the coordinate origin.

We used Wolfram Mathematica 12 to execute machine learning. Mathematica includes highly-automated functions for machine learning. We used the "Predict" command to generate the predictor functions **P**, **L**, and **Q**. The command is suitable for regression analysis. These predictor functions can be generated automatically using training samples. Additionally, it is possible with Mathematica to select an appropriate machine-learning algorithm. Since "k-nearest neighbor algorithm" was chosen in the previous study [43], we selected the algorithm also in this study for demonstrating superiority of our proposed method under conditions similar to [43]. The k-nearest neighbor algorithm is one of typical machine-learning algorithms [27], [44]. Furthermore, the algorithm is often used in indoor positioning with radio waves [5]. It is also suggested that, with the k-nearest neighbor algorithm, fairly good prediction accuracy is obtainable without spending vast amount of training time for magnetic-field-based positioning [43]. In the k-nearest neighbor algorithm of Mathematica, L2-norm is used as an error function. It is generally required for executing the k-nearest neighbor algorithm to specify the value of k. In this study, however, we did not specify k because Mathematica can automatically find appropriate k by using an original method. We used a standard laptop equipped with Core i7-9750H and 16-GB RAM for executing the machine learning.

The crosses shown in Figs. 5(a) and (b) denote the TX positions used for evaluating the prediction accuracy of the predictor functions generated by machine learning. To verify the generalization performance of machine learning, the crosses were placed at positions different from those of the dots, except for the coordinate origin.

To investigate the validity of using training data in logarithmic representations, we numerically calculated the EDF and plotted it for three TX conditions that were not used for the training. The results are presented in Fig. 6. The three images in the first column were obtained with the training data in linear representation. In other words, they were identical to the results obtained in [43]. As explained in Sect. 2, it was confirmed that the prediction accuracy deteriorated in the vicinity of the sensors located at four corners. The three images in the second column were obtained with training data in naïve-log representation. As expected, the prediction accuracy in the vicinity of the corners improved. The three images in the third column were obtained with training data in NSL representation. We confirmed that these images were almost identical to those of the second column. This means that the two different training datasets, which are in naïve-log and NSL representations, yielded almost identical learning results.

To quantitatively evaluate the prediction performances obtained with the different representations of training data, we analyzed the statistical distribution of the EDF values for the TX positions marked with crosses in Fig. 5. The distance between the two nearest crosses was 17 cm ($\Delta x = \Delta y = \Delta z = 0.17$ m). The angular variation of the TX was considered for each cross point ($\Delta\theta^{(t)} = \Delta\varphi^{(t)} = 30°$). Hence, the number of TX states used for the statistical analysis corresponded to 82,522. It is noteworthy that the TX-state parameters $\left(\mathbf{x}^{(t)}, \theta^{(t)}, \varphi^{(t)}\right)$ used for calculating the training data were not included in the 82,522 states for validation. Therefore, we can evaluate the generalization performance of machine learning through a statistical analysis using the 82,522 states of the TX.

The statistical distributions of the EDF values are shown in Fig. 7. It was confirmed that the prediction accuracy improved by using training data in logarithmic representations. Furthermore, we confirmed that the performances obtained with naïve-log and NSL representations were almost identical.

Table 1 summarizes the performances of predictor

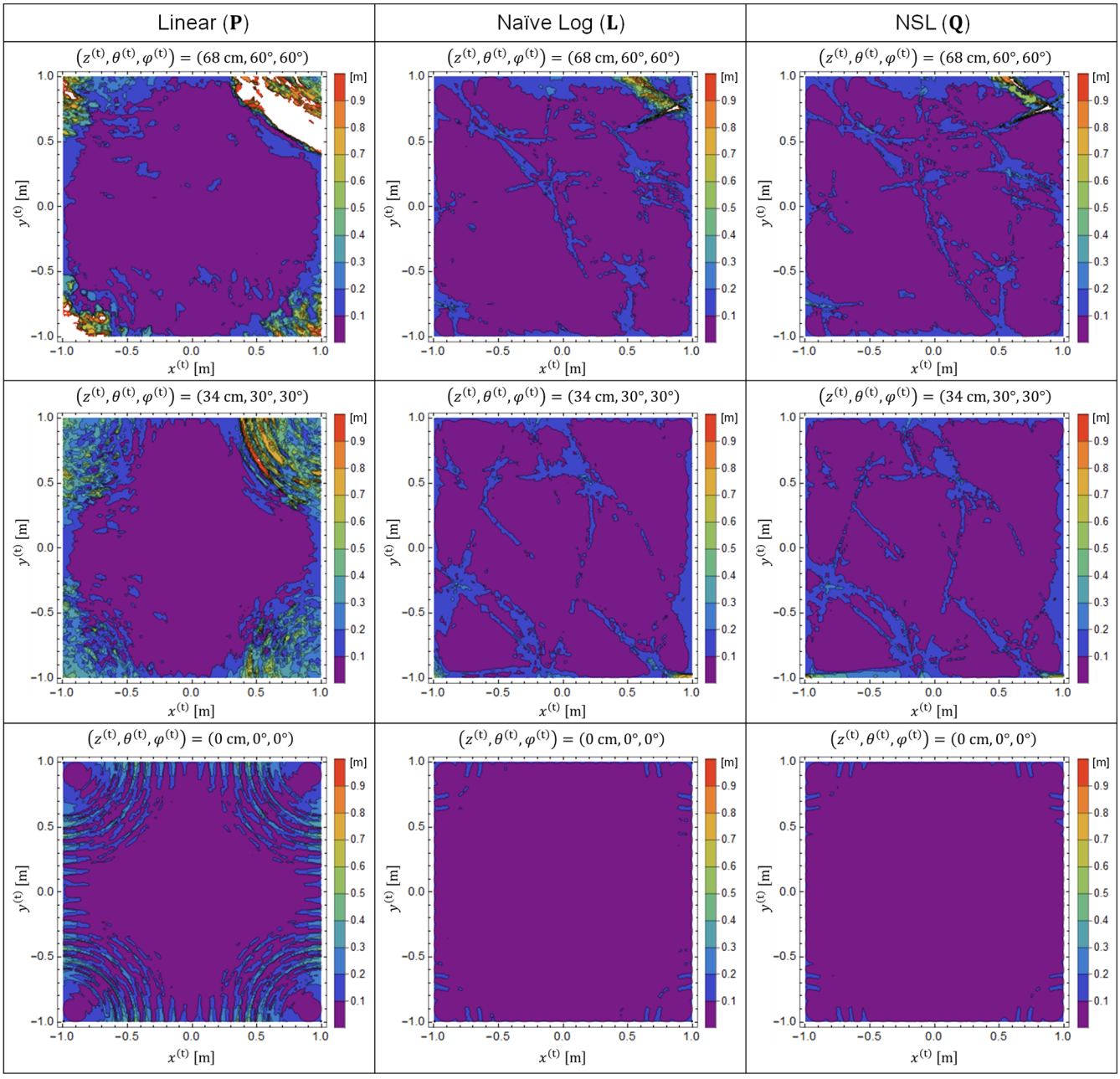


Fig. 6 EDF patterns plotted in several cross sections parallel to the x-y plane. EDF values in the vicinity of the corners are reduced when using predictor functions **L** and **Q**, which are generated from logarithmically transformed training data. Furthermore, the EDF patterns obtained using **L** and **Q** are almost identical.

functions obtained with the three different representations of training data. Prob ($d < 10\,\text{cm}$) means the probability that d becomes less than 10 cm, which is shown in Fig. 7. d_{av} is the average EDF values calculated from the results in Fig. 7. T_{learn} and T_{pred} are computational times required for learning and prediction, respectively. We can confirm several important results from Table 1. First, the prediction accuracy improved by using the logarithmically represented training data. Second, the two different logarithmic representations, i.e., the naïve-log and NSL representations, indicated almost identical accuracies in prediction. Third, T_{pred} reduced significantly with NSL representation compared with the case involving the naïve-log representation. Although T_{learn} increased with the naïve-log and NSL representations, it barely affects practical applications. In summary, the NSL representation can improve the prediction accuracy without increasing T_{pred} significantly.

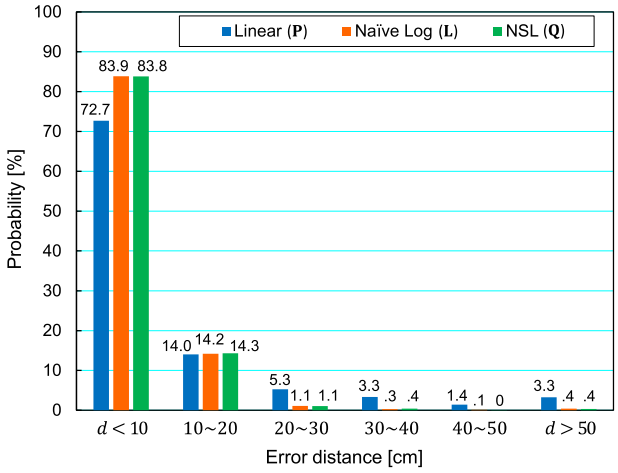


Fig. 7 Statistical distributions of EDF values obtained with three different representations of training data. Prediction accuracy improves with naïve-log and NSL representations. Performances obtained with naïve-log and NSL representations are almost identical.

Table 1 Performances of three different predictor functions.

	Prob($d < 10$ cm)	d_{av}	T_{learn}	T_{pred}
Linear (**P**)	72.7%	11.0 cm	130 s	3.5 ms
Naïve Log (**L**)	83.9%	6.5 cm	1472 s	322.0 ms
NSL (**Q**)	83.8%	6.6 cm	869 s	4.2 ms

5. Use of a Hybrid Predictor Function

As shown in Fig. 6, the prediction accuracy improved with logarithmic representations, particularly in the vicinity of the four corners. However, the accuracy obtained with the logarithmic representations degraded under certain conditions. For example, see EDF patterns in the first row of Fig. 6. These patterns were obtained for the condition $\left(z^{(\mathrm{t})}, \theta^{(\mathrm{t})}, \varphi^{(\mathrm{t})}\right) = (68\,\mathrm{cm}, 60^\circ, 60^\circ)$. In this case, the prediction accuracy around the center of the area decreased when logarithmic representations were adopted. Therefore, the logarithmic representations did not always perform well. In other words, linear representations are often better when the TX is around the center of the cubic space.

By considering these characteristics, it is natural to come up with an idea that we should appropriately select better predictor function between **P** and **Q**. However, the key problem is the selection of better predictor function between **P** and **Q** based on the signals detected by the sensors. A reasonable strategy is to select **Q** only when the TX appears in the vicinity of one of the four sensors. When the TX approaches the k-th sensor, the absolute value of the magnetic field at k-th sensor position is inevitably increased. Hence, we defined a hybrid predictor function **H**, expressed as follows:

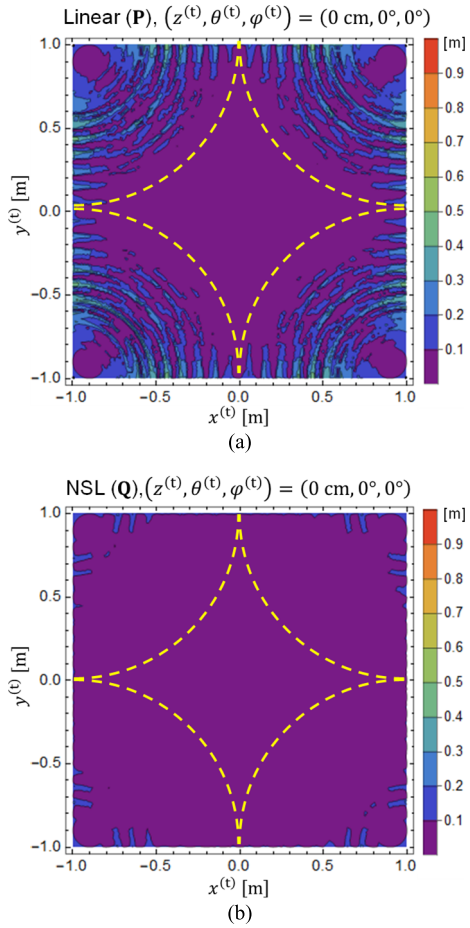


Fig. 8 EDF patterns obtained using (a) **P** and (b) **Q**, which are excerpts of Fig. 6. **Q** shows a better performance outside the closed curve shown as the dashed-yellow line. **P** shows a good performance inside the closed curve. Therefore, **Q** need not be used inside the closed curve.

$$\mathbf{H} \triangleq \begin{cases} \mathbf{Q}; \text{ if } \max\left\{\left|\mathbf{B}^{(1)}\right|, \left|\mathbf{B}^{(2)}\right|, \left|\mathbf{B}^{(3)}\right|, \left|\mathbf{B}^{(4)}\right|\right\} > B_{\mathrm{th}} \\ \mathbf{P}; \text{ else} \end{cases}, \tag{21}$$

where B_{th} is threshold-signal amplitude determined in advance. By introducing the hybrid predictor function **H** and substituting the appropriate value into B_{th}, **Q** is applied only when the TX appears in the vicinity of one of the sensors.

To determine the appropriate value of B_{th}, we observe the EDF patterns obtained using **P** and **Q** in Fig. 8. It is clear that **Q** should be used when the TX appears outside the closed curve shown as a dashed-yellow line. However, **Q** need not be used when the TX appears inside the closed curve. Therefore, a reasonably optimal B_{th} can be calculated as follows:

$$B_{\mathrm{th}}^{\mathrm{opt}} = \max\left\{\left|\mathbf{B}^{(1)}\left(\mathbf{x}_{\mathrm{cc}}^{(\mathrm{t})}, 0, 0\right)\right|, \cdots, \left|\mathbf{B}^{(4)}\left(\mathbf{x}_{\mathrm{cc}}^{(\mathrm{t})}, 0, 0\right)\right|\right\}, \tag{22}$$

where $\mathbf{x}_{\mathrm{cc}}^{(\mathrm{t})}$ means an arbitrary TX position on the closed curve. By considering the magnetic-field patterns described in (2), $B_{\mathrm{th}}^{\mathrm{opt}}$ can be written as follows:

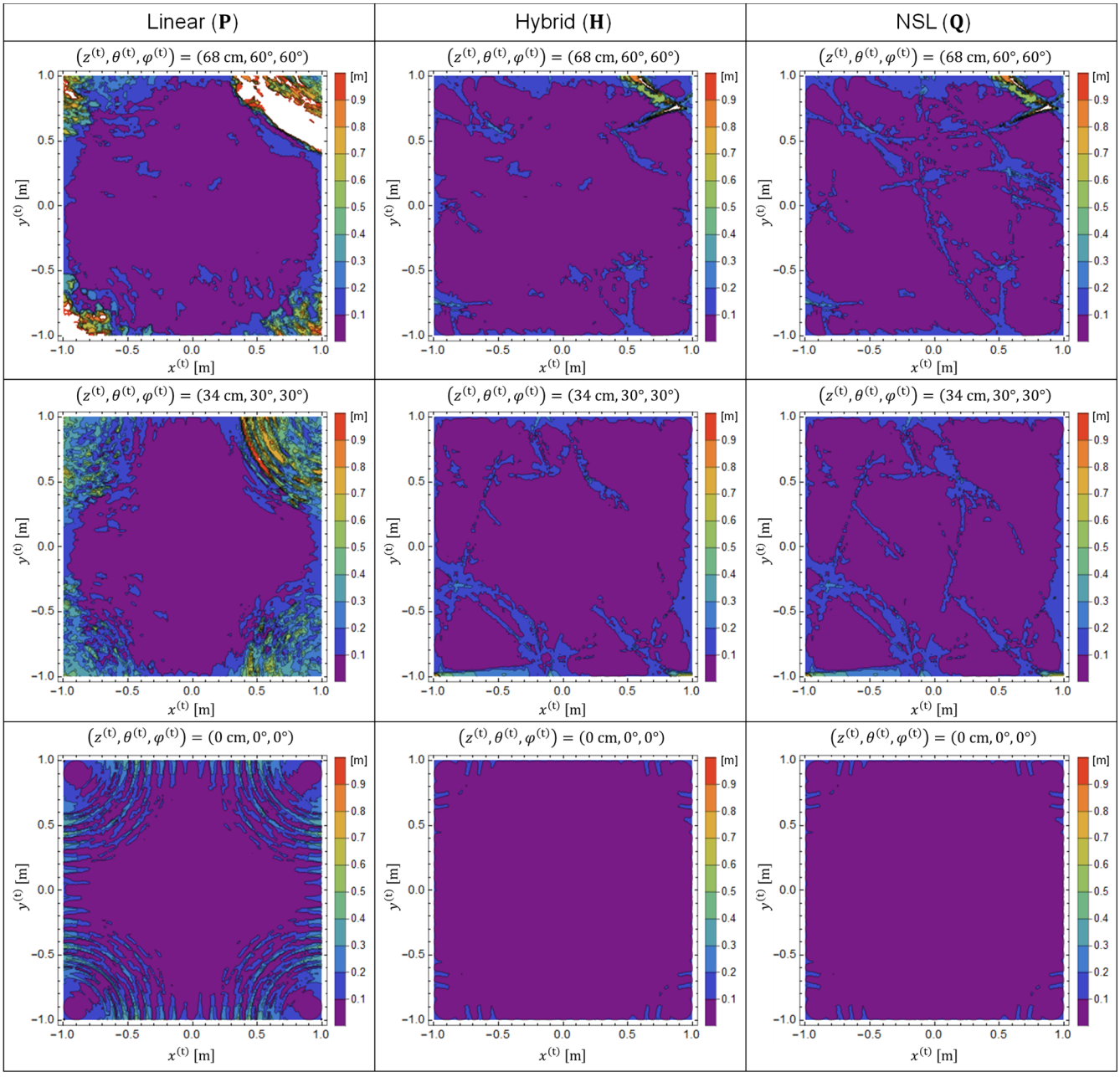


Fig. 9 EDF patterns plotted in several cross sections parallel to the x-y plane. The set of three images in the first, second, and third columns are obtained with the predictor functions **P**, **H**, and **Q**, respectively. Regarding four corners of the area, EDF patterns obtained with **H** is similar to that obtained with **Q**. Regarding the area center, the EDF patterns obtained with **H** is similar to that obtained with **P**. It is observed that the prediction accuracy improves when using the hybrid predictor function **H**.

$$B_{\text{th}}^{\text{opt}} = -B_z(1,0,0,0,0) = \frac{\mu_0}{4\pi}\left.\frac{m}{x^3}\right|_{x=1} = \frac{\mu_0 m}{4\pi}. \tag{23}$$

Because the EDF patterns in Fig. 8 were obtained for a specific condition, i.e., $\left(z^{(t)}, \theta^{(t)}, \varphi^{(t)}\right) = (0\,\text{cm}, 0°, 0°)$, the $B_{\text{th}}^{\text{opt}}$ obtained using the process above may not be the best value for other conditions. However, the hybrid predictor function **H** with the $B_{\text{th}}^{\text{opt}}$ is expected to demonstrate a better performance than the sole predictor function **Q**.

To demonstrate the effectiveness of **H**, we plotted EDF patterns obtained using **P**, **H**, and **Q** as shown in Fig. 9. The EDF patterns in the first and third columns were obtained using **P** and **Q**, respectively. The second column shows the EDF patterns obtained using **H**, which is the hybrid predictor function. It was observed that **H** was relatively effective because the EDF patterns obtained with **H** was better than those obtained using **P** and **Q**. As expected, the center of the area showed that the EDF patterns in the second column was identical to those in the first column. Similarly, at the vicinity of the four corners, the patterns of the second column

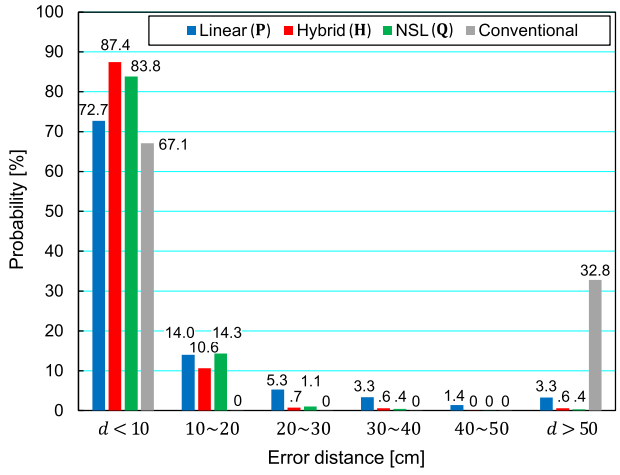


Fig. 10 Statistical distributions of EDF values obtained with three different predictor functions, i.e., **P**, **H**, and **Q**, and with a conventional optimization method. The best prediction accuracy is obtained using **H**, which is a hybrid predictor function composed of **P** and **Q**.

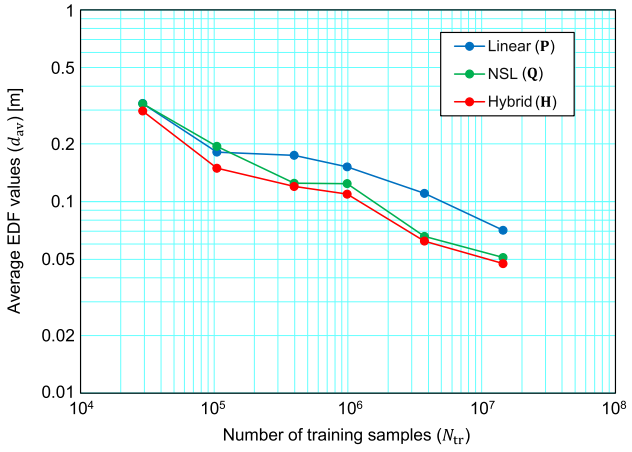


Fig. 11 Relation between d_{av} and the number of training samples for three different predictor functions **P**, **Q**, and **H**. Since d_{av} is monotonically decreasing, the situation of overfitting has not yet been occurred.

were identical to those of the third column.

To quantitatively compare the performance of **H** with those of **P** and **Q**, we again analyzed the statistical distribution of the EDF values. The condition of the analysis was similar to that for Fig. 7. The results are shown in Fig. 10. For comparison, a result obtained with the conventional optimization method is also added to Fig. 10. Details about the conventional method are given in [43]. It was confirmed that the best prediction accuracy was obtained using the hybrid predictor function **H**.

Furthermore, we investigated relations between d_{av} and the number of training samples (N_{tr}) for the predictor functions **P**, **Q**, and **H**. The calculated results are shown in Fig. 11. It is observed that d_{av} is monotonically decreasing. All calculated results given in this paper (Figs. 6–10, and Tables 1 and 2) were obtained for N_{tr} = 3,306,038. This means that a situation of overfitting has not yet been occurred in the calculated results. Moreover, it is expected that the estimation error is further reduced by increasing N_{tr} though larger computational resources are required.

Table 2 Summary of performances obtained with all predictor functions investigated in this study and with a conventional optimization method.

	Prob($d < 10$ cm)	d_{av}	T_{learn}	T_{pred}
Linear (**P**)	72.7%	11.0 cm	130 s	3.5 ms
Naïve Log (**L**)	83.9%	6.5 cm	1472 s	322.0 ms
NSL (**Q**)	83.8%	6.6 cm	869 s	4.2 ms
Hybrid (**H**)	87.4%	6.2 cm	999 s	4.2 ms
Combined (**C**)	88.5%	5.9 cm	2582 s	209.4 ms
Conventional	67.1%	54.0 cm	n.a.	162.1 ms

It was demonstrated that the estimation accuracy is improved by using **H**, which is defined in (21). A key idea of the hybrid predictor function is to appropriately select **P** or **Q**. However, one may conceive a simpler method to generate a better predictor function. The method is to embed both $B_i^{(k)}$ and $\hat{B}_i^{(k)}$ in training samples. In equation form, the training samples are written as

$$\left(B_x^{(1)}, \cdots, B_z^{(4)}; \hat{B}_x^{(1)}, \cdots, \hat{B}_z^{(4)}\right) \longmapsto \left(x^{(t)}, y^{(t)}, z^{(t)}\right). \quad (24)$$

Let us call the abovementioned representation of a training sample "combined representation." The predictor function **C** generated from the combined representation can be formally written as

$$\mathbf{x}^{(p)} = \mathbf{C}\left(B_x^{(1)}, \cdots, B_z^{(4)}; \hat{B}_x^{(1)}, \cdots, \hat{B}_z^{(4)}\right). \quad (25)$$

It is expected that **C** shows equal or better prediction accuracy in comparison with **H**. However, **C** has a serious drawback that the number of input parameters is increased from 12 to 24. This situation is similar to **L**, which is the predictor function generated from naïve-log representation (13). As shown in Table 1, T_{pred} obtained with **L** becomes much longer than others. This results from the fact that the number of input parameters of **L** is increased from 12 to 24. Therefore, we are concerned that T_{pred} becomes very long also with **C**. To clarify the concern, we quantitatively investigated performances of **C**.

The performances obtained with all predictor functions investigated in this study and with the conventional optimization method [43] are summarized in Table 2. Regarding the prediction accuracy, both **C** and **H** demonstrated better performances than others and **C** is even superior to **H** slightly. As we are concerned, however, T_{pred} obtained with **C** becomes much longer than that of **H**. Therefore, from the overall viewpoint, the best predictor function is considered to be **H**.

Because **H** is composed of **P** and **Q**, the following relation holds:

$$T_{learn}(\mathbf{H}) = T_{learn}(\mathbf{P}) + T_{learn}(\mathbf{Q}). \quad (26)$$

Furthermore, **H** and **Q** indicated similar results for T_{pred}.

This means that the computational time required to assess between **P** and **Q** is negligible. It is again confirmed that the prediction accuracy is improved by **Q** without introducing high computational costs. The prediction accuracy can be further improved by the hybrid use of **P** and **Q**.

6. Conclusion

Methods of machine learning suitable for magnetic-field-based position sensing have been investigated. The proposed methods are based on three original ideas.

First, magnetic-field information used for training data should be represented logarithmically. It was demonstrated that the predictor function **L** obtained with logarithmically represented training data exhibited better prediction accuracy than the predictor function **P** obtained with conventional linear data. However, the naïve usage of the logarithmic training data resulted in considerable increase in the computational time required for predicting the positions of a target device.

The second original idea to overcome the difficulty is encoding the sign information of the magnetic fields on the logarithmically represented training data. The predictor function **Q** obtained with the training data in NSL representation exhibited both accurate and rapid predictions of the target-device positions. Degraded prediction accuracies in the vicinity of sensors, which was a major problem in the previous machine-leaning approach [43], was successfully restored using **Q**.

The third original idea to further improve the prediction accuracy is introducing a hybrid predictor function composed of **P** and **Q**. We investigated a simple method to select a better predictor function between **P** and **Q** based on signals detected by multiple sensors. The successful hybrid usage of **P** and **Q** was demonstrated.

With the newly proposed method, the probability that the error of the predicted device positions was within 10 cm in a 2 m × 2 m × 2 m cubic space was improved from 73% to 87%, and the average error of the predicted position decreased from 11 cm to 6 cm. The computational time required to predict the target position was approximately 4 ms using a standard laptop. The high-speed nature of the proposed prediction method is suitable for the real-time tracking of moving objects and people.

The performance of the proposed method depends on algorithms used for learning. Machine-learning technologies are making progress every day. Although we adopted the *k*-nearest neighbor algorithm in this study, the performance of magnetic-field-based position sensing may be enhanced by using other algorithms. Actually, it is suggested that the prediction accuracy is further increased with the neural-network algorithm [45]. It is desired to find the best algorithms for magnetic-field-based position sensing in terms of both prediction accuracy and computational time.

References

[1] R. Mautz, "Indoor positioning technologies," Habilitation thesis, ETH-Zürich, Zürich, Switzerland, 2012. DOI: 10.3929/ethz-a-007313554

[2] V. Pasku, A.D. Angelis, G.D. Angelis, D.D. Arumugam, M. Dionigi, P. Carbone, A. Moschitta, and D.S. Ricketts, "Magnetic field-based positioning system," IEEE Commun. Surveys Tuts., vol.19, no.3, pp.2003–2017, 3rd Quart., 2017.

[3] P. Davidson and R. Piché, "A survey of selected indoor positioning methods for smartphones," IEEE Commun. Surveys Tuts., vol.19, no.2, pp.1347–1370, 2nd Quart., 2017.

[4] X. Guo, N. Ansari, F. Hu, Y. Shao, N.R. Elikplim, and L. Li, "A survey on fusion-based indoor positioning," IEEE Commun. Surveys Tuts., vol.22, no.1, pp.566–594, 1st Quart., 2020.

[5] A. Nessa, B. Adhikari, F. Hussain, and X.N. Fernando, "A survey of machine learning for indoor positioning," IEEE Access, vol.8, pp.214945–214965, Dec. 2020.

[6] F.H. Raab, E.B. Blood, T.O. Steiner, and H.R. Jones, "Magnetic position and orientation tracking system," IEEE Trans. Aerosp. Electron. Syst., vol.AES-15, no.5, pp.709–718, Sept. 1979.

[7] F.H. Raab, "Quasi-static magnetic-field technique for determining position and orientation," IEEE Trans. Geosci. Remote Sens., vol.GE-19, no.4, pp.235–243, Oct. 1981.

[8] J. Blankenbach and A. Norrdine, "Position estimation using artificial generated magnetic fields," Proc. 2010 Int. Conf. Indoor Positioning Indoor Navigat., Zurich, Switzerland, pp.1–5, Sept. 2010.

[9] A. Sheinker, B. Ginzburg, N. Salomonski, L. Frumkis, and B.-Z. Kaplan, "Localization in 3-D using beacons of low frequency magnetic field," IEEE Trans. Instrum. Meas., vol.62, no.12, pp.3194–3201, Dec. 2013.

[10] A. Sheinker, B. Ginzburg, N. Salomonski, and A. Engel, "Localization of a mobile platform equipped with a rotating magnetic dipole source," IEEE Trans. Instrum. Meas., vol.68, no.1, pp.116–128, Jan. 2019.

[11] D.D. Arumugam, J.D. Griffin, D.D. Stancil, and D.S. Ricketts, "Three-dimensional position and orientation measurements using magneto-quasistatic fields and complex image theory," IEEE Antennas Propag. Mag., vol.56, no.1, pp.160–173, Feb. 2014.

[12] G. De Angelis, V. Pasku, A. De Angelis, M. Dionigi, M. Mongiardo, A. Moschitta, and P. Carbone, "An indoor ac magnetic positioning system," IEEE Trans. Instrum. Meas., vol.64, no.5, pp.1275–1283, May 2015.

[13] V. Pasku, A. De Angelis, M. Dionigi, G. De Angelis, A. Moschitta, and P. Carbone, "A positioning system based on low-frequency magnetic fields," IEEE Trans. Ind. Electron., vol.63, no.4, pp.2457–2468, April 2016.

[14] N. Ayuso, J.A. Cuchí, F. Lera, and J.L. Villarroel, "Accurately locating a vertical magnetic dipole buried in a conducting earth," IEEE Trans. Geosci. Remote Sens., vol.48, no.10, pp.3676–3685, Oct. 2010.

[15] T.E. Abrudan, Z. Xiao, A. Markham, and N. Trigoni, "Underground incrementally deployed magneto-inductive 3-D positioning network," IEEE Trans. Geosci. Remote Sens., vol.54, no.8., pp.4376–4391, Aug. 2016.

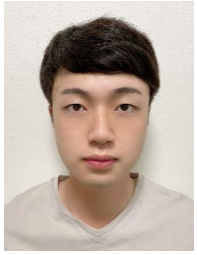
[16] R. Psiuk, A. Müller, T. Dräger, I. Ibrahim, H. Brauer, H. Töpfer, and A. Heuberger, "Simultaneous 2D localization of multiple coils in an LF magnetic field using orthogonal codes," Proc. IEEE Sensors 2017, Glasgow, Scotland, UK, Oct.–Nov. 2017. DOI: 10.1109/ICSENS.2017.8234183

[17] N. Zhang, H. Liu, S. Chen, M. Li, and Y. Jiang, "Indoor localization scheme for fire rescue based on super low frequency quasi-static field," Proc. 2nd IEEE Adv. Inf. Manage. Commun. Electron. Automat. Control Conf. (IMCEC 2018), Xi'an, China, pp.1095–1099, May 2018. DOI: 10.1109/IMCEC.2018.8469331

[18] J. Liu, G. Wang, L. Li, and H. Xia, "A magnetic beacon navigation

and positioning technology not relying on GPS," Proc. 8th Int. Conf. Instrum. Meas. Comput. Commun. Control (IMCCC 2018), Harbin, China, pp.861–866, July 2018. DOI: 10.1109/IMCCC.2018.00183
[19] M. Ralchenko, M. Roper, and P. Kwasniok, "Positioning pedestrians and cyclists in urban intersections using VLF magnetic fields," Proc. 2018 88th Vehicular Technol. Conf. (VTC-Fall), Chicago, IL, USA, Aug. 2018. DOI: 10.1109/VTCFall.2018.8690873
[20] M. Ralchenko and M. Roper, "VLF magnetic positioning in multistory parking garages," Proc. 9th Int. Conf. Indoor Posit. Indoor Navigat. (IPIN 2018), Nantes, France, Sept. 2018. DOI: 10.1109/IPIN.2018.8533786
[21] X. Qi, L. Chen, K. An, J. Wang, C. Ma, B. Zhang, K. Xu, H. Li, D. Ye, J. Huangfu, C. Li, and L. Ran, "Wireless indoor positioning with vertically uniform alternating magnetic fields," IEEE Trans. Instrum. Meas., vol.67, no.11, pp.2733–2755, Nov. 2018.
[22] J. Song, D. Sun, Z. Zhao, H. Zhou, H. Xia, and G. Ma, "Localization of low frequency magnetic beacons based on genetic algorithm," Proc. 39th Chin. Control. Conf., Shenyang, China, pp.3239–3244, July 2020.
[23] O.J.M. Scholle, E.A. Lomonova, and N.H. Vrijsen, "Analysis of the accuracy of a magnetic field-based positioning system including the environment of a parking vehicle," Proc. 2020 15th Int. Conf. Ecolog. Veh. Renew. Energ. (EVER), Monte-Carlo, Monaco, Sept. 2020. DOI: 10.1109/EVER48776.2020.9243121
[24] Y. Zheng, Q. Li, X. Wang, C. Wang, L. Wu, and X. Li, "Advanced positioning system for harsh environments using time-varying magnetic field," IEEE Trans. Magn., vol.57, no.6, art. no.4001412, June 2020.
[25] A. Sasaki, O. Ouellette, M.B.-Marchand, A. Hirata, and H. Morimura, "Analysis and experimental study of magnetic-field amplification by a double coil," IEEE Trans. Ind. Electron., vol.64, no.4, pp.3216–3226, April 2017.
[26] A. Sasaki, "Double-solenoid ferrite coils for magnetic-field amplification in LF and MF bands," IEEE Trans. Magn., vol.55, no.12, art. no.4004209, Dec. 2019.
[27] C.M. Bishop, Pattern Recognition and Machine Learning, Springer-Verlag, New York, NY, USA, 2006.
[28] E. Alpaydin, Introduction to Machine Learning, MIT Press, Cambridge, MA, USA, 2010.
[29] M. Mohri, A. Rostamizadeh, and A. Talwalkar, Foundations of Machine Learning, MIT Press, Cambridge, MA, USA, 2012.
[30] K.P. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, Cambridge MA, USA, 2012.
[31] S. Bozkurt, G. Elibol, S. Gunal, and U. Yayan, "A comparative study on machine learning algorithms for indoor positioning," Proc. 2015 Int. Symp. Innov. Intell. Syst. Appl. (INISTA), Madrid, Spain, Sept. 2015. DOI: 10.1109/INISTA.2015.7276725
[32] X. Lu, H. Zou, H. Zhou, L, Xie, and G.-B. Huang, "Robust extreme learning machine with its application to indoor positioning," IEEE Trans. Cybern., vol.46, no.1, pp.194–205, Jan. 2016.
[33] H. Zou, B. Huang, X. Lu, H. Jiang, and L. Xie, "A robust indoor positioning system based on the procrustes analysis and weighted extreme learning machine," IEEE Trans. Wireless Commun., vol.15, no.2, pp.1252–1266, Feb. 2016.
[34] A.H. Salamah, M. Tamazin, M.A. Sharkas, and M. Khedr, "An enhanced WiFi indoor localization system based on machine learning," Proc. 7th Int. Conf. Indoor Posit. Indoor Navigat. (IPIN 2016), Madrid, Spain, Oct. 2016. DOI: 10.1109/IPIN.2016.7743586
[35] E. Homayounvala, M. Nabati, R. Shahbazian, S.A. Ghorashi, and V. Moghtadaiee, "A novel smartphone application for indoor positioning of users based on machine learning," Adj. Proc. 2019 ACM Int. Joint Conf. Pervas. Ubiquit. Comput. & Proc. 2019 ACM Int. Symp. Wearable Comput., New York, NY, USA, pp.430–437, Sept. 2019. DOI: 10.1145/3341162.3349300
[36] N. Lee and D. Han, "Magnetic indoor positioning system using deep neural network," Proc. 8th Int. Conf. Indoor Posit. Indoor Navigat. (IPIN 2017), Sapporo, Japan, Sept. 2017. DOI: 10.1109/IPIN.2017.8115887
[37] F. Al-homayani and M. Mahoor, "Improved indoor geomagnetic field fingerprinting for smartwatch localization using deep learning," Proc. 9th Int. Conf. Indoor Posit. Indoor Navigat. (IPIN 2018), Nantes, France, Sept. 2018. DOI: 10.1109/IPIN.2018.8626558
[38] Y. Chen, M. Zhou, and Z. Zheng, "Learning sequence-based fingerprint for magnetic indoor positioning system," IEEE Access, vol.7, pp.163231–163244, 2019.
[39] B. Bhattarai, R.K. Yadav, H.-S. Gang, and J.-Y. Pyun, "Geomagnetic field based indoor landmark classification using deep learning," IEEE Access, vol.7, pp.33943–33956, 2019.
[40] I. Ashraf, M. Kang, S. Hur, and Y. Park, "MINLOC: Magnetic field patterns-based indoor localization using convolutional neural networks," IEEE Access, vol.8, pp.66213–66227, 2020.
[41] T.-H. Chiang, Z.-H. Sun, H.-R. Shiu, K.C.-J. Lin, and Y.-C. Tseng, "Magnetic field-based localization in factories using neural network with robot sampling," IEEE Sensors J. vol.20, no.21, pp.13110–13118, Nov. 2020.
[42] L. Fernandes, S. Santos, M. Barandas, D. Folgado, R. Leonardo, R. Santos, A. Carreiro, and H. Gamboa, "An infrastructure-free magnetic-based indoor positioning system with deep learning," Sensors, vol.20, no.22, 6664, 2020.
[43] A. Sasaki and E. Ohta, "Magnetic-field-based position sensing using machine learning," IEEE Sensors J., vol.20, no.13, pp.7292–7302, July 2020.
[44] N.S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," Amer. Statistician, vol.46, no.3, pp.175–185, Aug. 1992.
[45] E. Ohta and A. Sasaki, "Magnetic-field-based positioning of mobile devices using neural networks," Proc. IEICE Soc. Conf. '20, no.B-15-15, Sept. 2020 (in Japanese).

Ai-ichiro Sasaki received the B.S. degree in applied physics from the Tokyo University of Science in 1996, the M.S. degree in Physics from the University of Tokyo in 1998, and the Ph.D. degree in electrical engineering from the Tokyo University of Science in 2006. From 1998 to 2018, he worked as a researcher with the NTT Science and Core Technology Laboratory Group. In 2018, he was appointed as an associate professor at Kindai University. His research interests include electromagnetic technologies ranging from LF bands to optical wavelengths.



Ken Fukushima received the B.E. degree in electronic engineering and computer science from Kindai University in 2020. He is currently with Kansai Transmission and Distribution, Inc.