Sensor-Aided Learning for Wi-Fi Positioning with Beacon Channel State Information

Jeongsik Choi, *Member, IEEE*

Abstract—Because each indoor site has its own radio propagation characteristics, a site survey process is essential to optimize a Wi-Fi ranging strategy for range-based positioning solutions. This study examines an unsupervised learning technique that autonomously learns an optimal ranging strategy for each site using Wi-Fi and sensor data accumulated while users access a positioning application. Using the collected sensor data, the device trajectory is regenerated, and a Wi-Fi ranging module is optimized to generate the shape of the estimated trajectory using Wi-Fi, similar to that obtained from sensors. In this process, the ranging module learns the way to identify the channel conditions from each Wi-Fi access point (AP) and produces ranging results accordingly. Furthermore, we collect the channel state information (CSI) from beacon frames to investigate the benefit of using CSI in addition to received signal strength measurements. With the CSI, the ranging module can identify diverse channel conditions from each AP and more accurately generate the reliability of each distance estimate to achieve accurate positioning results. The effectiveness of the proposed learning technique is verified using a real-time positioning application implemented on a PC platform [1].

Index Terms—Indoor positioning, channel state information, neural networks, unsupervised learning, sensor fusion.

I. INTRODUCTION

WITH the advent of various types of mobile devices, interest in location-based services has greatly increased in recent decades. Precise and reliable positioning is a key technology that enhances the end-user experience and creates new business opportunities. To achieve high-quality positioning results, many efforts have been introduced in the literature [2]–[4]. In particular, there is a great demand for positioning solutions that rely only on the built-in components of mobile devices and do not require the installation of additional infrastructure.

In this context, Wi-Fi has been widely used for locating mobile devices indoors. Because Wi-Fi access points (APs) are easy and cost-effective to deploy, many indoor sites already have dense APs that can be used for locating mobile devices. Without being associated with an AP, mobile devices can listen to beacon frames broadcast from nearby APs, thereby having received signal strength (RSS). For this reason, RSS has been used as a primary source for positioning solutions based on either range-based [5]–[8] or fingerprinting methods [9]–[12]. However, the RSS is a single representative value for the current channel state, which is insufficient for exploring additional

useful features observed from the wideband behavior of the channel.

For this reason, many studies have focused on channel state information (CSI), which is available with commodity Wi-Fi devices such as Intel IWL5300 [13] and the Atheros series [14]. Because CSI provides fine-grained information about the propagation channel, it can improve the positioning performance in various ways, such as identifying channel conditions [15]–[22], extracting features for fingerprinting methods [23]–[27], and estimating the angle of arrival (AoA) or time of flight (ToF) of the wireless signal [28]–[33]. Nevertheless, the CSI tools used in these studies capture the CSI of only high throughput (HT) frames defined in the IEEE 802.11n standard [34]. There are therefore restrictions; the transmitter must be configured to transmit HT frames, and it is challenging to collect CSI from multiple APs simultaneously.

To further improve the CSI measurement capability, new tools have been developed recently. The Nexmon CSI tool enables the collection of the CSI of Wi-Fi frames transmitted with an orthogonal frequency division multiplexing (OFDM) format defined in the 802.11a/g/n/ac standards [35], and the ESP32 CSI tool captures the CSI of 802.11g/n frames [36]. Intel has also developed a new CSI reporting feature for the latest Wi-Fi chipsets, such as the Wireless-AC9260/9560 and Wi-Fi 6 AX200/201 series [37]. Owing to the new CSI tools, the CSI of beacon frames is available. Without modifying any setting of existing APs and without being associated with an AP, CSI from nearby APs can be simultaneously collected in the same manner as the device collects RSS from beacon frames.

Although the CSI of beacon frames is available, there are still challenges: (i) The transmission bandwidth of beacon frames is 20 MHz, which may not be sufficient to obtain high-resolution multipath propagation profiles from transmitters. Therefore, the performance of existing CSI-based processes may be degraded. (ii) APs and devices are not synchronized for beacon transmission and reception; thus, distance estimation based on the ToF of beacon frames can produce incorrect results. (iii) The amount of available CSI is limited because the beacon frames are broadcast with a predefined interval (e.g., 100 ms). Thus, it is difficult to monitor subtle changes in the channels over a short period of time.

Despite these challenges, we believe that the CSI of beacon frames still plays an important role in improving the performance of RSS-based ranging because it compensates for the unstable fluctuation nature of RSS using relatively wideband information about the channel. In addition, the current propagation channel condition can be identified according to the

J. Choi is with the School of Electronics Engineering, Kyungpook National University, Daegu 41566, South Korea (e-mail: jeongsik.choi@knu.ac.kr).

This work was done when the author was with the Intel Labs, Intel Corporation, Santa Clara, CA 95054, USA.
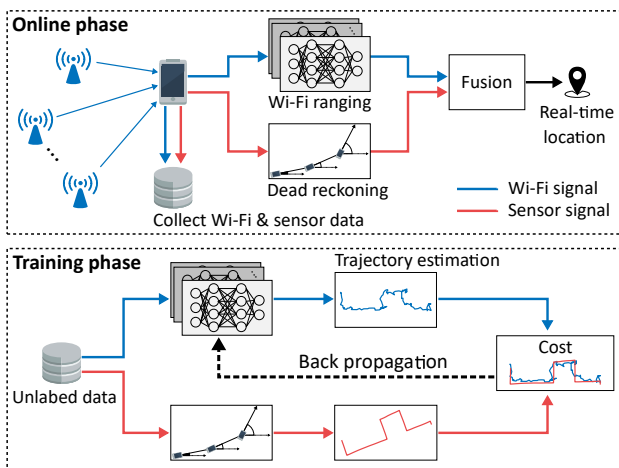
Fig. 1. Overview of the proposed sensor-aided learning framework.

frequency selectivity observed in the CSI, and the ranging performance can be improved by applying different ranging strategies depending on the identification result. One challenge is that the indoor propagation channel is too diverse and complex to be explained using a single model. Therefore, this paper primarily studies a machine learning approach to efficiently extract useful features from CSI.

This study is motivated by the fact that many positioning solutions exploit various types of sensors to estimate the device trajectory [7], [12], [38]–[40]. In particular, the pedestrian dead reckoning (PDR) method has been widely deployed for mobile navigation scenarios where pedestrians access location services. Using basic sensors incorporated in a mobile device, the PDR method estimates the trajectory of the device by integrating the separately obtained heading direction and moving distance of the device [41]–[44]. Because of its simplicity, the device trajectory obtained using the PDR method is used in this study, however, the learning technique proposed in this paper can be applied to other sensor fusion techniques that provide the device trajectory.

Fig. 1 illustrates an overview of the proposed learning framework, where a Wi-Fi ranging module is implemented using a neural network (NN). During the online phase, a positioning application integrates the Wi-Fi ranging and PDR outputs to obtain the real-time location of the device. At the same time, raw Wi-Fi and sensor data are stored in a database. In the training phase, the collected data are used to separately estimate the trajectory of the device, and a cost function is designed to measure the similarity between the two trajectories. The parameters in the ranging module are optimized in the direction of making the shape of the estimated trajectory using Wi-Fi similar to that provided by the PDR module. In this process, the ranging module autonomously learns a way to produce accurate ranging results from the input data. As the cost function does not include any ground truth information, training data can be collected whenever users access a positioning application, and the accuracy of the ranging module can be improved as data accumulate naturally.

The contributions of this study are summarized as follows:

i) We proposed a cost function that evaluates the similarity

of Wi-Fi and sensor trajectories to optimize a ranging strategy for each site. Because the cost function do not use the ground truth data, human intervention to perform a site survey process can be greatly reduced.

ii) We verified the advantage of using the CSI of beacon frames for Wi-Fi ranging. To this end, the CSI was collected from every beacon frame transmitted from nearby APs, and a convolutional NN (CNN) was designed to produce ranging results from the CSI measurements.

iii) To demonstrate the effectiveness of the proposed method, we implemented a real-time positioning application for a PC platform and conducted extensive experiments in a large-scale indoor office environment with 59 Wi-Fi APs installed on the ceiling. A real-time demo video is available online [1].

Organization: Related works are summarized in Section II, and the background of the CSI and PDR method is discussed in Section III. In Section IV, a Wi-Fi ranging module using an NN and positioning module is introduced. In Section V, a new cost function designed to train the ranging module with sensor information is presented. The experimental results are presented in Section VI, and the conclusions are presented in Section VII.

Notation: $\mathbf{A} \in \mathbb{R}^{N \times M}$ denotes an $N \times M$ real matrix (or vector), where $[\mathbf{A}]_{(n,m)}$ indicates the (n,m)-th element of the matrix. $\mathbf{I}_N \in \mathbb{R}^{N \times N}$ denotes the identity matrix and $\mathbf{A} = \mathrm{diag}(a_1, ..., a_N) \in \mathbb{R}^{N \times N}$ represents the diagonal matrix with diagonal elements $a_1, ..., a_N$. For a vector $\mathbf{a} \in \mathbb{R}^{N \times 1}$, we define $\|\mathbf{a}\| = \sqrt{\mathbf{a}^T \mathbf{a}}$. The transpose, inverse, and expectation operators are denoted by $(\cdot)^T$, $(\cdot)^{-1}$, and $\mathbb{E}[\cdot]$, respectively.

II. RELATED WORKS

We recommend referring to the comprehensive survey papers mentioned in the Introduction for numerous positioning solutions proposed in the literature. In this section, we summarize selected papers that are relevant to this study.

Channel condition identification using CSI: Because signal propagation characteristics vary widely depending on the presence of a direct path, identifying line-of-sight (LOS) and non-line-of-sight (NLOS) conditions is essential for improving the ranging performance using either RSS [22], [45], [46] or ToF [47] measurements. When a strong LOS path exists in a multipath propagation channel, the frequency response is flat over the wideband or the channel gain is stationary over time. Based on these observations, handcrafted features have been proposed to identify channel conditions, such as space-frequency correlation [15], Rician K-factor [16], skewness of dominant path power or kurtosis of frequency variation [17], and the variation in the phase of CSI [18]. In addition, various machine learning architectures have been applied to extract features from CSI efficiently, such as a recurrent NN (RNN) [19] and a CNN [20]–[22].

Wi-Fi ranging using CSI: Many approaches have been proposed to improve the ranging performance using CSI. FILA calculates a representative value, called effective CSI, by filtering out multipath components, and estimates the distance from an AP using this value [48]. To further improve the accuracy of

FILA, the automatic gain control process of the Wi-Fi chipset was considered in the calculation of the effective CSI [49], [50]. Similarly, the LOS component can be extracted from CSI to estimate the distance from an AP [29], [51]. In particular, CUPID applies different path loss models depending on the ratio between the power of the LOS component and the RSS to estimate the distance [29]. In addition, the error sources that contaminate CSI are mathematically analyzed to improve the ToF-based ranging performance [33], and an extended Kalman filter (EKF)-based method was introduced to consider a series of RSS and CSI measurements over time [52]. Machine learning approaches have also been introduced to improve the performance of time-based ranging [22], [53], [54].

Wi-Fi positioning using CSI: As summarized above, CSI has been used to improve the ranging performance, and thus, accurate positioning results can be obtained accordingly. In addition to range-based approaches, CSI can be directly used to locate devices using the fingerprinting method [23], [26], [27]. These approaches first process the raw CSI measurements, prepare a radio map, and estimate the position of the device. Moreover, various machine learning approaches have been applied to process CSI and extract features efficiently [24], [25], [55]. In particular, CNN architectures have been used to extract features from CSI images [24], [25], and an autoencoder architecture was applied in [55]. All machine learning approaches summarized in this section are based on a supervised learning technique, which requires the collection of labeled training data for each indoor environment.

III. BACKGROUND

A. Channel State Information

Under multipath propagation environments, each OFDM sub-carrier experiences a unique distortion. Thus, the Wi-Fi receiver performs a channel sounding procedure to measure the channel coefficient of each sub-carrier, called CSI.

CSI is related to the channel impulse response, which is expressed by

$$h(t) = \sum_{l=0}^{L-1} c_l \delta(t - \tau_l), \quad (1)$$

where $\delta(\cdot)$ represents the Dirac delta function, and L denotes the number of multipath components between the transmitter and receiver. The l-th multipath component is characterized by a complex channel coefficient and a time delay, which are denoted by c_l and τ_l, respectively. The baseband frequency response of sub-carrier n is expressed as

$$H(f_n) = \int_{-\infty}^{\infty} h(t) e^{-j2\pi f_n t} dt = \sum_{l=0}^{L-1} c_l e^{-j2\pi f_n \tau_l}, \quad (2)$$

where $f_n = n\Delta f$ is the frequency of sub-carrier n, and Δf represents the sub-carrier spacing. The measured CSI of sub-carrier n is expressed as

$$\hat{H}(f_n) = H(f_n) + \nu_n, \quad (3)$$

where ν_n denotes a complex-valued measurement noise for the sub-carrier n.

Because the main interest in this study is the CSI of beacon frames, parameters for the legacy OFDM format are used. The sub-carrier spacing is given by Δf = 20 MHz/64 = 312.5 kHz, and 48 and 4 sub-carriers are used for data and pilot transmissions, respectively. Therefore, the CSI tool can collect the complex frequency response for sub-carriers with an index $n \in \{-26, ..., -1, 1, ..., 26\}$. When multiple antennas are involved in beacon reception, multiple CSI sets are obtained.

B. Pedestrian Dead Reckoning

This work primarily focuses on a typical mobile navigation scenario in which users access a positioning application on their handheld device. Therefore, the PDR method is applied to estimate the trajectory of the device, which is used as a reference in the training phase. For other scenarios, such as a robot is deployed to perform site survey processes, another appropriate sensor fusion techniques can be applied to obtain the trajectory. We briefly summarize the PDR procedure, which is described in detail in [40].

As a first step, the orientation of the device with respect to a reference frame is obtained using accelerometer and gyroscope readings [56], [57]. The x-, y-, and z-axes of the reference frame point in the East, North, and up (ENU) directions relative to the position of the device on the Earth's surface. From the estimated orientation, the heading direction of the device on the x-y plane of the reference frame can be obtained by assuming that the device moves in the upward direction of the screen. We denote ϕ as the heading direction of the device. Note that a magnetometer is not used in this study because of the distortion of the magnetic field in an indoor environment. Therefore, ϕ refers to the heading relative to an arbitrary reference direction on the x-y plane, which is denoted as ϕ_{ref}.

The estimated orientation of the device is also used to transform the accelerometer readings measured with respect to the local frame into the reference frame. After the transformation, the z-axis acceleration component captures the vertical movement pattern of the device on the reference frame, which is generated when the user is walking. From the series of vertical acceleration values, a peak followed by a valley is considered as one step of the user, and the step length associated with the detected step can be obtained using a non-linear step length model as follows [41]:

$$\lambda = \alpha (a_{peak} - a_{valley})^{\frac{1}{4}}, \quad (4)$$

where a_{peak} and a_{valley} represent the peak and valley accelerations, respectively, and α is a constant coefficient.

According to the step detection results, the position of the device on the x-y plane is obtained by

$$\mathbf{p}(t) = \mathbf{p}(t-1) + \lambda(t) \mathbf{u}(\phi(t) + \phi_{ref}), \quad (5)$$

where $\mathbf{p}(t) = [x_{PDR}(t), y_{PDR}(t)]^T$ represents the x- and y-coordinates of the device obtained at sensor time t, and $\lambda(t)$ indicates the step length computed using equation (4) when a valid step is detected at time t and 0 if otherwise. In addition, $\phi(t)$ represents the heading direction obtained at time t, and

$\mathbf{u}(\phi) = [\cos\phi, \sin\phi]^T$ is the moving direction on the x-y plane corresponding to heading direction ϕ. In general, the PDR trajectory obtained from equation (5) differs from the true trajectory unless the initial position (e.g., $\mathbf{p}(0)$) and the reference direction are perfectly given.

IV. RANGING AND POSITIONING USING BEACON CSI

A. Assumptions

We consider a positioning scenario where N Wi-Fi APs are installed on the x-y plane of the reference frame. At each time step, the device scans all the Wi-Fi channels used by the APs in the vicinity. The device activates $A(\geq 1)$ receive antennas for beacon reception, and thus, A sets of RSS and CSI measurements are available from a single received beacon frame.

Within a single channel scanning procedure, the device can capture multiple beacon frames from each AP by increasing the probing time for each Wi-Fi channel. To address this, we denote $B(\geq 1)$ as the number of received beacon frames required for the ranging procedure. When the device receives fewer than B beacon frames from a certain AP, the ranging results from that AP are not available. Among the many APs with available ranging results, the positioning module selects up to $M(\leq N)$ APs to estimate the position of the device.

We denote $\mathbf{z} = [x, y]^T$ as the position of the device and $\mathbf{z}_n = [x_n, y_n]^T$ as the position of the n-th AP ($1 \leq n \leq N$) on the x-y plane. Depending on the position of the device, the selection of nearby M APs for positioning can be different. To address this, we denote $n_m^{(k)}$ as the index of the m-th selected AP at time step k ($1 \leq m \leq M$), and define $\mathbf{z}_m^{(k)} \triangleq \mathbf{z}_{n_m^{(k)}}$ to refer to the position of the m-th selected AP at time step k. We assume that the coordinates of all APs are known through a one-time manual effort or an automated method introduced in [58].

B. CNN-Based Ranging with Beacon CSI

To estimate distance and the reliability of distance estimate, which is expressed in terms of standard deviation, we design a ranging module in this section. The input and output relationship of the ranging module is expressed as a parametric function as follows:

$$\mathcal{R}(\mathcal{X}; \Theta) = \begin{bmatrix} \hat{d} \\ \hat{s} \end{bmatrix}, \qquad (6)$$

where $\mathcal{X}$ represents the list of measurements obtained from the received beacon frames, and Θ is the set of all trainable parameters in the module. The two outputs denoted by $\hat{d}$ and $\hat{s}$ are the distance estimate and its standard deviation, respectively. Note that the ranging module is used to obtain the ranging results for every AP.

Fig. 2 illustrates the amplitude of the CSI obtained from 20 consecutive beacon frames transmitted from an AP. Two receive antennas were used in this experiment. Fig. 2(a) and (c) show the results when the device is stationary under LOS and NLOS conditions, respectively. The coherence bandwidth of the channel under the LOS condition is wider than that under the NLOS condition. When the device is moving, the
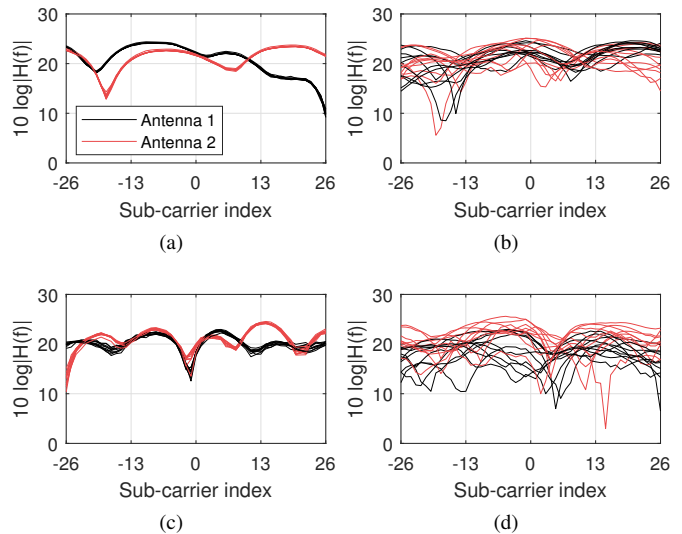

Fig. 2. Amplitude of CSI obtained from 20 beacon frames from an AP for the following scenarios: (a) The device is stationary under a LOS condition, (b) the device is moving under a LOS condition, (c) the device is stationary under an NLOS condition, and (d) the device is moving under an NLOS condition.
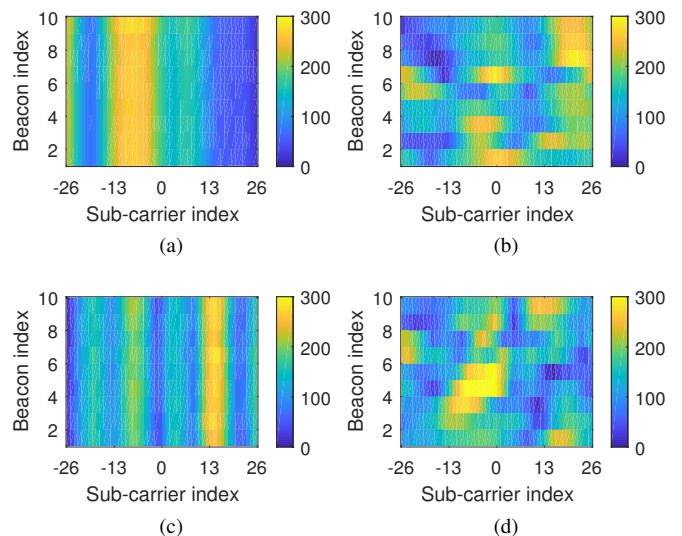

Fig. 3. Examples of one channel CSI input image constructed under: (a) A LOS condition when the device is stationary, (b) a LOS condition when the device is moving, (c) an NLOS condition when the device is stationary, and (d) an NLOS condition when the device is moving.

CSI fluctuates widely over time, as shown in Fig. 2(b) and (d). Nevertheless, the coherence bandwidth of the CSI of each beacon frame is relatively wide under the LOS condition.

To produce ranging results from the CSI and RSS measurements, we can include all information in the input layer as

$$\mathcal{X} = \{\mathcal{X}_{CSI}, \mathcal{X}_{RSS}\}, \qquad (7)$$

where $\mathcal{X}_{CSI}$ represents the input data constructed using the CSI. For simplicity, we only consider the amplitude of CSI and construct $\mathcal{X}_{CSI}$ as a two-dimensional image with A channels, where each channel of the image is generated from B received CSI using an antenna. Each channel is expressed as a two-dimensional matrix in $\mathbb{R}^{B \times 52}$, where each row consists of
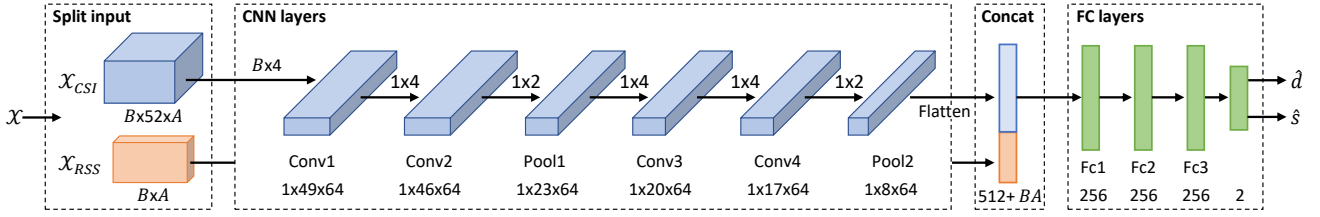
Fig. 4. The ranging module implemented using convolutional neural networks. This module is used to obtain ranging results for every AP.

the amplitude of $\hat{H}(f_n)$ for $n \in \{-26, ..., -1, 1, ..., 26\}$. Furthermore, $\mathcal{X}_{RSS}$ represents a matrix in $\mathbb{R}^{B \times A}$ with RSS measurements from B beacon frames using A antennas.

Fig. 3 shows one channel of the CSI input image. Fig. 3(a) and (c) depict the results when the user is stationary under LOS and NLOS conditions, respectively. Because the measured CSI is stationary over time, a few vertical stripes can be seen in both figures and each strip in Fig. 3(a) is wider than each strip in Fig. 3(c). By contrast, Fig. 3(b) and (d) show the CSI input image for a scenario in which the user is moving under LOS and NLOS conditions, respectively. Although the CSI fluctuates widely, each row of the input image shows different frequency selectivity for the two scenarios.

Fig. 4 illustrates the proposed ranging architecture. The CNN layers extract features from the A channels of the CSI input image. Because the number of collectible beacon frames from an AP is small in practice (e.g., $B \leq 10$), we use a $B \times 4$ sized kernel for the first convolution layer. The output of the first convolution layer becomes a one-dimensional vector, and we apply three more convolution layers with 1×4 sized kernels. In addition, 64 filters are used for all the convolution layers, and the 1×2 max-pooling layers are applied to every two convolution layers. The output of the CNN layers is concatenated with the RSS input after the flatten operation. Finally, three fully-connected (FC) layers, each with 256 hidden nodes, are applied to the concatenated layer.

The distance and standard deviation outputs are produced from the last FC layer as follows:

$$\begin{aligned} \hat{d} &= \bar{d}\sigma(\mathbf{W}_d\mathcal{X}_{-1} + b_d), \\ \hat{s} &= \bar{s}\sigma(\mathbf{W}_s\mathcal{X}_{-1} + b_s), \end{aligned} \quad (8)$$

where $\mathcal{X}_{-1} \in \mathbb{R}^{256 \times 1}$ represents the activation vector of the last hidden layer corresponding to the input layer $\mathcal{X}$. Matrices $\mathbf{W}_d$ and $\mathbf{W}_s \in \mathbb{R}^{256 \times 1}$ represent the weights between the last hidden layer and each output, and scalars b_d and b_s represent biases. Throughout the NN architecture, the rectified linear unit (ReLU) is used for activation of every layer except for the output layer. For output activation, the sigmoid function $\sigma(\cdot)$ is applied to specify the upper bounds of the distance and standard deviation outputs, denoted by $\bar{d}$ and $\bar{s}$, respectively.

C. RSS Offset Compensation

The ranging module designed in the previous subsection is used to obtain ranging results for every AP. However, each AP may have different radio frequency characteristics, such as antenna gain and transmission power. To address this, we introduce a vector of trainable parameters $\mathbf{o} = [o_1, ..., o_N]^T$,

where the n-th element refers to the RSS offset of the n-th AP. To obtain ranging results from the n-th AP, we process the raw measurements and prepare the input data for the ranging module as

$$\mathcal{X}_n = \{\mathcal{X}_{CSI,n}, \mathcal{X}_{RSS,n} + o_n\}, \quad (9)$$

where $\mathcal{X}_{CSI,n}$ and $\mathcal{X}_{RSS,n}$ represent the raw CSI and RSS inputs for the n-th AP, respectively. By feeding $\mathcal{X}_n$ into the ranging module, we can obtain the ranging results from the n-th AP as $\mathcal{R}(\mathcal{X}_n; \Theta) = [\hat{d}_n, \hat{s}_n]^T$. Note that the offset of each AP is optimized during the training phase as errors propagate backward to the input layer of the ranging module.

D. Positioning with Wi-Fi Ranging Results

The ranging results from multiple APs in the vicinity are used to estimate the position of the device. In this work, we apply an EKF-based positioning method that achieves accurate results by taking into account a series of measurements over time [8]. The unknown state is assumed to be the position of the device, and the EKF procedure is summarized as follows:

1) Initialization: We initialize the position estimate of the device as the center of M nearby APs as

$$\hat{\mathbf{z}}^{(0)} = \frac{1}{M}\sum_{m=1}^{M}\mathbf{z}_m^{(0)}, \quad (10)$$

and the covariance matrix of the initial position estimate as

$$\mathbf{P}^{(0)} = \text{diag}\left(s_x^2, s_y^2\right), \quad (11)$$

where s_x and s_y represent the standard deviations of the initial x- and y-coordinate estimates, respectively.

2) State Prediction: At time step $k(\geq 1)$, the EKF predicts the current state from the previous state based on the state transition model given by

$$\mathbf{z}^{(k)} = \mathbf{z}^{(k-1)} + v\Delta t\mathbf{u}(\Phi), \quad (12)$$

where v represents the moving speed of the device, which can be assumed to be a constant, and Δt is the time interval between the two time steps. Without external information, the moving direction Φ is assumed to be a uniform random variable in $[0, 2\pi]$. Using the state transition model, the predicted state and its covariance matrix are obtained as

$$\begin{aligned} \hat{\mathbf{z}}^{(k|k-1)} &= \hat{\mathbf{z}}^{(k-1)}, \\ \mathbf{P}^{(k|k-1)} &= \mathbf{P}^{(k-1)} + \mathbf{Q}^{(k)}, \end{aligned} \quad (13)$$

where $\mathbf{Q}^{(k)} = (v\Delta t)^2\mathbb{E}[\mathbf{u}(\Phi)\mathbf{u}(\Phi)^T] = \frac{1}{2}(v\Delta t)^2\mathbf{I}_2$.
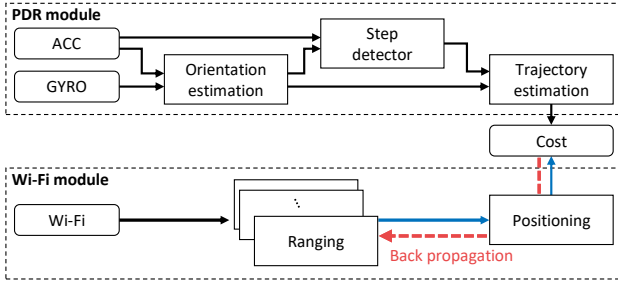
Fig. 5. Overview of the proposed sensor-aided learning technique.

3) State Update: The predicted state is then corrected using the ranging results. The measurement model is expressed as

$$\mathbf{d}^{(k)} = \mathbf{h}^{(k)}(\mathbf{z}) + \boldsymbol{\omega}^{(k)} = \begin{bmatrix} \|\mathbf{z} - \mathbf{z}_1^{(k)}\| \\ \vdots \\ \|\mathbf{z} - \mathbf{z}_M^{(k)}\| \end{bmatrix} + \boldsymbol{\omega}^{(k)}, \qquad (14)$$

where $\mathbf{d}^{(k)} = [\hat{d}_1^{(k)}, ..., \hat{d}_M^{(k)}]^T$ is a vector of distance estimates from M selected APs at time step k. In addition, $\boldsymbol{\omega}^{(k)} = [\omega_1^{(k)}, ..., \omega_M^{(k)}]^T$ indicates a vector of ranging errors, which is modeled as a zero-mean Gaussian random vector with the covariance matrix of

$$\boldsymbol{\Lambda}^{(k)} = \mathbb{E}\left[\boldsymbol{\omega}^{(k)}(\boldsymbol{\omega}^{(k)})^T\right] = \text{diag}\left((\hat{s}_1^{(k)})^2, ..., (\hat{s}_M^{(k)})^2\right). \quad (15)$$

The innovation of the EKF and its covariance matrix are computed as

$$\mathbf{e}^{(k)} = \mathbf{d}^{(k)} - \mathbf{h}^{(k)}(\hat{\mathbf{z}}^{(k|k-1)}),$$
$$\mathbf{S}^{(k)} = \mathbf{H}^{(k)}\mathbf{P}^{(k|k-1)}(\mathbf{H}^{(k)})^T + \boldsymbol{\Lambda}^{(k)}. \qquad (16)$$

Here, $\mathbf{H}^{(k)} \in \mathbb{R}^{M \times 2}$ represents the Jacobian matrix of $\mathbf{h}^{(k)}(\cdot)$, which is defined as

$$\mathbf{H}^{(k)} \triangleq \frac{\partial \mathbf{h}^{(k)}(\mathbf{z})}{\partial \mathbf{z}}\bigg|_{\mathbf{z}=\hat{\mathbf{z}}^{(k|k-1)}}. \qquad (17)$$

The Kalman gain is computed as

$$\mathbf{G}^{(k)} = \mathbf{P}^{(k|k-1)}(\mathbf{H}^{(k)})^T(\mathbf{S}^{(k)})^{-1}, \qquad (18)$$

and the updated state and its covariance matrix are given by

$$\hat{\mathbf{z}}^{(k)} = \hat{\mathbf{z}}^{(k|k-1)} + \mathbf{G}^{(k)}\mathbf{e}^{(k)},$$
$$\mathbf{P}^{(k)} = \left(\mathbf{I}_2 - \mathbf{G}^{(k)}\mathbf{H}^{(k)}\right)\mathbf{P}^{(k|k-1)}, \qquad (19)$$

respectively. Note that $\hat{\mathbf{z}}^{(k)}$ refers to the estimated position of the device at time step k, and the estimated trajectory using Wi-Fi varies widely depending on each output of the ranging module as the distance and standard deviation estimates are used in the EKF procedure to compute the innovation $\mathbf{e}^{(k)}$ and the covariance matrix $\boldsymbol{\Lambda}^{(k)}$, respectively.

V. SENSOR-AIDED LEARNING TECHNIQUE

Fig. 5 provides an overview of the sensor-aided learning technique, where the blue arrows indicate that the output depends on the trainable parameters. For the training, the device trajectory is separately estimated using Wi-Fi and PDR modules. Then, a cost function that compares the shapes of
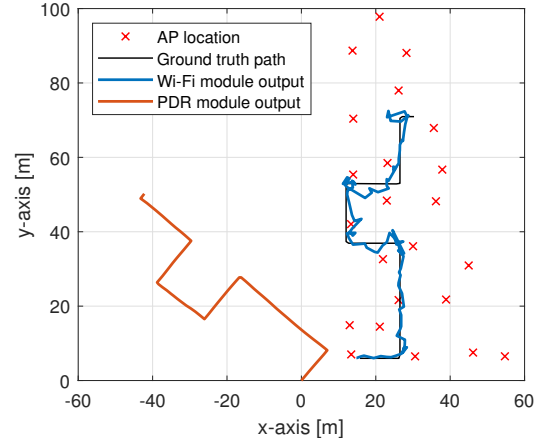

Fig. 6. Example of trajectory estimation results using the Wi-Fi and PDR modules.

the two trajectories is defined. Because the sampling rate of the sensors is much faster than the frequency of the Wi-Fi ranging procedure, we first synchronize the two trajectories by defining $\mathbf{p}^{(k)} \triangleq \mathbf{p}(t_k)$, where t_k indicates the sensor time when the k-th Wi-Fi ranging procedure is performed.

In this section, we design the cost function of a single training dataset collected during K time steps. In the case that multiple datasets are available, the same process can be applied to each dataset, and the overall cost is obtained by simply taking the average of all the costs. For ease of exposition, we define $\mathcal{Z} \triangleq \{\hat{\mathbf{z}}^{(k)}\}_{k=1}^{K}$ as the sequence of the estimated position of the device using the Wi-Fi module and $\mathcal{P} \triangleq \{\mathbf{p}^{(k)}\}_{k=1}^{K}$ as the sequence of the PDR output. In addition, all summation operations used in this section represent the summation from $k=1$ to K.

Fig. 6 shows the trajectory estimation results. Because the ranging module produces errors, the estimated trajectory using the Wi-Fi module fluctuates widely. Nevertheless, the estimated trajectory follows the ground truth path shown in the figure as the positioning module exploits the true location of the APs. On the other hand, the PDR module produces a smoother trajectory because built-in sensors are less affected by the external environment, and the shape of the trajectory is similar to that of the ground truth path. However, the PDR trajectory starts at an arbitrary position, and the direction of the movement is not aligned with the ground truth path as ϕ_{ref} is randomly determined. In this figure, the PDR trajectory is generated with the assumption of $\phi_{ref} = 0$, which may differ from the true value.

To compare the similarity of the Wi-Fi and PDR trajectories, we transform the PDR trajectory with a rotation angle and an offset as follows:

$$\tilde{\mathbf{p}}^{(k)} = \mathbf{R}(\varphi)\mathbf{p}^{(k)} + \boldsymbol{\Omega}, \ 1 \le k \le K, \qquad (20)$$

where $\boldsymbol{\Omega} \in \mathbb{R}^{2 \times 1}$ indicates the x- and y-coordinate offsets, and φ is the rotation angle. In addition, $\mathbf{R}(\varphi) \in \mathbb{R}^{2 \times 2}$ represents the rotation matrix on the x-y plane, which is defined as

$$\mathbf{R}(\varphi) \triangleq \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} = \cos\varphi \mathbf{I}_2 + \sin\varphi \tilde{\mathbf{I}}_2, \qquad (21)$$

where $\tilde{\mathbf{I}}_2$ is a 2×2 anti-diagonal matrix with $[\tilde{\mathbf{I}}_2]_{(1,2)} = -1$ and $[\tilde{\mathbf{I}}_2]_{(2,1)} = 1$. An optimal transformation can be obtained to minimize the sum squared error between the transformed PDR output and the estimated trajectory using the Wi-Fi module, which is given by

$$\mathcal{L}\left(\mathcal{Z}, \mathcal{P}; \varphi, \mathbf{\Omega}\right) = \sum_k \|\tilde{\mathbf{p}}^{(k)} - \hat{\mathbf{z}}^{(k)}\|^2. \quad (22)$$

We use the following lemma to obtain an optimal transformation.

Lemma 1: An optimal rotation angle and offset that minimize the cost function in equation (22) are derived as

$$\varphi^* = \pi + \arctan\frac{\Gamma}{\tilde{\Gamma}},$$

$$\mathbf{\Omega}^* = \frac{\sum_k \hat{\mathbf{z}}^{(k)} - \mathbf{R}(\varphi^*) \sum_k \mathbf{p}^{(k)}}{K}, \quad (23)$$

respectively, where Γ and $\tilde{\Gamma}$ are related to $\mathcal{Z}$ and $\mathcal{P}$ as follows:

$$\Gamma = \frac{(\sum_k \hat{\mathbf{z}}^{(k)})^T (\sum_k \mathbf{p}^{(k)})}{K} - \sum_k (\hat{\mathbf{z}}^{(k)})^T \mathbf{p}^{(k)},$$

$$\tilde{\Gamma} = \frac{(\sum_k \hat{\mathbf{z}}^{(k)})^T \tilde{\mathbf{I}}_2 (\sum_k \mathbf{p}^{(k)})}{K} - \sum_k (\hat{\mathbf{z}}^{(k)})^T \tilde{\mathbf{I}}_2 \mathbf{p}^{(k)}. \quad (24)$$

And the error after the transformation is computed as

$$\mathcal{L}(\mathcal{Z}, \mathcal{P}; \varphi^*, \mathbf{\Omega}^*) = \sum_k \|\hat{\mathbf{z}}^{(k)}\|^2 + \sum_k \|\mathbf{p}^{(k)}\|^2$$

$$+ \frac{\|\sum_k \hat{\mathbf{z}}^{(k)}\| + \|\sum_k \mathbf{p}^{(k)}\|}{K} - 2\sqrt{\Gamma^2 + \tilde{\Gamma}^2}. \quad (25)$$

Proof. See Appendix A. □

This lemma explains that if two trajectories are given, one can be transformed close to the other to compute the cost that measures the similarity of the shapes of the two trajectories. Using this cost, we can train the ranging module so that the shape of the Wi-Fi trajectory becomes similar to that of the PDR trajectory, which shape is almost the same as that of the ground truth path. Based on this observation, we can define a cost function that utilizes sensor information as

$$\mathcal{L}_{sen}(\mathcal{Z}, \mathcal{P}) = \mathcal{L}(\mathcal{Z}, \mathcal{P}; \varphi^*, \mathbf{\Omega}^*), \quad (26)$$

which is computed using equation (25).

In addition to the above cost function, we can include the geometric cost function introduced in [8]. This cost function depends only on the Wi-Fi trajectory as

$$\mathcal{L}_{geo}(\mathcal{Z}) = \sum_{k=1}^{K} \sum_{m=1}^{M} \left(\|\hat{\mathbf{z}}^{(k)} - \mathbf{z}_m^{(k)}\| - \hat{d}_m^{(k)} \right)^2. \quad (27)$$

By combining the two cost functions, we obtain a unified cost function for training as

$$\mathcal{L}(\mathcal{Z}, \mathcal{P}) = \mu_1 \mathcal{L}_{sen}(\mathcal{Z}, \mathcal{P}) + \mu_2 \mathcal{L}_{geo}(\mathcal{Z}), \quad (28)$$

where μ_1 and μ_2 are non-negative constants that balance the two cost functions. Because the unified cost function depends on $\mathcal{Z}$ and $\mathcal{P}$, where $\mathcal{P}$ is fixed as the output of the PDR module, the gradient of the unified cost function with respect to every element in $\mathcal{Z}$ can be easily computed, and these gradients propagate to the ranging module to optimize every trainable parameter.
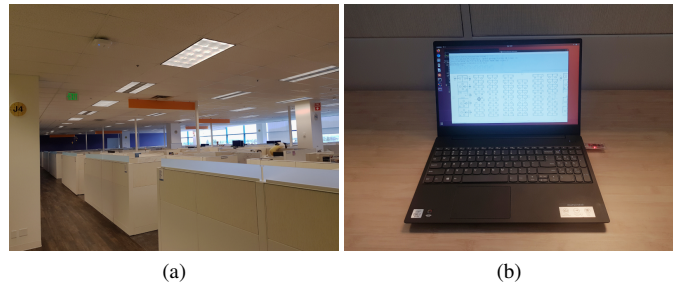

(a) (b)

Fig. 7. Experimental site and device. (a) Office environment with multiple Wi-Fi APs installed on the ceiling and (b) laptop equipped with Intel AX200 Wi-Fi chipset and an external USB sensor stick.

VI. EXPERIMENTAL RESULTS

A. Data Collection

We performed measurements in a practical indoor office environment with 59 Wi-Fi APs installed on the ceiling. A laptop equipped with an Intel Wi-Fi 6 AX200 chipset and running Ubuntu 18.04 operating system was used to collect the training data. We also implemented a real-time positioning application to collect the training data and demonstrate the proposed method. Fig. 7 shows pictures of the experiment site and the device with the real-time application running on the screen [1].

CSI data collection was performed using the CSI reporting tool mentioned in the Introduction. We believe that similar experiments can be performed using Nexmon or ESP32 CSI tools [35], [36], which can capture the CSI of legacy OFDM frames. To flexibly allocate the scan time for each channel, we configured the Wi-Fi interface to the monitor mode that captures all packet transmissions on a specific channel[1], and activated two receive antennas (i.e., $A = 2$). In addition, an external USB sensor stick from Bosch Sensortech was used to collect accelerometer and gyroscope readings at a sampling rate of 100 Hz [59]. The collected sensor data were used to estimate the trajectory of the device using the PDR method with a predefined step length coefficient of $\alpha = 0.55$.

The existing APs use three non-overlapping Wi-Fi channels for the 2.4 GHz frequency band, namely channels 1, 6, and 11. We therefore allocated 300 ms to each channel to collect beacon frames transmitted at that channel; thus, a single ranging and positioning procedure took less than 1 s. Furthermore, each AP used in this experiment site transmits four different service set identifiers (SSIDs) on the 2.4 GHz frequency band, and a beacon for each SSID is broadcast with a 100 ms interval. Therefore, it was possible to receive multiple beacon frames from each AP during a single ranging procedure. The positioning performance was evaluated with various choices of the number of beacon frames used in the ranging module (e.g., $B = 1, 2, 4$, and 8). However, the performance was almost the same regardless of B. Thus, we assumed that $B = 4$ throughout the experiment.

---

[1]Without using the monitor mode, the CSI of beacon frames can be collected by executing an active channel scanning procedure provided by the *iw* command. In this case, the Wi-Fi chipset monitors the activity of each channel for 100 ms.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2021.3138850
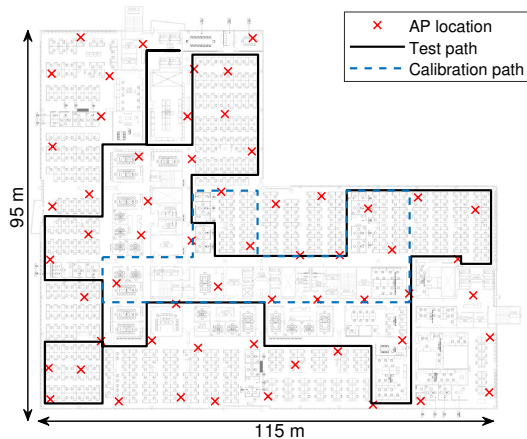
8

Fig. 8. Floor plan of the experiment site.

To simulate several users accessing a positioning application in practice, six persons participated in the experiment, and each person collected unlabeled data by naturally walking around the site for 10 minutes. The collected data were partitioned into multiple datasets, each corresponding to 100 time steps (i.e., 100 s). After this process, 36 datasets were generated, with 24 datasets used for training and the remaining 12 datasets used for validation purposes. In addition, we further collected calibration data by following the calibration path presented in Fig. 8 to optimize parameters in conventional ranging methods. However, the calibration data were not used in the proposed method, and thus, human intervention to perform such a manual site survey process can be greatly minimized. Finally, we collected test data by following the test path to evaluate the ranging and positioning performances. The total length of the test path is 615 m and it took about 10 minutes to collect the test data.

Throughout the experiment, we used up to $M = 5$ nearby APs in the positioning module. Note that the experiment site has dense APs, and thus, more than five APs were available for almost every time step. Therefore, we selected the five best APs depending on the measured RSS values for the validation and test scenarios. On the other hand, for the training scenario, each of 24 training datasets was used to generate 10 times more datasets by randomly selecting five APs among all active APs at each time step. Therefore, 240 training datasets were generated, each consisting of 500 ranging results (i.e., 100 time steps $\times$ 5 APs).

B. Conventional Ranging Models

For performance comparisons, we considered various ranging scenarios, as summarized in Table I. First, the path loss-based ranging model estimates the distance from an AP as [5]

$$\hat{d}_{PL} = d_0 10^{\frac{RSS(d_0) - RSS}{10\eta}}, \tag{29}$$

where RSS represents the average of all RSSs measured from B beacon frames using A antennas, $RSS(d_0)$ is the RSS at a reference distance $d_0 = 1$ m, and η is the path loss exponent.

TABLE I
WI-FI RANGING SCENARIOS

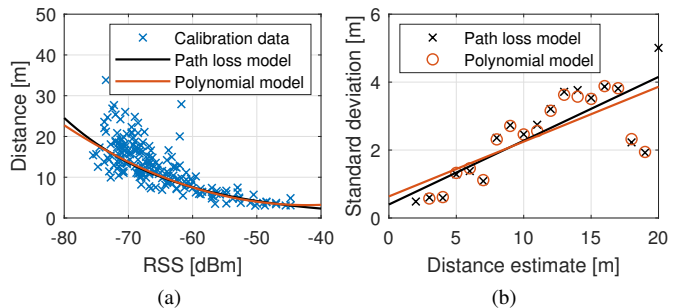| Ranging model	Source	Training Data
Path loss [5]	RSS	Calibration data
Polynomial [6]	RSS	Calibration data
CUPID [29]	RSS, CSI	Calibration data
FC (unsupervised) [8]	RSS	Unlabeled data
FC (sensor-aided)	RSS	Unlabeled data (w/ sensor)
CNN (sensor-aided)	RSS, CSI	Unlabeled data (w/ sensor)

Fig. 9. Calibration of path loss and polynomial parameters: (a) RSS versus distance, and (b) distance estimate versus standard deviation.

In addition, the distance from an AP can be estimated using a quadratic polynomial as [6]

$$\hat{d}_{poly} = g_2 RSS^2 + g_1 RSS + g_0, \tag{30}$$

where g_2, g_1, and g_0 represent the coefficients of the polynomial, which should be chosen appropriately.

The calibration data were used to optimize the parameters in these models. The parameters were selected to minimize the normalized mean squared error (NMSE) between the distance estimate $\hat{d}$ and the true distance d^*, which is defined as $NMSE = \mathbb{E}[((\hat{d} - d^*)/d^*)^2]$. The selected parameters for the path loss model are given by $RSS(d_0) = -25.8$ dBm and $\eta = 3.9$, and those for the polynomial model are given by $g_2 = 0.0138$, $g_1 = 1.1642$, and $g_0 = 27.7688$. The path loss and polynomial curves with the selected parameters are presented as solid lines in Fig. 9(a).

In addition, the standard deviation of each distance estimate is empirically obtained by taking the standard deviation of the ranging errors with respect to all data whose distance estimates are less than 1 m from the target distance estimate. Fig. 9(b) illustrates the relationship between the estimated distance and the empirically obtained standard deviation. A linear regression line was used to make a model $\hat{s}_{PL} = 0.1897\hat{d}_{PL} + 0.3672$ for the path loss and $\hat{s}_{poly} = 0.1622\hat{d}_{poly} + 0.6156$ for the polynomial-based ranging scenarios.

In the same way, we optimized the parameters for the CUPID model that exploits both RSS and CSI for the ranging procedure [29]. To this end, the energy of the direct path (EDP) was extracted from the CSI, and different path loss exponents were selected depending on the ratio between EDP and RSS to estimate the distance using the path loss model. The standard deviation for CUPID was modeled in the same way in the previous models.
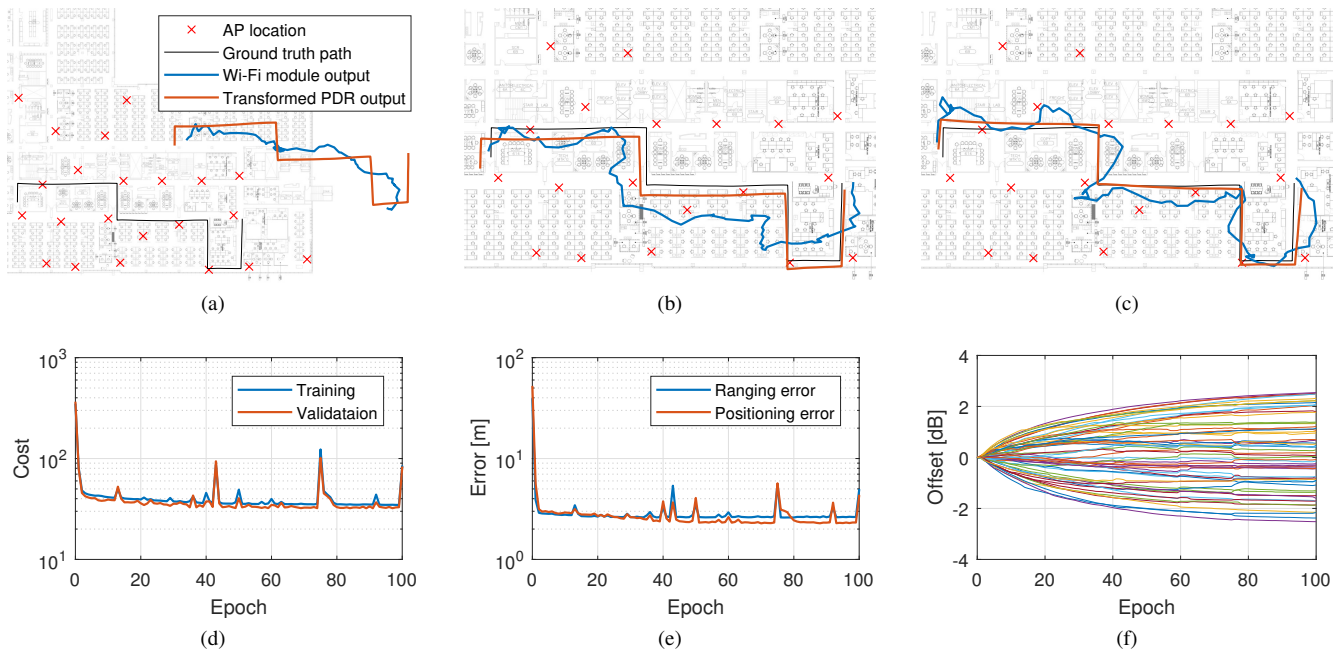
Fig. 10.   Training details of the sensor-aided learning: (a)-(c) Estimated trajectories at epoch 0, 5, and 50, respectively, (d) training and validation cost, (e) ranging and positioning error with respect to the test data, and (f) trained offset of each AP.

C. Training Phase

We consider two NN-based ranging scenarios in which the first scenario relies only on RSS measurements. For this case, we simply deployed FC layers that consist of an input layer of size $2B$ (i.e., RSS collected from B beacon frames using $A = 2$ receive antennas), followed by two hidden layers, each with 128 nodes. In addition, we evaluated the performance of the CNN-based ranging module using the CSI of beacon frames together with RSS measurements. The maximum distance estimate and standard deviation were assumed to be $\bar{d} = 100$ m and $\bar{s} = 10$ m for all NN-based ranging modules.

We trained the FC layers using the previous unsupervised learning technique proposed in [8] and the sensor-aided learning technique to verify the effectiveness of the proposed learning framework. Unfortunately, the previous unsupervised learning technique was unable to train the CNN-based ranging module because of an overfitting issue. Therefore, we trained the CNN-based ranging module using only the sensor-aided learning technique. We assumed that $(\mu_1, \mu_2) = (1, 1)$ for sensor-aided learning scenarios, and $(\mu_1, \mu_2) = (0, 1)$ for the previous unsupervised learning scenarios. The Adam optimizer with a learning rate of 0.001 was used for the training. In addition, we trained each ranging module with 10 different initializations and selected the best one with the minimum validation cost.

Fig. 10 illustrates the details of the sensor-aided learning procedure. The parameters in the ranging module are initialized randomly at the beginning. As a result, the Wi-Fi trajectory produces an incorrect result at epoch 0, as shown in Fig. 10(a). Accordingly, the PDR trajectory is transformed close to the Wi-Fi trajectory, and the cost is computed. After a few training epochs, the Wi-Fi trajectory closely approaches the test path because the ranging module produces accurate

ranging results, and the transformed PDR trajectory overlaps the test path, as shown in Fig. 10(b) and (c). Therefore, the cost computed using equation (26) can be considered to be equivalent to the error between the Wi-Fi trajectory and the test path. Fig. 10(d) illustrates that the costs with respect to the training and validation data decrease with the epoch. As a result, the ranging and positioning errors with respect to the test data also decrease with epoch, as shown in Fig. 10(e). In addition, the offsets of all 59 APs are optimized during the training phase, as shown in Fig. 10(f).

Fig. 11(a) depicts the cumulative density function (CDF) of the ranging error. Because every scenario primarily relies on RSS measurements for the ranging, there was no meaningful difference observed in the ranging performance. However, an interesting result is observed in Fig. 11(b). This figure shows the relationship between the distance and standard deviation outputs for selected ranging scenarios. The relationship between the two outputs for the path loss-based ranging scenario is presented as a straight line, as we modeled it using a linear regression line. Similarly, the FC-based ranging module produces always same standard deviation output for a specific distance estimate.

However, the CNN-based ranging module produces various standard deviation outputs for the same distance estimate. One possible scenario is that the ranging module identifies the channel condition from the CSI of beacon frames and produces the outputs in different ways depending on the identified channel condition. For instance, if input data is highly likely to be observed in an NLOS condition, the ranging module may produce a high standard deviation output for a distance estimate to make the positioning module less reliant on this distance estimate. By doing so, an accurate positioning result can be achieved. Fig. 11(c) depicts the CDF of the positioning

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2021.3138850
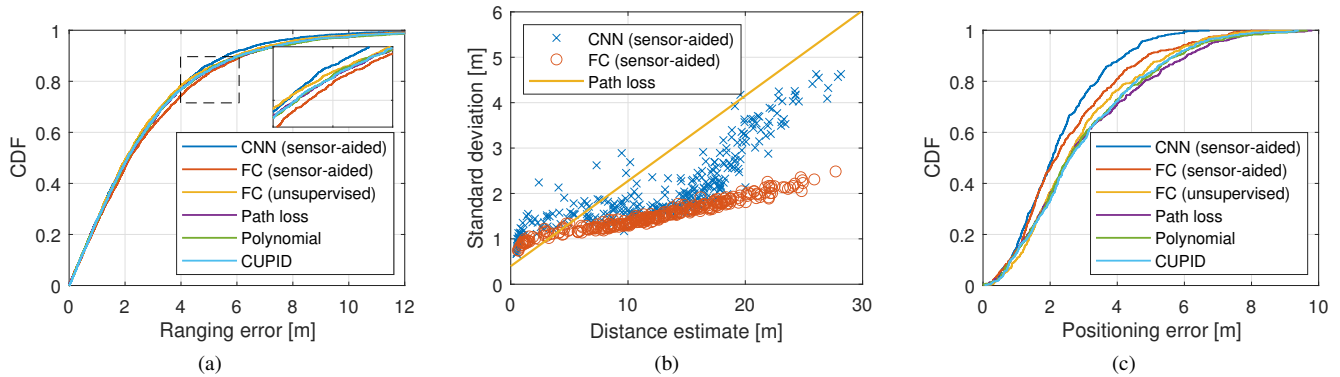
10

Fig. 11. Training results: (a) CDF of ranging error, (b) relationship between distance and standard deviation outputs, and (c) CDF of positioning error when only Wi-Fi ranging is used in the positioning module.

TABLE II
RANGING AND POSITIONING* PERFORMANCES

	Ranging model	MAE [m]	RMSE [m]	90%-tile [m]
Ranging	Path loss	3.019	4.175	6.746
	Polynomial	3.011	4.203	6.578
	CUPID	2.971	4.111	6.659
	FC (unsupervised)	2.717	3.813	6.127
	FC (sensor-aided)	2.874	3.882	6.206
	CNN (sensor-aided)	2.657	3.602	5.628
Positioning	Path loss	2.887	3.442	5.436
	Polynomial	2.851	3.400	5.300
	CUPID	2.880	3.411	5.338
	FC (unsupervised)	2.967	3.468	5.436
	FC (sensor-aided)	2.595	3.079	5.008
	CNN (sensor-aided)	2.300	2.626	3.903

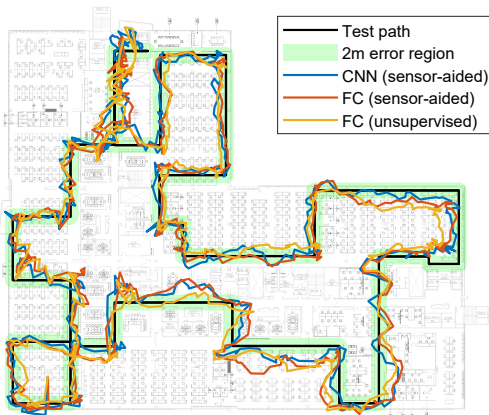*Positioning performance is based only on Wi-Fi ranging.

Fig. 12. Estimated trajectory using Wi-Fi ranging only.

error. Although all ranging scenarios yielded similar ranging performance, the CNN-based ranging module outperforms the existing methods by providing more accurate standard deviation output for each distance estimate.

Table II summarizes the ranging and positioning performances for every ranging scenario. Note that the positioning performance described in this section indicates the result using

Wi-Fi ranging only. The positioning performance using Wi-Fi and sensors will be evaluated in the next subsection. The performance metrics are the mean absolute error (MAE), root mean squared error (RMSE), and 90th percentile error. The NN-based ranging scenarios outperform the ranging and positioning performances of the conventional ranging scenarios, even through the ranging modules were trained using the unlabeled training data.

In addition, the positioning performances of the two FC-based ranging scenarios show the effectiveness of the proposed sensor-aided learning technique. As sensor data was used in the training phase, every trainable parameter could be optimized in a more reliable manner, and more accurate positioning results were achieved, compared to the previous unsupervised learning technique that does not exploit sensor data [8]. Moreover, the proposed learning technique could train the CNN-based ranging module with a much larger size of trainable parameters and more complex input data compared to the FC layers, whereas the previous unsupervised learning technique failed to train such a large network due to overfitting. According to Table II, the CSI of beacon frames can improve the average positioning accuracy by 30–60 cm and the 90th percentile accuracy by 1.1–1.5 m from RSS-based ranging scenarios. This is a meaningful gain if we remind that the CSI of beacon frames is collected in the exactly same manner as the device collects RSS from beacon frames.

Finally, Fig. 12 shows the estimated trajectories using Wi-Fi ranging only. Because of ranging errors, the estimated trajectory for every ranging scenario fluctuates widely. Therefore, we present the trajectories for selected scenario for a clear figure. The green area represents the 2 m error region, indicating that any points in this area are less than 2 m from the closest test path. The estimated trajectory for the CNN-based ranging scenario produced the best performance in terms of every performance metric.

D. Online Phase

Once training is completed, the Wi-Fi ranging module produces distance and standard deviation estimates for all nearby APs during the online phase. When built-in sensors are temporally unavailable or a positioning application does not
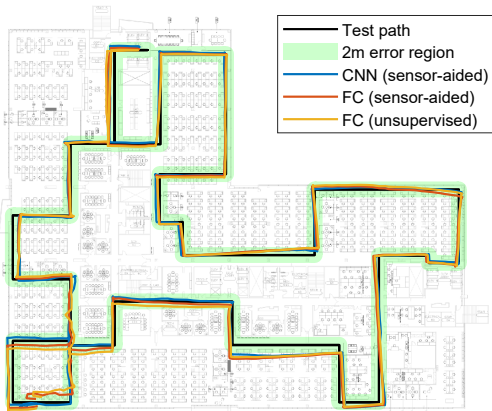
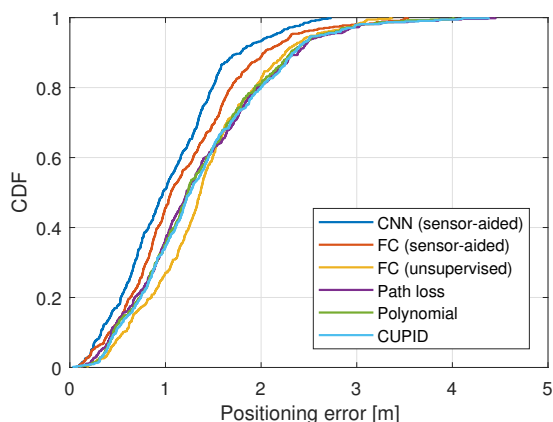Fig. 13. Estimated trajectory using Wi-Fi and sensors.

Fig. 14. CDF of positioning error using Wi-Fi and sensors.

utilize sensors to improve the battery life of mobile devices, we can achieve exactly the same performance as that summarized in Table II. In this subsection, we evaluate the positioning performance when both Wi-Fi ranging and sensors are used. Of course, in this case, unlabeled Wi-Fi and sensor data will be generated and these data can be used to retrain the ranging module using the sensor-aided learning technique.

One issue is that the heading angle of the device is reported relative to an arbitrary reference direction ϕ_{ref}. According to the experimental results in [40], the EKF can track the correct reference direction using Wi-Fi ranging results if it is initialized well. Therefore, to further improve the reliability of the initial estimation of the reference direction, we modified the EKF design such that multiple candidates of reference directions are considered in the beginning and the best candidate is selected depending on the ranging results. The detailed process is summarized in Appendix B.

Fig. 13 illustrates the estimated trajectory using Wi-Fi ranging and the PDR method. Similar to Fig. 12, the estimated trajectories for the selected scenarios are presented for a clear figure. Because the PDR method provides an accurate trajectory of the device, it is possible to obtain a smoother trajectory compared with the positioning results using Wi-Fi only. Fig. 14 depicts the CDF of the positioning error for every

TABLE III
POSITIONING PERFORMANCE WITH WI-FI RANGING AND SENSORS

Ranging method	MAE [m]	RMSE [m]	90%-tile [m]
Path loss	1.356	1.552	2.384
Polynomial	1.351	1.529	2.311
CUPID	1.373	1.555	2.340
FC (unsupervised)	1.403	1.543	2.252
FC (sensor-aided)	1.192	1.362	2.034
CNN (sensor-aided)	1.038	1.180	1.787

scenario. Similar to Fig. 11(c), which shows the positioning performance using Wi-Fi ranging only, the CNN-based ranging scenario achieves the best positioning performance.

Finally, Table III summarizes the positioning results using Wi-Fi ranging and sensors. With the PDR method, the positioning performance of every scenario was improved from the positioning results using the Wi-Fi module only. In particular, the CNN-based ranging scenario yields the best performance for all metrics because it produces precise distance and standard deviation outputs by identifying the channel condition from the CSI of beacon frames. Using the CSI of beacon frames improves the average positioning accuracy and 90th percentile accuracy by 15–36 cm and 25–60 cm, respectively, from the RSS-based ranging scenarios.

VII. CONCLUSION

In this paper, we studied an unsupervised learning technique to optimize a Wi-Fi ranging module using the sensor data generated in a mobile device. Because the PDR method provides an accurate device trajectory, which shape is almost same as that of the ground truth path, the output of the PDR module could be used as a reference in the training phase. With the proposed cost function that measures the similarity between the Wi-Fi and PDR trajectories, the ranging module autonomously learned how to identify the current channel condition and produce accurate ranging outputs accordingly. In this way, the proposed learning technique could significantly minimize human intervention for performing a site survey process. This paper also verified the benefit of using the CSI of beacon frames for Wi-Fi ranging. As the CSI identifies the channel condition and produces an accurate standard deviation output of each distance estimate, the positioning performance could be improved over using RSS alone. We believe that the CSI can improve the ranging and positioning performances in more complicated indoor environments.

APPENDIX A
PROOF OF LEMMA 1

An offset that minimizes the cost function in equation (22) should satisfy the following relationship:

$$\frac{\partial \mathcal{L}(\mathcal{Z}, \mathcal{P}; \varphi, \boldsymbol{\Omega})}{\partial \boldsymbol{\Omega}} = \sum_k 2(\tilde{\mathbf{p}}^{(k)} - \hat{\mathbf{z}}^{(k)}) = [0, 0]^T. \quad (31)$$

From the above relationship, an optimal offset is derived as

$$\boldsymbol{\Omega}^*(\varphi) = \frac{1}{K} \sum_k \left(\hat{\mathbf{z}}^{(k)} - \mathbf{R}(\varphi)\mathbf{p}^{(k)} \right). \quad (32)$$

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2021.3138850

12

Note that the optimal offset depends on the rotation angle; thus, it is expressed as a function of φ. If we substitute the optimal offset into the original cost function (22), we obtain the following relationship:

$$\mathcal{L}(\mathcal{Z},\mathcal{P};\varphi,\mathbf{\Omega}^*(\varphi)) = \sum_k \|\hat{\mathbf{z}}^{(k)}\|^2 + \sum_k \|\mathbf{p}^{(k)}\|^2$$
$$-\frac{1}{K}\|\sum_k \hat{\mathbf{z}}^{(k)}\|^2 - \frac{1}{K}\|\sum_k \mathbf{p}^{(k)}\|^2 + 2\mathcal{L}(\varphi), \quad (33)$$

where $\mathcal{L}(\varphi)$ contains all terms related to the rotation angle φ, which is given by

$$\mathcal{L}(\varphi) = \frac{\left(\sum_k \hat{\mathbf{z}}^{(k)}\right)^T \mathbf{R}(\varphi)\left(\sum_k \mathbf{p}^{(k)}\right)}{K} - \sum_k (\hat{\mathbf{z}}^{(k)})^T \mathbf{R}(\varphi)\mathbf{p}^{(k)}. \quad (34)$$

Using equation (21) and the symbols defined in (24), the above equation can be rewritten as

$$\mathcal{L}(\varphi) = \Gamma\cos\varphi + \tilde{\Gamma}\sin\varphi, \quad (35)$$

and it satisfies the following inequality:

$$\mathcal{L}(\varphi) = \sqrt{\Gamma^2 + \tilde{\Gamma}^2}\cos(\varphi - \psi) \geq -\sqrt{\Gamma^2 + \tilde{\Gamma}^2}, \quad (36)$$

where $\psi = \arctan\frac{\tilde{\Gamma}}{\Gamma}$. An equality condition for this inequality is given as $\varphi - \psi = \pi$. Therefore, an optimal angle that minimizes the original cost function is expressed as

$$\varphi^* = \pi + \arctan\frac{\tilde{\Gamma}}{\Gamma}, \quad (37)$$

and the optimal offset is accordingly determined as $\mathbf{\Omega}^* = \mathbf{\Omega}(\varphi^*)$. The minimum value of the cost function can be obtained from equation (33) by replacing $\mathcal{L}(\varphi)$ with its minimum value $-\sqrt{\Gamma^2 + \tilde{\Gamma}^2}$.

Appendix B
Positioning with Wi-Fi Ranging and Sensors

To obtain the correct moving direction of the device, the reference direction should be estimated. To this end, we include the unknown reference direction ϕ_{ref} in the state as

$$\boldsymbol{\zeta} = [\mathbf{z}^T, \phi_{ref}]^T = [x, y, \phi_{ref}]^T. \quad (38)$$

Since the reference direction can be any direction in the x-y plane, we simultaneously consider multiple initializations of the state in the beginning, similar to the particle filter [60]. Once the estimated reference direction converges to a certain value, we consider only a single state to reduce the computational complexity. The proposed EKF procedure is summarized as follows.

1) Initialization: We consider C candidates of the EKF state, and the i-th candidate is initialized as

$$\hat{\boldsymbol{\zeta}}_i^{(0)} = [(\hat{\mathbf{z}}^{(0)})^T, \hat{\phi}_{ref,i}^{(0)}]^T, \; 1 \leq i \leq C, \quad (39)$$

where $\hat{\mathbf{z}}^{(0)}$ is the same as in equation (10), and $\hat{\phi}_{ref,i}^{(0)} = \frac{2\pi i}{C}$ is initial reference direction estimate of the i-th candidate. In addition, the covariance matrix of $\hat{\boldsymbol{\zeta}}_i^{(0)}$ is initialized as

$$\tilde{\mathbf{P}}_i^{(0)} = \text{diag}\left(s_x^2, s_y^2, s_\phi^2\right), \quad (40)$$

where s_ϕ is the standard deviation of initial reference direction estimate.

2) State Prediction: The state transition model is given by

$$\boldsymbol{\zeta}^{(k)} = \tilde{\mathbf{f}}(\boldsymbol{\zeta}^{(k-1)}, \Delta\mathbf{p}^{(k)}), \quad (41)$$

where $\Delta\mathbf{p}^{(k)} = \mathbf{p}^{(k)} - \mathbf{p}^{(k-1)}$ represents the movement of the device reported from the PDR module with the assumption that $\phi_{ref} = 0$. The relationship between the elements in the state is given by

$$\mathbf{z}^{(k)} = \mathbf{z}^{(k-1)} + \mathbf{R}(-\phi_{ref}^{(k-1)})\Delta\mathbf{p}^{(k)}, \; \phi_{ref}^{(k)} = \phi_{ref}^{(k-1)}. \quad (42)$$

Using the state transition model, the predicted state of the i-th candidate is obtained as

$$\hat{\boldsymbol{\zeta}}_i^{(k|k-1)} = \tilde{\mathbf{f}}(\hat{\boldsymbol{\zeta}}_i^{(k-1)}, \Delta\mathbf{p}^{(k)}), \quad (43)$$

and its covariance matrix is updated accordingly as

$$\tilde{\mathbf{P}}_i^{(k|k-1)} = \tilde{\mathbf{F}}_i^{(k)}\tilde{\mathbf{P}}_i^{(k-1)}(\tilde{\mathbf{F}}_i^{(k)})^T, \quad (44)$$

where $\tilde{\mathbf{F}}_i^{(k)} \in \mathbb{R}^{3\times3}$ represents the Jacobian matrix defined as

$$\tilde{\mathbf{F}}_i^{(k)} \triangleq \frac{\partial\tilde{\mathbf{f}}(\boldsymbol{\zeta}, \Delta\mathbf{p}^{(k)})}{\partial\boldsymbol{\zeta}}\bigg|_{\boldsymbol{\zeta}=\hat{\boldsymbol{\zeta}}_i^{(k-1)}}. \quad (45)$$

3) State Update: The measurement model is expressed by

$$\mathbf{d}^{(k)} = \tilde{\mathbf{h}}^{(k)}(\boldsymbol{\zeta}^{(k)}) + \boldsymbol{\omega}^{(k)} = \mathbf{h}^{(k)}(\mathbf{z}^{(k)}) + \boldsymbol{\omega}^{(k)}, \quad (46)$$

where $\mathbf{h}^{(k)}(\cdot)$ and $\boldsymbol{\omega}^{(k)}$ are defined in equation (14). The innovation of the i-th candidate and its covariance matrix are given by

$$\tilde{\mathbf{e}}_i^{(k)} = \mathbf{d}^{(k)} - \tilde{\mathbf{h}}^{(k)}(\hat{\boldsymbol{\zeta}}_i^{(k|k-1)}),$$
$$\tilde{\mathbf{S}}_i^{(k)} = \tilde{\mathbf{H}}_i^{(k)}\tilde{\mathbf{P}}_i^{(k|k-1)}(\tilde{\mathbf{H}}_i^{(k)})^T + \mathbf{\Lambda}^{(k)}, \quad (47)$$

respectively. Here, $\tilde{\mathbf{H}}_i^{(k)} \in \mathbb{R}^{N\times3}$ represents the Jacobian matrix defined as

$$\tilde{\mathbf{H}}_i^{(k)} \triangleq \frac{\partial\tilde{\mathbf{h}}^{(k)}(\boldsymbol{\zeta})}{\partial\boldsymbol{\zeta}}\bigg|_{\boldsymbol{\zeta}=\hat{\boldsymbol{\zeta}}_i^{(k|k-1)}}. \quad (48)$$

The remaining processes are similar to those described in Section IV-D. The Kalman gain, updated state, and its covariance matrix of the i-th candidate are calculated as

$$\tilde{\mathbf{G}}_i^{(k)} = \tilde{\mathbf{P}}_i^{(k|k-1)}(\tilde{\mathbf{H}}_i^{(k)})^T(\tilde{\mathbf{S}}_i^{(k)})^{-1},$$
$$\hat{\boldsymbol{\zeta}}^{(k)} = \hat{\boldsymbol{\zeta}}^{(k|k-1)} + \tilde{\mathbf{G}}_i^{(k)}\tilde{\mathbf{e}}_i^{(k)},$$
$$\tilde{\mathbf{P}}_i^{(k)} = \left(\mathbf{I}_3 - \tilde{\mathbf{G}}_i^{(k)}\tilde{\mathbf{H}}_i^{(k)}\right)\tilde{\mathbf{P}}_i^{(k)}. \quad (49)$$

4) Best Candidate Selection: Once the EKF updates the state of every candidate using the latest measurement results, the best candidate is selected based on the innovation (or accumulated innovation) of each candidate as

$$i^* = \underset{i}{\text{argmin}}\|\tilde{\mathbf{e}}_i^{(k)}\|^2. \quad (50)$$

Then, the state of the selected candidate $\hat{\boldsymbol{\zeta}}_{i^*}^{(k)}$ is reported as the state estimate of the device at time step k, where the first two elements are the x- and y-coordinates, and the last element is the reference direction. After a sufficient number of time steps (e.g., 10-20 time steps), we only update the single candidate that was selected as the best.

ACKNOWLEDGMENT

REFERENCES

[1] [Online]. Available: https://youtu.be/-6vfLEBS9M8

[2] Y. Gu, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 1, pp. 13–32, First Quarter 2009.

[3] A. Correa, M. Barcelo, A. Morell, and J. L. Vicario, "A review of pedestrian indoor positioning systems for mass market applications," *Sensors*, vol. 17, no. 8, p. 1927, Aug. 2017.

[4] C. Laoudias, A. Moreira, S. Kim, S. Lee, L. Wirola, and C. Fischione, "A survey of enabling technologies for network localization, tracking, and navigation," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 3607–3644, Fourth Quarter 2018.

[5] Y.-C. Wang, X. Jia, and H. K. Lee, "An indoor wireless positioning system based on wireless local area network infrastructure," in *Proc. 6th Intl. Symp. Satell. Navigat. Technol. Including Mobile Positioning and Location Services*, July 2003, pp. 1–13.

[6] J. Yang and Y. Chen, "Indoor localization using improved RSS-based lateration methods," in *Proc. IEEE Global Telecommun. Conf. (GLOBE-COM)*, Nov. 2009, pp. 1–6.

[7] H. Zou, Z. Chen, H. Jiang, L. Xie, and C. Spanos, "Accurate indoor localization and tracking using mobile phone inertial sensors, WiFi and iBeacon," in *Proc. IEEE Intl. Symp. Inertial Sensors Syst. (INERTIAL)*, Mar. 2017, pp. 1–4.

[8] J. Choi, Y.-S. Choi, and S. Talwar, "Unsupervised learning techniques for trilateration: From theory to android APP implementation," *IEEE Access*, vol. 7, pp. 134,525–134,538, 2019.

[9] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. IEEE Intl. Conf. Comput. Commun. (INFOCOM)*, vol. 2, Mar. 2000, pp. 775–784.

[10] P. Prasithsangaree, P. Krishnamurthy, and P. Chrysanthis, "On indoor position location with wireless LANs," in *Proc. 13th IEEE Intl. Symp. Personal, Indoor, Mobile Radio Commun. (PIMRC)*, vol. 2, Sept. 2002, pp. 720–724.

[11] M. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *Proc. 3rd Intl. Conf. Mobile Syst., Appl., Services (MobiSys)*, June 2005, pp. 205–218.

[12] J. Liu, R. Chen, L. Pei, R. Guinness, and H. Kuusniemi, "A hybrid smartphone indoor positioning solution for mobile LBS," *Sensors*, vol. 12, no. 12, pp. 17,208–17,233, Dec. 2012.

[13] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *ACM SIGCOMM Comput. Commun. Review*, vol. 41, no. 1, p. 53, Jan. 2011.

[14] Y. Xie, Z. Li, and M. Li, "Precise power delay profiling with commodity WiFi," in *Proc. 21st Annu. Intl. Conf. Mobile Comput. Netw. (MobiCom)*, 2015, pp. 53–64.

[15] W. Xu and S. A. Zekavat, "A high-performance measure for non-line-of-sight identification in MIMO-OFDM-based sensor networks," *IEEE Syst. J.*, vol. 8, no. 1, pp. 125–130, Mar. 2014.

[16] Z. Zhou, Z. Yang, C. Wu, W. Sun, and Y. Liu, "LiFi: Line-of-sight identification with WiFi," in *Proc. IEEE Intl. Conf. Comput. Commun. (INFOCOM)*, Apr. 2014, pp. 2688–2696.

[17] Z. Zhou, Z. Yang, C. Wu, L. Shangguan, H. Cai, Y. Liu, and L. M. Ni, "WiFi-based indoor line-of-sight identification," *IEEE Trans. Wireless Commun.* vol. 14, no. 11, pp. 6125–6136, Nov. 2015.

[18] C. Wu, Z. Yang, Z. Zhou, K. Qian, Y. Liu, and M. Liu, "PhaseU: Real-time LOS identification with WiFi," in *Proc. IEEE Intl. Conf. Comput. Commun. (INFOCOM)*, Apr. 2015, pp. 2038–2046.

[19] J. Choi, W. Lee, J. Lee, J. Lee, and S. Kim, "Deep learning based NLOS identification with commodity WLAN devices," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 3295–3303, Apr. 2018.

[20] V.-H. Nguyen, M.-T. Nguyen, J. Choi, and Y.-H. Kim, "NLOS identification in WLANs using deep LSTM with CNN features," *Sensors*, vol. 18, no. 11, pp. 4057, Nov. 2018.

[21] Q. Liu, Z. Huang, and J. Wang, "Indoor non-line-of-sight and multipath detection using deep learning approach," *GPS Solutions*, vol. 23, no. 3, p. 75, May 2019.

[22] K. Bregar and M. Mohori, "Improving indoor localization using convolutional neural networks on computationally restricted devices," *IEEE Access*, vol. 6, pp. 17,429–17,441, 2018.

[23] J. Xiao, K. Wu, Y. Yi, and L. M. Ni, "FIFS: Fine-grained indoor fingerprinting system," in *Proc. 21st Intl. Conf. Comput. Commun. Netw. (ICCCN)*, 2012, pp. 1–7.

[24] 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.

[25] H. Chen, Y. Zhang, W. Li, X. Tao, and P. Zhang, "ConFi: Convolutional neural networks based indoor Wi-Fi localization using channel state information," *IEEE Access*, vol. 5, pp. 18,066–18,074, 2017.

[26] Q. Song, S. Guo, X. Liu, and Y. Yang, "CSI amplitude fingerprinting-based NB-IoT indoor localization," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1494–1504, June 2018.

[27] J. Hua, H. Sun, Z. Shen, Z. Qian, and S. Zhong, "Accurate and efficient wireless device fingerprinting using channel state information," in *Proc. IEEE Intl. Conf. Comput. Commun. (INFOCOM)*, 2018, pp. 1700–1708.

[28] J. Xiong and K. Jamieson, "ArrayTrack: A fine-grained indoor location system," in *Proc. 10th USENIX Conf. Networked Syst. Des. Implementation*, 2013, pp. 7184.

[29] S. Sen, J. Lee, K.-H. Kim, and P. Congdon, "Avoiding multipath to revive inbuilding WiFi localization," in *Proc. 11th Annu. Intl. Conf. Mobile Syst., Appl., Services (MobiSys)*, 2013, p. 249262.

[30] S. Kumar, S. Gil, D. Katabi, and D. Rus, "Accurate indoor localization with zero start-up cost," in *Proc. 20th Annu. Intl. Conf. Mobile Comput. Netw. (MobiCom)*, 2014, pp. 483494.

[31] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "SpotFi: Decimeter level localization using WiFi," in *ACM SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 4, pp. 269-282, Aug. 2015.

[32] A. U. Ahmed, R. Arablouei, F. d. Hoog, B. Kusy, R. Jurdak, and N. Bergmann, "Estimating angle-of-arrival and time-of-flight for multipath components using WiFi channel state information," *Sensors*, vol. 18, no. 6, p. 1753, May 2018.

[33] N. Tadayon, M. T. Rahman, S. Han, S. Valaee, and W. Yu, "Decimeter ranging with channel state information," *IEEE Trans. Wireless Commun.*, vol. 18, no. 7, pp. 3453–3468, July 2019.

[34] *IEEE standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks–specific requirements - part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications*, IEEE Standard 802.11-2016, Dec. 2016, pp. 1–3534.

[35] F. Gringoli, M. Schulz, J. Link, and M. Hollick, "Free your CSI: A channel state information extraction platform for modern Wi-Fi chipsets," in *Proc. 13th Intl. Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization (WinTECH)*, 2019, pp. 21–28.

[36] S. M. Hernandez and E. Bulut, "Lightweight and standalone IoT based WiFi sensing for active repositioning and mobility," in *21st Intl. Symp. World of Wireless, Mobile and Multimedia Netw. (WoWMoM)*, 2020, pp. 277–286.

[37] [Online]. Available: https://git.kernel.org/pub/scm/linux/kernel/git/iwlwifi/iwlwifi-fixes.git/commit/?id=5213e8a8a28d2c4c143fec94e57c866a958ed52d

[38] Z. Xiao, H. Wen, A. Markham, and N. Trigoni, "Robust indoor positioning with lifelong learning," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 11, pp. 2287–2301, 2015.

[39] S. Xu, R. Chen, Y. Yu, G. Guo, and L. Huang, "Locating smartphones indoors using built-in sensors and Wi-Fi ranging with an enhanced particle filter," *IEEE Access*, vol. 7, pp. 95,140–95,153, 2019.

[40] J. Choi and Y. S. Choi, "Calibration-free positioning technique using Wi-Fi ranging and built-in sensors of mobile devices," *IEEE Internet Things J.*, vol. 8, no. 1, pp. 541–554, 2021.

[41] J. Jahn, U. Batzer, J. Seitz, L. Patino-Studencka, and J. Gutirrez Boronat, "Comparison and evaluation of acceleration based step length estimators for handheld devices," in *Proc. Intl. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sept. 2010, pp. 1–6.

[42] W. Kang and Y. Han, "SmartPDR: Smartphone-based pedestrian dead reckoning for indoor localization," *IEEE Sensors J.*, vol. 15, no. 5, pp. 2906–2916, May 2015.

[43] H. Ju, S. Y. Park, and C. G. Park, "A smartphone-based pedestrian dead reckoning system with multiple virtual tracking for indoor navigation," *IEEE Sensors J.*, vol. 18, no. 16, pp. 6756–6764, Aug. 2018.

[44] L. Xu, Z. Xiong, J. Liu, Z. Wang, and Y. Ding, "A novel pedestrian dead reckoning algorithm for multi-mode recognition based on smartphones," *Remote Sensing*, vol. 11, no. 3, p. 294, Feb. 2019.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2021.3138850

14

[45] Z. Xiao, H. Wen, A. Markham, N. Trigoni, P. Blunsom, and J. Frolik, "Non-line-of-sight identification and mitigation using received signal strength," *IEEE Trans. Wireless Commun.*, vol. 14, no. 3, pp. 1689–1702, Mar. 2015.

[46] H. J. Jo and S. Kim, "Indoor smartphone localization based on LOS and NLOS identification," *Sensors*, vol. 18, no. 11, p. 3987, Nov. 2018.

[47] C. Wu, H. Hou, W. Wang, Q. Huang, and X. Gao, "TDOA based indoor positioning with NLOS identification by machine learning," in *Proc. 10th Intl. Conf. Wireless Commun. Signal Process. (WCSP)*, 2018, pp. 1–6.

[48] K. Wu, Jiang Xiao, Youwen Yi, Min Gao, and L. M. Ni, "FILA: Fine-grained indoor localization," in *Proc. IEEE Intl. Conf. Comput. Commun. (INFOCOM)*, 2012, pp. 2210–2218.

[49] J.-W. Yang and G.-H. Cho, "Utilizing CSI to improve distance estimation precision in the indoor environment," *Intl. J. Softw. Eng. Appl.*, vol. 9, no. 3, pp. 49–56, 2015.

[50] Y. Wang, J. Yang, G. Sun, and X. Wang, "Ranging and fingerprinting-based indoor Wi-Fi localisation using channel state information," *IET Commun.*, vol. 13, no. 14, pp. 2106–2114, 2019.

[51] Z. Li, T. Braun, and D. C. Dimitrova, "A passive WiFi source localization system based on fine-grained power-based trilateration," in *Proc. IEEE 16th Intl. Symp. World of Wireless, Mobile and Multimedia Netw. (WoWMoM)*, 2015, pp. 1–9.

[52] J. Wang and J. Park, "A novel indoor ranging algorithm based on a received signal strength indicator and channel state information using an extended Kalman filter," *Appl. Sci.*, vol. 10, p. 3687, May 2020.

[53] A. Niitsoo, T. E, and C. Mutschler, "Convolutional neural networks for position estimation in TDoA-based locating systems," in *Proc. Intl. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, 2018, pp. 1–8.

[54] N. Dvorecki, O. Bar-Shalom, L. Banin, and Y. Amizur, "A machine learning approach for Wi-Fi RTT ranging," in *Intl. Technical Meeting of The Institute of Navigat. 2019*, Jan. 2019.

[55] Z. Gao, Y. Gao, S. Wang, D. Li, and Y. Xu, "CRISLoc: Reconstructable CSI fingerprinting for indoor smartphone localization," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3422–3437, 2021.

[56] K. Feng, J. Li, X. Zhang, C. Shen, Y. Bi, T. Zheng, and J. Liu, "A new quaternion-based Kalman filter for real-time attitude estimation using the two-step geometrically-intuitive correction algorithm," *Sensors*, vol. 17, no. 9, p. 2146, Sept. 2017.

[57] P. Bernal-Polo and H. Martínez-Barberá, "Kalman filtering for attitude estimation with quaternions and concepts from manifold theory," *Sensors*, vol. 19, no. 1, p. 149, Jan. 2019.

[58] J. Choi, Y.-S. Choi, and S. Talwar, "Unsupervised learning technique to obtain the coordinates of Wi-Fi access points," in *Proc. Intl. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sept. 2019, pp. 1–6.

[59] [Online]. Available: https://github.com/selyunin/bno055_usb_stick_py

[60] A. Doucet, N. d. Freitas, K. P. Murphy, and S. J. Russell, "Rao-blackwellised particle filtering for dynamic bayesian networks," in *Proc. 16th Conf. Uncertainty in Artificial Intell. (UAI)*, 2000, pp. 176–183.

Jeongsik Choi received the B.S. degree in electrical engineering from Pohang University of Science and Technology (POSTECH), Pohang, South Korea, in 2010, and the M.S. and Ph.D. degrees in electrical engineering and computer science from Seoul National University, Seoul, South Korea, in 2012 and 2016, respectively. From 2016 to 2017, he was a Senior Researcher with the Institute of New Media and Communications, which is a research institute in Seoul National University. From 2017 to 2021, he was a Research Scientist AI/ML with Intel Labs, Santa Clara, CA, USA. Since September 2021, he has been an Assistant Professor with the School of Electronics Engineering, Kyungpook National University, Daegu, South Korea. His research interests include wireless propagation channel measurement/modeling, sensor fusion and positioning techniques, and he is currently focusing on AI/ML for communication and sensing systems.