Selective Unsupervised Learning-Based Wi-Fi Fingerprint System Using Autoencoder and GAN

J. H. Seong and D. H. Seo

Abstract—In this article, we propose an automatic Wi-Fi fingerprint system that combines an unsupervised dual radio mapping (UDRM) algorithm with the aim of reducing the time-cost needed to acquire Wi-Fi signals. Our proposed system is appropriate for indoor environments and utilizes a minimum description length principle (MDLP)-based radio map feedback (RMF) algorithm that simultaneously optimizes and updates the radio map. In the training phase, the proposed UDRM algorithm generates a radio map of the entire building based on the measured radio map of one reference floor. It does this by selectively applying a modified autoencoder and a generative adversarial network according to the spatial structures. Our proposed learning-based UDRM algorithm does not require labeled data, which is essential for supervised and semisupervised learning algorithms. It has a relatively low dependence on received signal strength indicator (RSSI) data sets. Our proposed RMF algorithm analyzes the distribution characteristics of the RSSIs for newly measured access points (APs) and feeds the analyzed results back to the radio map. The MDLP, which is applied to the proposed algorithm, improves the positioning performance and optimizes the size of the radio map by preventing the indefinite updating of the RSSI and by updating the newly added APs in the radio map.

Index Terms—Autoencoder, fingerprint, generative adversarial network (GAN), radio map, unsupervised learning.

I. INTRODUCTION

GLOBAL positioning system (GPS)-based positioning services have applications in various fields, such as the automotive industry, logistics, and security. However, providing such services in indoor spaces is difficult owing to the low signal transmittance of the GPS systems through the walls of buildings. Recently, as building structures have become larger and more complex, the need for positioning services in indoor environments has increased. Indoor positioning technologies, which can replace GPS systems, are therefore being constantly studied [1], [2]. Representative indoor positioning methods are broadly classified into either time-of-arrival (TOA) methods [3]–[5] or fingerprint methods.

TOA methods estimate the distance to a point by measuring the signal arrival times of the transmitter and receiver, and fingerprint methods estimate the indoor location by measuring the intensity of the relative access point (AP) signals. In contrast, fingerprint methods are suitable for most types of indoor communication, such as Wi-Fi, Bluetooth low energy, and ZigBee [6]–[8] as they measure the received signal strength indicator (RSSI), which indicates the relative intensities of radio signals between transmitters and receivers. This method has an advantage in that positioning is possible using only the universal device, negating the need for additional transmitters and receivers.

The fingerprint method is divided into the training phase and the positioning phase. During the training phase, a radio map database is produced using the measured RSSIs of APs at different indoor positions known as reference points (RPs). The RPs are set at intervals of 2–3 m. Wi-Fi signals measured at the RPs are used to create a radio map using a preprocessing algorithm, such as a deterministic or probabilistic model [9], [10], crowdsourcing [10], [11], or a classification model [9], [10]. Because these algorithms are simply aggregated into a radio map based on the RSSI measured on each floor, measuring the RSSIs at all RPs in the building is essential. In recent years, algorithms based on radio modeling [12]–[14] and neural networks [15], [16] have been studied for constructing radio maps.

During the positioning phase, the real-time position is estimated by comparing the radio map created in the training phase with the measured Wi-Fi RSSI from the receiver of the user. By comparing the RSSIs measured in real time with the previous radio map that was created while the user was moving, the RP of the radio map with the most similar distribution of RSSIs is recognized as the location of the user. Most algorithms used in this phase are relatively simple for real-time positioning, such as the support vector machine (SVM) [17], [18] and the k-nearest neighbor algorithm [19], [20]. Therefore, as the RSSI measurement process is required in all indoor spaces, considerable time and effort are required for the creation of the radio map in the fingerprint system. However, in recent years, as building sizes and the complexity of building structures have increased, the workload and time required have consequently increased rapidly.

To solve this problem, Wang *et al.* [15] proposed a novel deep-learning-based indoor fingerprinting system that employs

channel state information (CSI). This algorithm estimates the location by storing the weights of the deep network obtained by learning CSI data measured from one AP in the form of a radio map. Because the proposed deep neural network (DNN) is based on supervised learning, labeled data must be inputted separately from positioning data during the learning process; however, it struggles to cope with variations in Wi-Fi signal strength caused by indoor structures. Jiang *et al.* [16] and Gu *et al.* [21] proposed a fingerprint system based on a semisupervised extreme learning machine. This system significantly reduces the workload through a semisupervised learning framework and utilizes a fusing fingerprint to markedly improve the location accuracy. However, it requires considerable workload to create a radio map and to obtain RSSI measurements for all RPs. Kim *et al.* [22] proposed a DNN architecture comprising a stacked autoencoder to reduce the feature space dimensions and a feedforward multi-label classifier to classify scalable buildings/floors and estimate the floor-level location. This system has the advantage of organically managing the radio map of the entire building through a single DNN. However, the RSSI measurements are essential at all RPs in the building, and modifications of the network are essential if new APs are installed.

In recent years, the use of fingerprints has been a major research objective aimed at improving accuracy and reducing workloads. Existing fingerprint methods, including the aforementioned studies, have essentially required measurements of the entire building or a partial area of each floor. Consequently, such studies require considerable time-cost and workload.

To solve these problems, we propose an automatic Wi-Fi fingerprint system based on unsupervised learning that combines an unsupervised dual radio mapping (UDRM) algorithm with a minimum description length principle (MDLP)-based radio map feedback (RMF) algorithm. To create an initial radio map of other floors based on the RSSI measured for one floor, our proposed unsupervised learning-based UDRM algorithm, which is divided into a modified autoencoder and a modified generative adversarial network (GAN), is separately applied to these algorithms depending on the indoor structures on each floor. On the basis of the initial entire radio map created by our proposed UDRM algorithm, the proposed RMF algorithm is adapted to cope with not only the positioning but also the creation and removal of new APs during the positioning phase. Therefore, the proposed Wi-Fi fingerprint system significantly reduces the workload and construction time required for measuring Wi-Fi RSSIs through postprocessing. It does so by subordinating the RSSI measurement process during the training phase to the positioning phase. In addition, because the indoor structure and location of the AP are considered without measuring the RSSIs on new floors, the radio map obtained is very similar to the measured radio map. Our approach is a novel fingerprint algorithm based on unsupervised learning considering the space environment.

The remainder of this article is organized as follows. Section II reviews the principles of unsupervised learning. Section III presents the proposed fingerprint system, and Section IV shows the experiment and results. Finally, Section V presents the conclusions.
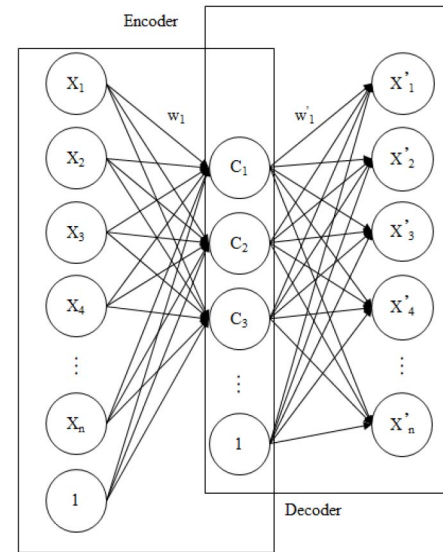

Fig. 1. Structure of a conventional autoencoder.

II. Unsupervised Learning

A. Autoencoder

Supervised and semisupervised learning based on neural network algorithms [e.g., convolutional neural networks (CNNs) and recurrent neural networks (RNNs)] use the expected output data to update connected weights around the hidden layer. Thus, to train the neural network, these algorithms require minimal input data and labeling processes. The labeling involves inducing the output of the desired data by allowing the user to directly compare the expected output data with the trained output data. This is identical to the concept of passive clustering. Data created through this process are called "labeled data" and are essential for acquiring a properly trained model for supervised and semisupervised learning. Supervised/semisupervised learning and unsupervised learning are classified depending on the use of labeled data. Therefore, because the output generated by learning is highly dependent on the labeled data, the predictable range of the output data is limited compared with that for unsupervised learning.

However, as the autoencoder is a type of unsupervised learning algorithm, labeled data are not required, as shown in Fig. 1. Instead, this algorithm performs labeling based on the probability distribution of features by using only the input data. When a set of the training data for output generation is $X = [X_1, X_2, X_3, \ldots, X_n]$, the input data of the algorithm are as follows:

$$X_{\text{in}} = [X_1, X_2, X_3, \ldots, X_n, 1]. \tag{1}$$

Here, the added input value 1 is a bias used to maximize the approximation by quickly adjusting the characteristic curve of the probability distribution to derive the feature C in the hidden layer, according to the training data. Thus, the iterative computation of the weight (W, W') is reduced through this value. The hidden layer is applied to both the encoder and the
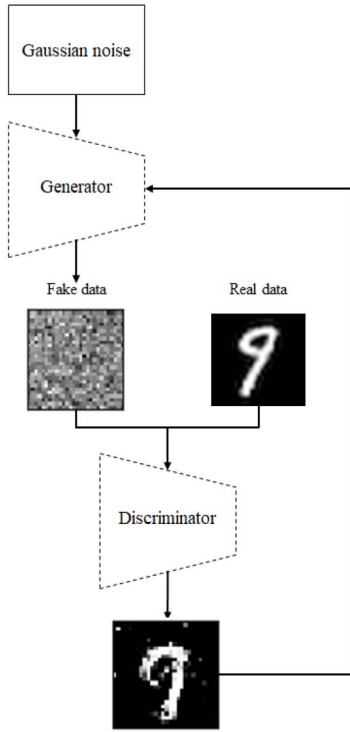
Fig. 2. Structure of the conventional GAN.

decoder, as indicated, respectively, by

$$C_n = \sigma(WX_n + 1) \qquad (2)$$

$$X_{\text{out}} = \sigma\left(W'^{X'_n} + 1\right). \qquad (3)$$

Here, X'_n represents the output data for each input datum X_n, σ represents the activation function, C_n represents the nth feature vector function, and 1 is the bias of the RSSI data. This function is used to recreate the output from the decoder. As the amount of input data (training data) increases, C_n is trained as a function with a more accurate output value [23]. Thus, because the number of input and output data sets in the autoencoder is always identical, the fields of application are limited.

B. Generative Adversarial Networks

GANs, a type of unsupervised learning algorithm, were recently proposed to improve the prediction performance using mutual learning of the training model. The GAN creates the learning function while increasing the accuracy via competition between the neural network-based "generators," which creates data and the "discriminator," which compares real data and generator-based data. Fig. 2 shows the structure of this network. For the initial training, the generator, which receives Gaussian noise as the input, generates the fake data of a predefined 3 m × 3 m pixel size, such as an interval of RPs. The discriminator compares the real data with the fake data to be trained and iteratively confirms and trains only the matched pixels in the fake data. Simultaneously, for adversarial learning of the generator and the discriminator, the generator receives only the information pertaining to unmatched pixels from the discriminator. The adversarial learning process of the
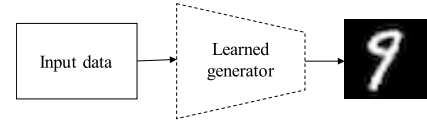

Fig. 3. Application of the learned generator.

fake data is iteratively performed until the fake data are similar to the real data. This adversarial learning process (i.e., minimax game) can be expressed as the function $V(D, G)$ as follows [24]:

$$\min_G \max_D \quad V(D, G) = E_{x \sim p_{\text{data}}(x)}\big[\log D(x)\big]$$
$$+ E_{z \sim p_z(z)}\big[\log(1 - D(G(z)))\big] \qquad (4)$$

where z and x represent the input data of the generator (G) and discriminator (D), respectively. To the generator, G is learned to the distribution curve over data x. G implicitly defines a probability distribution p_g as the distribution of samples $G(z)$ that are obtained when $z \sim p_z$. $D(x)$ represents the probability that x came from the data rather than from the distribution curve. The differential function (fake data) of the first multilayer perceptron with Gaussian noise ($p_z(z)$) and parameters θ_g is expressed as $G(z; \theta_g)$ in the data space.

As G and D progressively increase their learning probability through adversarial learning, the fake data generated by the generator become similar to the real data. Moreover, because D, which classifies the data as true and false, finally discriminates only the true data, the ideal distribution curve of D appears in the form of a straight line when learning is completed. When the generator becomes a generative machine, in that the fake data are similar to the real data, the discriminator is finally removed to use the learning machine and is applied, as shown in Fig. 3. Here, the learned generator is a network that generates the number 9. When a generative machine comprising these multiple generators is composed, a new network that can distinguish and generate more various data from the input data can be created.

C. Minimum Description Length Principle

As one of the discretization algorithms that can separate data into several signals, according to continuity, the MDLP is an algorithm that can simultaneously optimize and classify data. This algorithm divides the sets of RSSIs for discretization using entropy, which is a measure of disorder [25], [26]. Entropy expresses the probability that a certain value will appear in a data set as follows:

$$P_i = \frac{D_i}{T} \qquad (5)$$

where T represents the total amount of data and D_i represents the number of times that any arbitrary value has appeared. Therefore, P_i is the probability that a single value is derived from the entire data set. According to the definition of the log function, it can be expressed using the following equation:

$$- \log_2 P_i = I(m). \qquad (6)$$

Here, I represents the amount of information and m represents an arbitrary value in the set. When $P_i = 1$, for which

the data set is constant, it can be confirmed that this data set has regularity with an amplitude of one RSSI. In contrast, if $P_i = 0$, the value m does not exist in the data set. The amount of information for such a value can be mathematically expressed as follows:

$$\text{Ent}(N) = -\sum P_i \log_2 P_i. \tag{7}$$

Here, Ent represents the entropy, which is the average information amount. If the elements in the set have regularity or appear as one value, the entropy is reduced. A greater irregularity of the data corresponds to higher entropy. Thus, a set that explicitly presents the distinction of measured elements in a continuous time order indicates that the number of subsets divided by the discretization increases and that the values of the subset clearly differ from each other. The MDLP induced by a cut point C for a set S of N examples is expressed as follows:

$$\text{Gain}(X, C; S) > \frac{\log_2(N-1)}{N} + \frac{\Delta(X, C; S)}{N} \tag{8}$$

where S refers to all data in the set and C represents the cut point for the condition X. The information gains (IGs) of S, S_1, and S_2 are computed using the cut-point selection algorithm. Therefore, the increasing number of subsets that result from the discretization of the set indicates that the RPs are properly classified automatically. Accordingly, if the MDLP is applied to the relationship between AP aggregates rather than to a single set, the APs can be removed according to the property, whereby signals that should have similar distributions of elements are not divided [27].

III. PROPOSED FINGERPRINT SYSTEM

The proposed system is divided into the training phase and the positioning phase, similar to the conventional fingerprint method. Therefore, we separately describe the UDRM proposed for the training phase and the MDLP-based RMF proposed for the positioning phase.

A. Proposed UDRM Algorithm

In contrast to a Wi-Fi measurement-based radio map, a prediction-based radio map must consider the indoor environment owing to the large attenuation of the RSSI depending on the indoor area and structure. To solve this problem, the proposed UDRM is divided into two algorithms (modified autoencoder and GAN) that predict and generate the radio maps of each floor, as shown in Fig. 4. These algorithms are separately applied depending on a comparison of the indoor structures of the building on each floor. To learn the minimum indoor environment and Wi-Fi signals, we select a reference floor in a building that measures real Wi-Fi signals to train the modified autoencoder and GAN. The measured Wi-Fi signals [service set identifier (SSID) and RSSI] on the reference floor are indispensable for learning and generating a 2-D map of each floor. First, to minimize the complexity of the learning algorithm, the proposed algorithm measures the SSID and relative RSSI according to a set of RP on the reference floor of the building in which user locations can be recognized. According
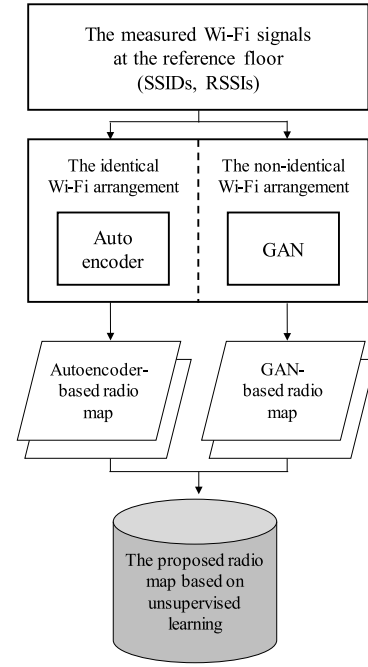

Fig. 4. Structure of the proposed UDRM algorithm.

to the measured Wi-Fi signals, a real radio map for learning is preferentially generated on the reference floor. The RSSIs in the radio map only measure fixed APs whose coordinates are not changed for learning. Generally, the main APs installed in a building cover the entire area and are always fixed. In addition, the coordinates of the APs are easily obtained using indoor 2-D maps. Then, depending on the floor on which the radio map is to be generated, the autoencoder or the GAN is selectively applied according to the indoor map and the coordinates of APs on other floors. If there is a 2-D map of the new floor with AP coordinates, the proposed algorithm can be automatically applied. To reduce the iterative learning time and ensure the positioning accuracy during the positioning phase, the initially measured APs (i.e., learning objects) are selected only for the main Wi-Fi APs with an extremely low probability of being removed. The modified autoencoder in the proposed UDRM algorithm is then applied to other floors that have indoor structures and AP coordinates that are similar to those of the reference floor. Unlike the original autoencoder having one input, the proposed modified autoencoder learns by designing the RSSI of the reference floor and the AP coordinates of the other floor as the input data and generates a radio map by mapping it to a 2-D map of the other floor. In the modified GAN, unlike the original GAN network where Gaussian noise is the input, the design is such that it generates a trained RSSI centered on the AP coordinates by adding an imaged 2-D radio map and AP coordinates as inputs.

Fig. 5 shows the structure of a modified autoencoder that learns measured RSSIs on the reference floor and that maps the expected RSSIs in the 2-D map for other floors of the same structure. The input radio map, which is measured on the reference floor, is updated by comparing the weight of
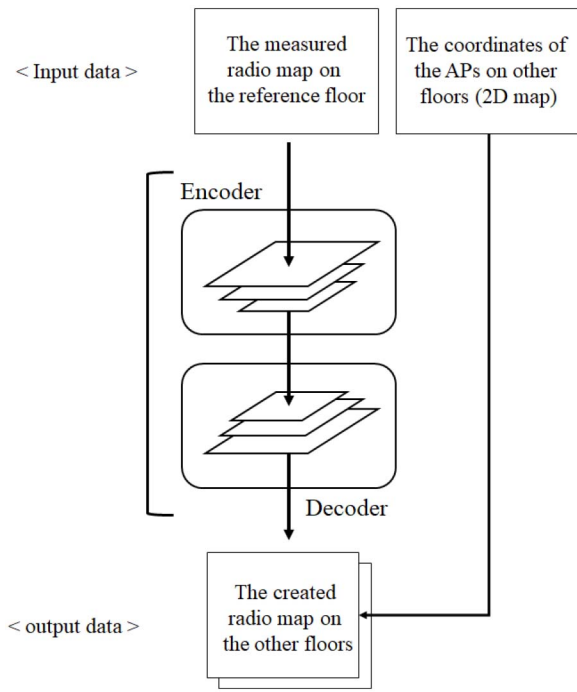
Fig. 5. Learning structure of the modified autoencoder.

Fig. 6. Learning structure of the modified GAN.

Fig. 7. Application method of the modified learned generator.

the autoencoder with that of the generated radio map. The autoencoder comprises both encoders and decoders.

In the modified autoencoder, the encoders and decoders are symmetrically designed to compress and expand the types of input data. Because the output radio map is expected as an RSSI for each reference position on the reference floor, combining it with fixed AP coordinates of real 2-D maps is necessary. As the output radio map is a radio map of a new floor that has not been measured, the fixed AP coordinates of the new floor are inputted. Therefore, since a radio map is generated based on the coordinates of the new floor, the modified autoencoder is applied to AP coordinates that are almost similar to or coincident with the reference floor. Finally, the RSSI distribution of each AP is mapped to the 2-D map to complete the predicted radio map.

Therefore, in the proposed modified autoencoder, the input values are as follows:

$$AP_k = \left[RSSI_{Re1\_k}, RSSI_{Re2\_k}, RSSI_{Re3\_k}, \ldots, RSSI_{Ren\_k}, 1 \right] \quad (9)$$

where k represents an arbitrary AP measured on the reference floor and n represents the number of RPs and the measurement position. The signal distribution of the AP output through the autoencoder is generated as new floor AP signals by matching them with the coordinates of APs of similar positions on different floors of the uniform building to be predicted.

Fig. 6 shows the structure of the proposed GAN. In contrast to the typical GAN structure in which only Gaussian noise is inputted to the generator, the modified GAN receives Gaussian noise as the initial data input, along with a 2-D map and AP coordinates of the new floors. The 2-D map represents the input data for effective radio map matching because it reflects the indoor structures of the floors based on the measured radio
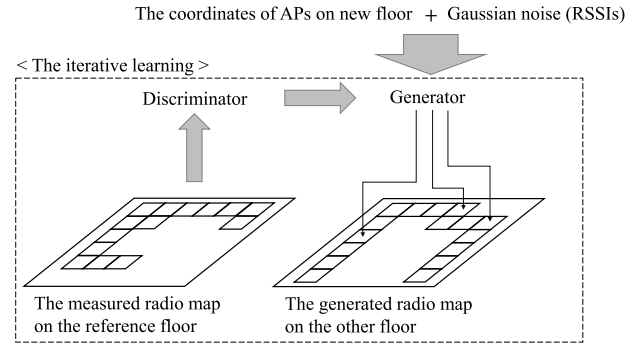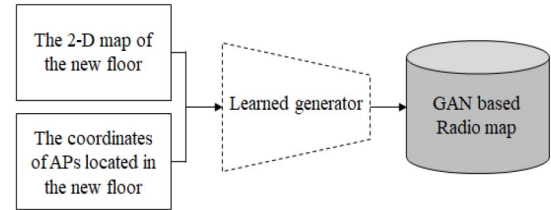
map of a reference floor. Therefore, the proposed GAN predicts and generates the new radio map reflecting the spatial distance and structure around the new AP coordinates based on the learned RSSI distribution according to the space of the reference floor.

Here, the adversarial learning process is as follows:

$$\min_G \max_D \ V(D, G) = E_{x \sim p_{\text{data}}(x\_\text{wifi})} \left[\log D(x\_\text{wifi}) \right] \\ + E_{z \sim p_z(z\_\text{wifi})} \left[\log(1 - D(G(z\_\text{wifi}))) \right] \quad (10)$$

where x_wifi represents the actual RSSI collected from one Wi-Fi AP in the form of a 2-D radio map, and z_wifi represents the noise data reshaped into a 2-D radio map input to the generator. This equation maximizes $1 - D(G(z\_\text{wifi}))$, which is the probability of identifying the virtual radio map generated from the generator as a useful radio map by inputting the real radio map data x_wifi into the discriminator.

Thus, the AP coordinates are the essential data for generating the radio map at the correct position and predicting the RSSIs. An initially nonlearned generator creates random RSSI distributions from the Wi-Fi signals that are generated by Gaussian noise using the input 2-D map and AP coordinates. The fake radio map, which is generated by the Gaussian noise, categorizes the real measured RSSI on the reference floor and each generated RSSI by the discriminator as either true or false for each pixel in the map.

Fig. 7 shows the radio map generated by the learned generator of the GAN for new floors. The learned generator combines the various learned types of kernels and predicts the RSSI distributions using the 2-D map and AP coordinates for floors that are different. Thus, the proposed radio map is generated by the combined autoencoder and GAN maps, so that the RSSI distributions of all floors and areas are generated without the need for complex measurement processes in a new area. In contrast to the conventional radio map that is applied in
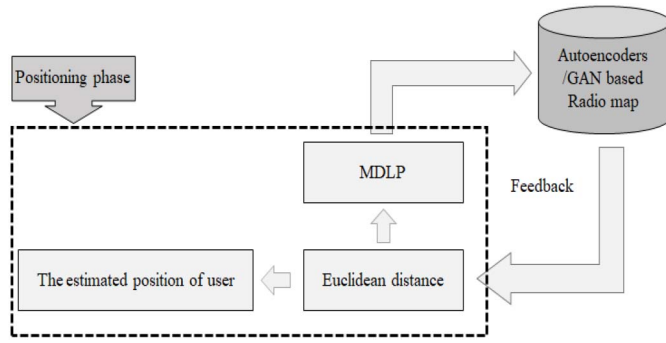
Fig. 8. Proposed flowchart of the positioning phase.

the fingerprint method, the radio map applied in the proposed system has learned not only the propagation characteristics but also the indoor structures of the building. The real radio map of the reference floor used as the input data during the learning of the autoencoder, which was applied to similar areas and floors, remained the same. However, the outputs derived from the prediction of the autoencoder were interpolated, generating RSSIs that were not 100% identical. Therefore, the radio map was preferentially generated using fake RSSIs, which linearly approximated the attenuation of the RSSIs. Then, the generated RSSIs were input at the AP coordinates, and a modified autoencoder-based radio map was finally completed.

In the case of a neural network that learns 100% of the input data, overfitting occurs because the modified autoencoder and GAN-based unsupervised learning algorithms are designed on the basis of networks, such as the CNN and RNN [28]–[30]. In contrast, if the amount of learning data is small, the learning rate is low, and the expected accuracy of the radio map is reduced. Therefore, the network is designed with an accuracy of 90%–100% depending on the application. The RSSI of Wi-Fi is generally not constant and contains noise; thus, it is suitable for application to the proposed network.

B. MDLP-Based RMF Algorithm

The radio maps generated for each floor using the proposed autoencoder and GAN do not measure or predict high-mobility APs, reducing the time-cost and workload required for RSSI measurement. Therefore, additional updating of individual APs is needed to improve the positioning accuracy. To accomplish this, an MDLP-based RMF algorithm is proposed, which can perform updating during positioning using the generated radio map of the UDRM.

Fig. 8 shows the structure of the proposed algorithm in the positioning phase. To recognize user locations using the radio map based on the proposed UDRM, it is essential that the users have Wi-Fi-enabled smart devices. When the radio map is acquired through such a device in real time, the radio map based on the UDRM and the measured AP signals are compared using the Euclidean distance algorithm as follows:

$$P = \min\left(\sqrt{\sum_{j=1}^{n} \left(AP_j - AP_{rj}\right)^2}\right) \tag{11}$$

where P represents the final position of the user, n represents the number of measured RSSIs, and AP_j and AP_{rj} represent the RSSIs of the jth SSID in the radio map and the RSSI measured by the user in real time, respectively. The RP of the radio map with the highest similarity is identified as the current position of the user. The Euclidean distance algorithm [31]–[33] is often used for real-time multi-positioning because it can specify a position quickly without complex computation. While the above-described process occurs during the conventional positioning phase, the proposed MDLP-based RMF algorithm optimizes and updates the new APs that are not present in the radio map during the calculation using the Euclidean distance algorithm. This method improves the positioning performance by filtering out unnecessary AP signals and automatically reducing the dimensions of the radio map in the area where there are many users and a high Wi-Fi AP density.

The measured real-time Wi-Fi signals can be positioned according to the Euclidean distance, and the RSSIs are updated to the UDRM-based radio map. However, if all measured APs are inputted, the radio map expands as new APs are created or discovered, reducing the operation speed. Therefore, the proposed MDLP-based RMF algorithm removes unnecessary APs and updates the radio map. In contrast to the existing methods, such as the SVM, which discriminate and process a set of continuous RSSIs measured at each AP or RP, the proposed algorithm obtains the signal distribution characteristics of each AP. Then, it reduces the radio map by leaving only one representative AP from among multiple APs that have the same signal distribution characteristics according to the RP.

APs with similar signal characteristics have a very small influence on positioning, but they increase the number of computations and the size of the radio map. The conventional MDLP algorithm cannot numerically express the degree of signal separation according to the signal RPs. Thus, after the MDLP is applied, the proposed algorithm performs the IG operation to quantify it. The IG is expressed as follows:

$$H(y) = -\sum p(y)\log_2(p(y)) \tag{12}$$

$$H(y|x) = \sum p(x) \sum p(y|x)\log_2(p(y|x)) \tag{13}$$

$$IG = H(Y) - H(Y|X) \tag{14}$$

where x and y represent an AP and an RP, respectively. $H(Y)$ represents the entropy based on one RP, and p represents the probability that the RSSI value will be measured. The data characteristics of the quantified AP numerically represent the similarity between different AP signals, so that only those with the best classification among signals of similar distribution remain. Thus, each AP can visually confirm the RSSI distinguishing ability according to the RPs and manage the Wi-Fi signal strictly through that threshold. Therefore, the proposed MDLP-based RMF algorithm is designed to update and optimize the collected Wi-Fi signals simultaneously with the positioning stage.

As the MDLP only splits the continuous data, there is no numerical value to check the result separately. Therefore, to verify the performance of the MDLP, the IG, which indicates the amount of entropy, is applied to determine the continuity of data after the application of the MDLP. In this case,
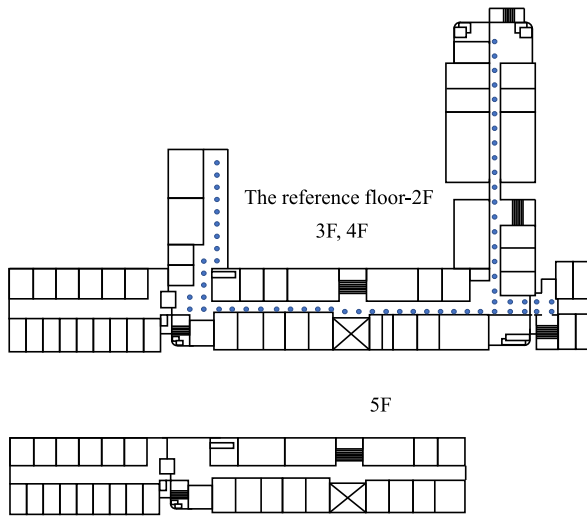
Fig. 9. Structure of the reference floor and experimental floors.

the IG values of the RSSI set, which clearly distinguish the data according to each RP, are larger, and the APs that the RSSIs find difficult to distinguish are converged to 0 by the IG. Thus, the RSSI set of APs with a value of 0 can be defined as APs that cause confusion in positioning because the measured RSSI results are similar at different RPs. Therefore, APs are removed through thresholding as follows:

$$IG(M(AP)) > 0 \tag{15}$$

where M and IG represent the MDLP function and the IG function, respectively. After the MDLP is applied to each AP data set, the IG is applied for mathematical discrimination. "0" represents the data that are unnecessary for the operation as they indicate that it is very difficult to classify the signal strength of each AP.

Class-attribute interdependency maximization and the class-attribute contingency coefficient are discretization algorithms similar to the MDLP [27]. In contrast to the MDLP, which is based on entropy, both these algorithms partition data according to probability-based interdependencies. Therefore, as these algorithms do not consider entropy during the computation process, an entropy-derivation process is needed to apply the IG. In addition, their discretization performance is lower than that of the MDLP.

IV. EXPERIMENTS AND RESULTS

A. Experimental Environment and Configuration

To verify the performance of the proposed UDRM algorithm and MDLP-based RMF algorithm, an experimental area was selected at the College of Engineering, No. 1 Building, Korea Maritime and Ocean University, as shown in Fig. 9. This figure (2-D map) was manufactured using the computer-aided design tools. If a design drawing of the building can be secured as an image file, a separate indoor map-drawing process is unnecessary, and the size of the image can be easily changed through the settings without incurring any initial cost during preprocessing. The size of the 2-D map in the experiment was set to 23×29 pixels. The first floor, which

was difficult to position, was excluded from the experimental area owing to the small number of APs. Because the second, third, and fourth floors had the same structure, a radio map was generated through a modified autoencoder, and a modified GAN was applied to the fifth floor, which had a different indoor structure. The second floor, where a sufficient number of APs were measured, was set as a reference floor to obtain real RSSIs for learning.

In this process, 63 RPs were set to generate a radio map for learning, and 284 AP signals were measured on all floors. In the training phase, because the personal APs could easily change via generation and movement, only the public APs were measured on all floors. Personal APs that were omitted during the training phase could be updated through the proposed RMF algorithm in the positioning phase. In a corridor of the reference floor, eight public APs were measured from the walking survey (out of 139). A setting of the 2-D map for learning assumed a line on the image to be a structure that completely blocked Wi-Fi signals. In contrast, the measured RSSIs of APs in the corridor spaces outside the boundaries were designed to estimate the Wi-Fi signal in a multipath environment, which is reflected by its unique characteristics.

Based on 63 RPs of the experimental environment, the modified autoencoder consists of three encoder and three decoder layers. In addition, it was designed with the same input and the output so that it can easily train the RSS distribution of the AP in the target space.

In the modified GAN model, the actual building structure is mapped equally at intervals of 3 m in accordance with the environment in order to input the geographical information. Then, because the coverage of one AP was specified in the multipath environment, it was designed as a fully connected layer. Therefore, it was composed of three layers such as the modified autoencoder because for typical positioning, the application environment requires a sufficient number of dimensions for training, and it has a large number of APs and RPs.

As shown in Fig. 10, an Android-based measurement app was developed using a smartphone to measure RSSIs. The measurement items were SSIDs and RSSIs, and the generated radio map used MySQL, enabling the management of the radio map in real time. The modified autoencoder applied the measurement data as input data without preprocessing. Thus, it directly transmitted the collected Wi-Fi data to MySQL and generated the radio map.

B. Results of UDRM Algorithm

To verify the performance of the proposed UDRM comprising the modified autoencoder and GAN, the generated radio map was compared and analyzed with the real measured radio map.

Fig. 11 shows the radio map of the third floor generated using the measured RSSIs of the second floor. One pixel on the x- and y-axes represents an interval of 3 m on the 2-D map plane. Fig. 11(a) shows the real measured radio map, and Fig. 11(b) shows the radio map generated by the modified autoencoder on the third floor. The RSSIs of the propagated Wi-Fi signal from the AP in the corridor are expressed in
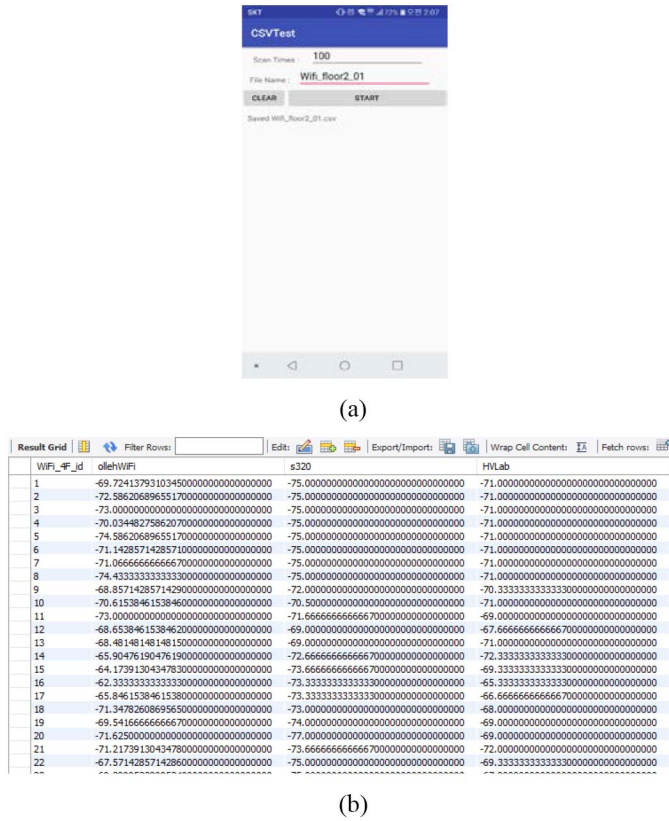
(a)

(b)

Fig. 10. Generated program for operating the proposed algorithm. (a) Android-based Wi-Fi app. (b) Management server of radio map.

(a)

(b)

Fig. 11. Radio map generated by the modified autoencoder (third floor). (a) Measured radio map. (b) Radio map generated by the autoencoder.

color intervals of 3 m, as with the arrangement of RPs. Those that are expressed in red are stronger, whereas blue indicates a weaker signal. According to the AP coordinates, the real RSSIs and the modified autoencoder-based radio map were replicated almost identically. However, there were differences between the predicted and measured RSSIs when the relative distance from the AP coordinate was >30 m. Because the measured AP signal strength deceases as the distance increases, compared with the RSSI of strong Wi-Fi signals, there was a smaller number of measurements in the indoor interval where the RSSI of the Wi-Fi was measured from −90 to −100 dBm. Thus, when the same number of measurements were performed, APs with strong signals were measured as closer to 100%. APs with low signal intensities had relatively low collection rates compared with the number of measurements. Because it was difficult to acquire a sufficient number of measurement signals in the area where the RSSI was weak, depending on the increasing relative distance between the AP and the RP, there were differences in the amount of learning. This causes an error between the real measured radio map and the radio map generated via learning. For the second and third floors, nearly identical radio maps were acquired because the indoor structures and AP coordinates were the same. A detailed analysis of this result reveals a unique signal for the AP: an RSSI measured at an unexpected location P. This resulted from a unique indoor structure where RSSIs are transmitted through windows rather than walls. As P was relatively far from the AP, an error occurred.
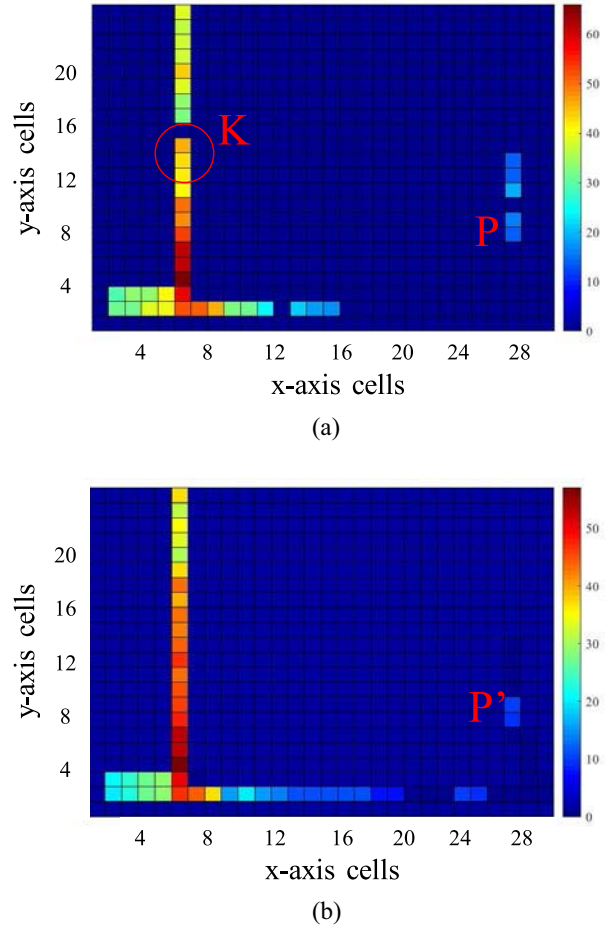
However, the predicted signal measured in the unexpected area by the indoor structure was similar to that of P'. K represents a missing data space caused by nonreceipt of the signal. This is often caused by a walking survey. Despite the existence of this space, the replicated Wi-Fi distribution generated an RSSI via prediction, indicating that the resilience against the missing signal was excellent.

Fig. 12(a) and (b) shows the measured and predicted radio maps, respectively, based on the modified autoencoder on the fourth floor. The predicted radio map is similar to that seen in Fig. 11, indicating that the prediction accuracy was slightly reduced in spaces where the RSSI was very weak.

Thus, the modified autoencoder accurately predicted AP signals in the area where measured RSSIs were strong, and the predicted RSSI accuracy was lower in the areas where the measured RSSIs were weak. Because the relatively weak RSSI reduced the number of measurements, the same RSSI measurement time was used for the RPs when designing the radio map. There was a limit to increasing the measurement time at each RP. However, if RSSIs were designed as a radio map according to the RPs of all APs, there would be many measured APs with strong signals, and their influence on the positioning would be reduced.
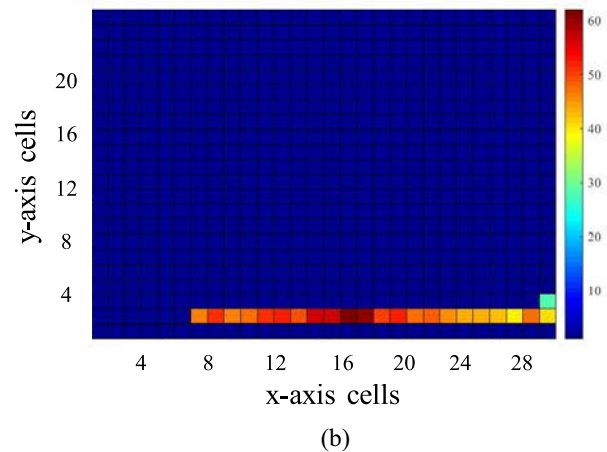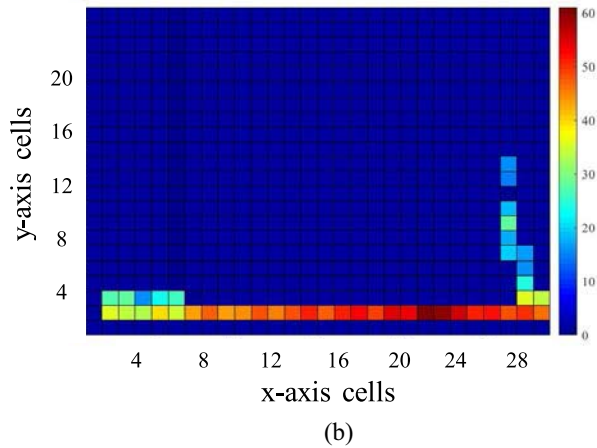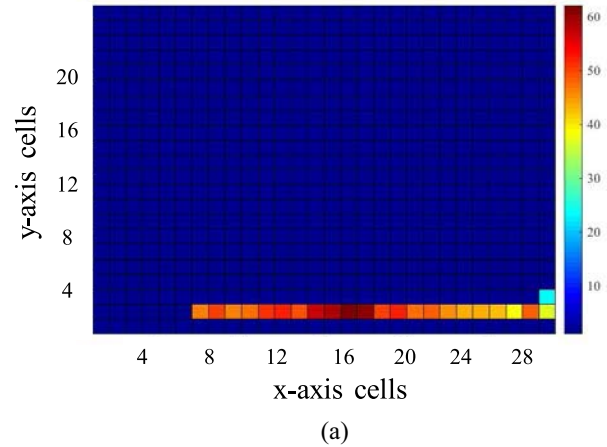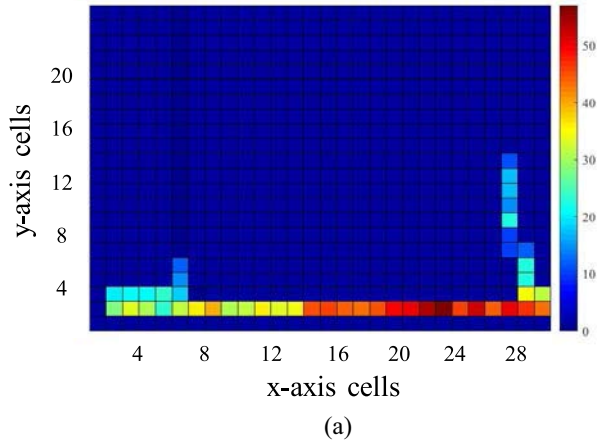
Fig. 12. Radio map generated by the modified autoencoder (fourth floor). (a) Measured radio map. (b) Radio map generated by the autoencoder.

Fig. 13. Radio map generated by the modified GAN (fifth floor). (a) Measured radio map. (b) Radio map generated by the GAN.

Fig. 13(a) and (b) shows the measured and predicted radio maps, respectively, based on the modified GAN on the fifth floor. The fifth floor had fewer public APs, a different floor plan, and a smaller indoor area. Compared with Fig. 12, the AP coordinate was slightly different. Because the right area represents the outdoors, the AP was arranged such that it covers the indoor area. Thus, the RSSIs that learned from the reference floor containing only the indoor space were excluded from the prediction because the RSSI characteristics could not be learned in the outdoor environment. The unlearned area was interpolated through the proposed MDLP-based RMF algorithm during the positioning phase. The fifth floor differed from the other floors with regard to its indoor structures. However, because the learning was performed using various kernels, all of the kernels other than 1×1 were applied to combine the RSSI and indoor structures. Therefore, the modified GAN indicated that the predicted radio map was similar to the real radio map only when the AP coordinates were inputted exactly as in the autoencoder.

Table I presents the accuracy of the two types of radio maps generated by the UDRM. As a result of applying the algorithm to generate the radio map for all floors, when the real measured radio map reference was set as 100%, the accuracies of the

TABLE I
ACCURACY OF THE RADIO MAP ACCORDING TO THE UDRM

Algorithm	Modified autoencoder	Modified GAN
Accuracy of radio map	90.40%	92.70%

modified autoencoder and the modified GAN were determined to be 90.40% and 92.70%, respectively. Although there was a limitation in generating an exact 100% radio map, the error rates of the two algorithms were 9.6% and 7.3%, respectively, and were within the range of 5%–10%, which is a suitable error rate for a general learning network. The UDRM results confirm that the presence of stable systems is a characteristic of neural networks.

Table II presents the computation time of the proposed algorithms according to the size of the data set. A batch represents a subset of the total data, and an epoch represents the number of data iterations. In the proposed algorithm, 120 epochs (the size of all RSSIs) were divided into groups of 20 epochs to prevent overfitting instead of slowing down the operation, even though this increased the amount of RSSI data for learning.

TABLE II
ALGORITHM PROCESSING SPEED ACCORDING TO
THE NUMBER OF ITERATIONS

Algorithm	1 batch (μs/step)	20 batches (ms/step)	20 batches / 5000 iterations (min)
Autoencoder	334	2	18.12
GAN	1000	8	64.40

While reducing the number of epochs increased the computation speed, it caused overfitting owing to the stronger data characteristics of each epoch.

The 20 epochs were repeated 5000 times in the proposed UDRM, and the total learning time was 18.12 min for the modified autoencoder and 65.40 min for the modified GAN. Thus, the autoencoder exhibited a relatively high processing speed. When all operations were performed, the processing speeds of the two algorithms differed by a factor of three. When the operation was performed 20 times, the difference was fourfold. Thus, the difference in the processing speed increased with the number of operations. Owing to the structural characteristics of the GAN, the processing speed increased sharply with an increase in the number of operations. The modified autoencoder had a nearly constant learning time when the iteration count was 5000. However, the modified GAN took 65.40 min less than the expected time because the generator had learned, and the operation speed was higher. Therefore, excluding the first floor, the time-cost required to create a radio map of the entire building based on the proposed UDRM was <2 h.

C. Result of MDLP-Based RMF Algorithm

Fig. 14 shows the MDLP result obtained for the radio map during the positioning phase. The x-axis indicates the number of personal APs measured on the second floor, and the y-axis indicates the IG values of the corresponding APs. The IG is a measure of the independence within a group of APs or RSSIs. Fig. 14(a) shows the result of the application of the IG to all measured APs by the user, without applying the MDLP on the second floor. Therefore, all 170 APs measured on the second floor were expressed as IG values without being removed.

Fig. 14(b) shows the result of AP removal via the MDLP. The x-axis indicates the number of personal APs measured on the second floor, and the y-axis indicates the IG values of the corresponding APs. Compared with the results shown in Fig. 14(a), the proposed MDLP removed 90 of the 170 APs. During this process, the measured radio map was input as is, in real time. The side effect of the radio map experiencing indiscriminate updates was prevented in advance. Thus, only the necessary APs were added to the radio map. According to the gradual slope of the IG values of Fig. 14(a), the distributions of the RSSIs and indoor structures of the second floor were not very complex.

APs with very small IG values were removed by the proposed MDLP because the change in the RSSIs was small,
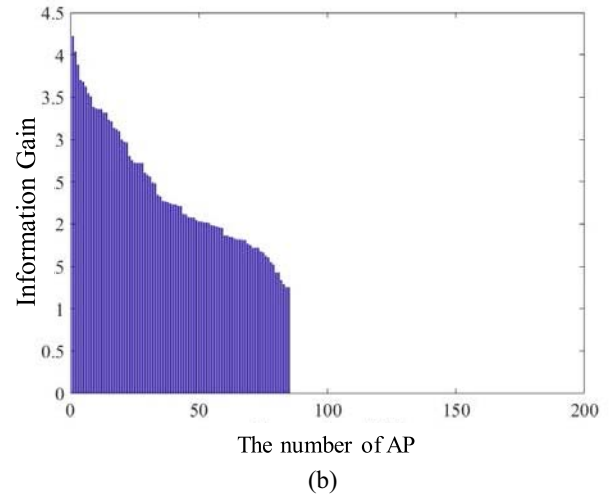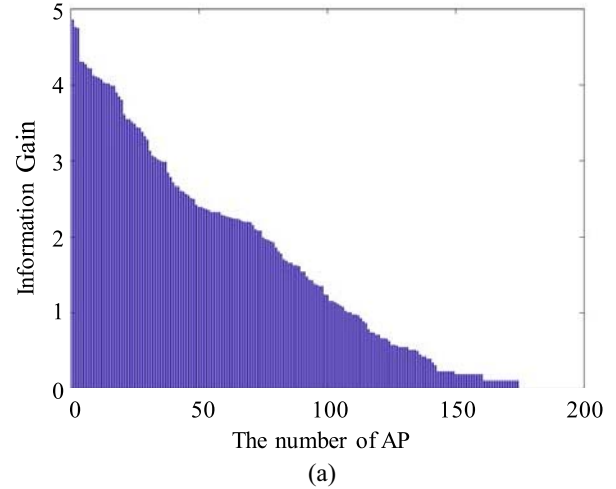

Fig. 14. MDLP results for the radio map during the positioning phase (second floor). (a) IG results for measured APs by user. (b) MDLP results for measured APs.

and it was difficult to distinguish them from the RPs. In addition, the APs were well-separated, but the RSSIs lacked continuity. Thus, APs with large IG values were clustered around APs with similar RSSI distributions. Otherwise, APs with the same RSSI appeared randomly at other RPs. Therefore, the IG value was not a criterion for the removal of APs, but it was useful for visually confirming the RSSIs of an AP. The removal of APs was performed by the MDLP, according to the continuity of RSSIs and the distribution of APs.

Fig. 15 shows the MDLP result obtained for the radio map based on the modified GAN for the fifth floor. The x-axis indicates the number of measured APs, and the y-axis indicates the IG of the corresponding APs. Fig. 15(a) shows that only 75 signals were detected, in contrast to the number of APs measured on other floors. Further, the measured APs on the fifth floor were updated using the same method that was employed for the other floors. Fig. 15(b) shows the result of AP removal via the MDLP and the IG obtained using the radio map based on the GAN. The signal measured in (a) was finally reduced to 46 signals through the MDLP and IG.
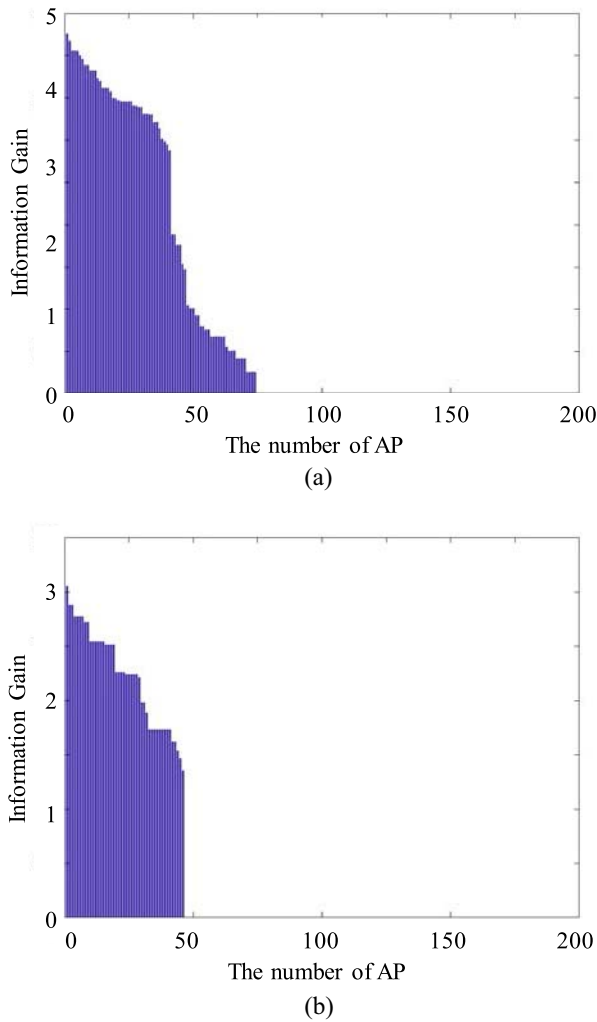
Fig. 15. MDLP results for the radio map during the positioning phase (fifth floor). (a) IG results for measured APs on the fifth floor. (b) MDLP results for measured APs on the fifth floor.

TABLE III
POSITIONING ACCURACY OF THE PROPOSED FINGERPRINT ALGORITHM AND THE MEASUREMENT-BASED FINGERPRINT SYSTEM

Floor	Measurement-based fingerprint system	Proposed fingerprint system
2nd	90.00%	90.00%
3rd	89.12%	88.77%
4th	89.75%	89.41%
5th	89.77%	86.17%

could not be 100% correct owing to the inherent variation of RSSI. As indicated by the results, the conventional fingerprint system produced a radio map of all the APs, and the proposed algorithm assumed that a sufficient number of AP RSSIs were collected during the moving processes. If a user does not have sufficient updates, the positioning performance is degraded because of insufficient movement. However, the same performance can be obtained when sufficient data are acquired via user movement, making the system very effective for generating a radio map. In order to verify the validity of the algorithm, we conducted additional experiments at the international university building. By performing this experiment, we obtained results that are almost similar to those.

V. CONCLUSION

To minimize the time-cost and workloads required to acquire Wi-Fi signals in automatic fingerprint systems, we proposed a fingerprint system that combines UDRM and MDLP-based RMF algorithms. The UDRM algorithm selects either the modified autoencoder or the modified GAN according to the indoor structure of each floor in order to improve the processing speed and accurately predict the radio map. The MDLP-based RMF algorithm copes with changes (e.g., positioning, installation, and AP removal), reducing the time-cost to acquire RSSIs by shifting the generation process of the radio map from the training phase to the positioning phase. The experimental results of the proposed automatic Wi-Fi fingerprint system based on unsupervised learning were as follows. For the proposed UDRM algorithm, the learning times of the 20 batches used for learning and prediction of the modified autoencoder and GAN-based radio map were 2 and 8 ms/step, respectively. When the accuracy of the real measured radio map was set to 100%, the concordance rates of the generated radio map obtained via the modified autoencoder and GAN were 90.4% and 92.7%, with the prediction errors of 9.6% and 7.3%, respectively. These errors were ideal for preventing overfitting caused by a high prediction accuracy of >95% during the training of the neural network. During the positioning phase, when the MDLP-based RMF algorithm was applied to the radio map of the modified autoencoder and the GAN, the update rates of the radio map were 52.9% and 60.5%, respectively. In addition, the positioning accuracy of the proposed fingerprint system was 88.59%, compared with the 89.66% accuracy obtained with the real measured radio map; this indicates that there were very few errors. Thus, the automatic Wi-Fi fingerprint system based on unsupervised

These experimental results validate the proposed MDLP-based RMF algorithm, which can be applied to APs on all floors, reducing their number by >50% in the experimental building. The proposed algorithm can be applied during the training phase and exhibits excellent performance for data discretization via the continuity of RSSIs. In particular, the effect of discretization using the MDLP is maximized in spaces where the APs are distributed evenly. Therefore, the proposed MDLP-based RMF algorithm is an optimization algorithm that prevents the rapid expansion of the radio map as the number of indoor AP increases, and it minimizes the degradation of the processing speed for positioning.

Table III presents the positioning results obtained for the automatic Wi-Fi fingerprint system based on unsupervised learning, which combined the UDRM and MDLP-based RMF algorithms. The positioning performance of the proposed system and the conventional fingerprint algorithm averaged 88.59% and 89.66%, respectively. This result is similar to those obtained for the proposed algorithm and the measurement-based algorithm because the real radio map

learning reduced the radio map by an average of 43.3%, even when the positioning accuracy was almost the same. The time-cost and workload involved in RSSI acquisition are the greatest problems encountered during fingerprinting. However, these can be significantly mitigated by automatically updating and creating radio maps.

REFERENCES

[1] V. Anagnostopoulos, M. Havlena, P. Kiefer, I. Giannopoulos, K. Schindler, and M. Raubal, "Gaze-informed location-based services," *Int. J. Geogr. Inf. Sci.*, vol. 31, no. 9, pp. 1770–1797, 2017.

[2] J.-H. Seong, E.-C. Choi, J.-S. Lee, and D.-H. Seo, "High-speed positioning and automatic updating technique using Wi-Fi and UWB in a ship," *Wireless Pers. Commun.*, vol. 94, no. 3, pp. 1105–1121, 2017.

[3] S. G. Nagarajan, P. Zhang, and I. Nevat, "Geo-spatial location estimation for Internet of Things (IoT) networks with one-way time-of-arrival via stochastic censoring," *IEEE Internet Things J.*, vol. 4, no. 1, pp. 205–214, Feb. 2017.

[4] J.-H. Seong and D.-H. Seo, "Environment adaptive localization method using Wi-Fi and Bluetooth low energy," *Wireless Pers. Commun.*, vol. 99, no. 2, pp. 765–778, 2018.

[5] O. Daniel, H. Wymeersch, and J. Nurmi, "Delay-accuracy tradeoff in opportunistic time-of-arrival localization," *IEEE Signal Process. Lett.*, vol. 25, no. 6, pp. 763–767, Jun. 2018.

[6] G. Chen, X. Meng, Y. Wang, Y. Zhang, P. Tian, and H. Yang, "Integrated WiFi/PDR/smartphone using an unscented Kalman filter algorithm for 3D indoor localization," *Sensors*, vol. 15, no. 9, pp. 24595–24614, 2015.

[7] C. Wu, Z. Yang, and Y. Liu, "Smartphones based crowdsourcing for indoor localization," *IEEE Trans. Mobile Comput.*, vol. 14, no. 2, pp. 444–457, Feb. 2015.

[8] E. Wang, M. Wang, Z. Meng, and X. Xu, "A study of WiFi-aided magnetic matching indoor positioning algorithm," *J. Comput. Commun.*, vol. 5, no. 3, pp. 91–101, 2017.

[9] S. H. Jung, B.-C. Moon, and D. Han, "Performance evaluation of radio map construction methods for Wi-Fi positioning systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 4, pp. 880–889, Apr. 2017.

[10] Q. Jiang, Y. Ma, K. Liu, and Z. Dou, "A probabilistic radio map construction scheme for crowdsourcing-based fingerprinting localization," *IEEE Sensors J.*, vol. 16, no. 10, pp. 3764–3774, May 2016.

[11] W. Zhao, S. Han, R. Q. Hu, W. Meng, and Z. Jia, "Crowdsourcing and multisource fusion-based fingerprint sensing in smartphone localization," *IEEE Sensors J.*, vol. 18, no. 8, pp. 3236–3247, Apr. 2018.

[12] I. Bisio *et al.*, "A trainingless WiFi fingerprint positioning approach over mobile devices," *IEEE Antennas Wireless Propag. Lett.*, vol. 13, pp. 832–835, May 2014.

[13] N. Hernández, M. Ocaña, J. M. Alonso, and E. Kim, "Continuous space estimation: Increasing WiFi-based indoor localization resolution without increasing the site-survey effort," *Sensors*, vol. 17, no. 1, p. 147, 2017.

[14] G. Caso and L. De Nardis, "Virtual and oriented WiFi fingerprinting indoor positioning based on multi-wall multi-floor propagation models," *Mobile Netw. Appl.*, vol. 22, no. 5, pp. 825–833, 2017.

[15] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-based fingerprinting for indoor localization: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 763–776, Jan. 2017.

[16] X. Jiang, Y. Chen, J. Liu, Y. Gu, and L. Hu, "FSELM: Fusion semi-supervised extreme learning machine for indoor localization with Wi-Fi and Bluetooth fingerprints," *Soft Comput.*, vol. 22, no. 11, pp. 3621–3635, 2018.

[17] Z. Turgut, S. ́Ustebay, G. Z. G. Aydin, and A. Sertbaş, "Deep learning in indoor localization using WiFi," in *Proc. Int. Telecommun. Conf.*, 2018, pp. 101–110.

[18] M. Mohammadi, A. Al-Fuqaha, M. Guizani, and J.-S. Oh, "Semisupervised deep reinforcement learning in support of IoT and smart city services," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 624–635, Apr. 2018.

[19] F. Yu *et al.*, "5G WiFi signal-based indoor localization system using cluster *k*-nearest neighbor algorithm," *Int. J. Distrib. Sensor Netw.*, vol. 10, no. 12, pp. 1–12, 2014.

[20] D. Li, B. Zhang, and C. Li, "A feature-scaling-based *k*-nearest neighbor algorithm for indoor positioning systems," *IEEE Internet Things J.*, vol. 3, no. 4, pp. 590–597, Aug. 2016.

[21] Y. Gu, Y. Chen, J. Liu, and X. Jiang, "Semi-supervised deep extreme learning machine for Wi-Fi based localization," *Neurocomputing*, vol. 166, pp. 282–293, Oct. 2015.

[22] K. S. Kim, S. Lee, and K. Huang, "A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting," *Big Data Analytics*, vol. 3, no. 1, 2018, Art. no. 4.

[23] J. Yu, C. Hong, Y. Rui, and D. Tao, "Multitask autoencoder model for recovering human poses," *IEEE Trans. Ind. Electron.*, vol. 65, no. 6, pp. 5060–5068, Jun. 2018.

[24] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[25] R. F. Ropero, S. Renooij, and L. C. van der Gaag, "Discretizing environmental data for learning Bayesian-network classifiers," *Ecol. Model.*, vol. 368, pp. 391–403, Jan. 2018.

[26] L.-Y. Wen, F. Min, and S.-Y. Wang, "A two-stage discretization algorithm based on information entropy," *Appl. Intell.*, vol. 47, no. 4, pp. 1169–1185, 2017.

[27] J.-H. Seong and D.-H. Seo, "Wi-Fi fingerprint using radio map model based on MDLP and Euclidean distance based on the Chi squared test," *Wireless Netw.*, vol. 25, no. 6, pp. 3019–3027, 2019.

[28] C. Zhang, O. Vinyals, R. Munos, and S. Bengio, "A study on overfitting in deep reinforcement learning," *arXiv preprint arXiv:1804.06893*, 2018.

[29] D. Li, C. Lei, Q. Jin, and M. Han, "Regularization in DQN for parameter-varying control learning tasks," in *Proc. Int. Symp. Neural Netw.*, 2019, pp. 35–44.

[30] S. Whiteson, B. Tanner, M. E. Taylor, and P. Stone, "Protecting against evaluation overfitting in empirical reinforcement learning," in *Proc. IEEE Symp. Adapt. Dyn. Program. Reinforcement Learn. (ADPRL)*, 2011, pp. 120–127.

[31] S. I. Konomi, T. Sasao, S. Hosio, and K. Sezaki, "Using ambient WiFi signals to find occupied and vacant houses in local communities," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 2, pp. 779–789, 2019.

[32] H. Cao, Y. Wang, J. Bi, and H. Qi, "An adaptive Bluetooth/Wi-Fi fingerprint positioning method based on Gaussian process regression and relative distance," *Sensors*, vol. 19, no. 12, p. 2784, 2019.

[33] H. Zou, Y. Zhou, J. Yang, and C. J. Spanos, "Unsupervised WiFi-enabled IoT device-user association for personalized location-based service," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 1238–1245, Feb. 2019.