

# SiFu: A Generic and Robust Multimodal Signal Fusion Platform for Pervasive Localization

Man Yu Wong<sup>1</sup>, Qing Yan Lu<sup>1</sup>, and S.-H. Gary Chan<sup>1</sup>, *Senior Member, IEEE*

**Abstract**—We consider pervasive localization for a roaming user who may sample widely varying signal modes (GPS, WiFi, geomagnetism, Bluetooth low energy, etc.) and values over time and space. Previous works can only apply to specific (two or three) modes and environment, and cannot accommodate arbitrary signal modes and environmental changes due to signal noise, device heterogeneity, phone carriage states, etc. We propose SiFu, a simple, accurate, and generic multimodal signal fusion platform robust against environmental deviation from its design point. As a generic framework, SiFu treats a single-modal localization algorithm as a black box to embrace any existing, emerging or future signals with only incremental training. It employs Bayesian deep learning and data augmentation to mitigate the location bias of the single-modal localization algorithm and run-time deviation from the training data, respectively. Using a unified multimodal likelihood formulation and particle filter, it fuses with inertial sensor measurements for localization. We conduct extensive experiments in different venues (campus, mall, and subway station), and show that SiFu achieves significantly higher localization accuracy as compared to state-of-the-art (cutting the error by more than 20%). It is also robust against environmental variations (reducing error by 30%), even when the signal values deviate greatly from their original design settings.

**Index Terms**—Bayesian neural network and model averaging, data augmentation, fusion framework, generic platform, robust localization.

## I. INTRODUCTION

THE LOCALIZATION technology has wide and important applications in navigation, location-based marketing, geofencing, etc. [1], [2], [3]. In this work, we consider the case of pervasive localization, where location is to be estimated online continuously and seamlessly when the user roams across diverse indoor and outdoor environments. This indicates that we need to locate not only indoor environments, but also outdoor and semi-indoor environments. We illustrate an example in Fig. 1, where a user roams from one building to another. As he/she moves, the signals the user samples change markedly over space and time, due to drastically different available signal modes (GPS, WiFi, Bluetooth, geomagnetic field, etc.), fluctuating signal values, and dynamic collected set of signal modes.

Numerous localization algorithms have been proposed and studied for individual signal mode, such as GPS, WiFi,

Manuscript received 26 January 2022; revised 20 June 2022; accepted 17 September 2022. Date of publication 21 September 2022; date of current version 22 December 2022. This work was supported in part by the Hong Kong General Research Fund under Grant 16200120. (Corresponding author: Qing Yan Lu.)

The authors are with the Department of Computer Science, Hong Kong University of Science and Technology, Hong Kong (e-mail: mywongar@connect.ust.hk; qlyuag@connect.ust.hk; gchan@cse.ust.hk).

Digital Object Identifier 10.1109/IJOT.2022.3208367

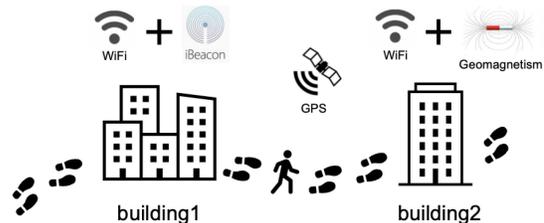


Fig. 1. Case of pervasive localization, where a user is to be localized as he/she roams in dynamic and diverse signal environment.

Bluetooth low energy (BLE), magnetic field, video, image, UWB, Lidar, etc. [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20]. These works in general estimate user location from the spatial likelihood, which is the probability of the user at different points in the area of interest. While impressive, they confirm that every mode has its own limitations, as evident from its strong performance under some conditions and weak in another. For example, GPS is effective in outdoor open environments and not so indoor, while WiFi works well indoors but not so away from city. To support pervasive localization, fusion localization has recently been explored to leverage the signal strength while mitigating the limitations, where multimodal signals sampled by sensors of modern mobile phones are combined to estimate locations.

We consider, for the first time, designing a multimodal fusion platform for pervasive localization achieving the following design goals.

- 1) *Generosity Toward Signal Modes*: As a user roams, the sampled signals are spatially and temporally varying. Spatial variation is due to signal coverage and infrastructure in different areas. For instance, WiFi may be widely available indoors, but GPS would be more accessible outdoors. Temporal variation means that a user, even standing at a location, may sample a dynamic set of signal modes over time, mainly due to the heterogeneous sampling rates of the sensors and nonuniform beaconing intervals of the asynchronous signal emitters. Our fusion platform hence must be a “generic” framework independent of specific signal modes, being able to accommodate flexibly not only existing, but also emerging and future, signal modes.
- 2) *High Robustness and Accuracy to Signal Value Changes*: Needless to say, our fusion algorithm should achieve high localization accuracy by combining different position estimations of the sampled signals. Note that the

environment, such as partitions, objects, human activities, phone carriage states, and device heterogeneity, could affect the measured value of a signal mode at a location, leading to received signal noise or bias, missing values of some (WiFi) access points (APs), distortion of magnetic field signals, etc. The impacts of such signal perturbations should be accounted for to maintain high localization accuracy, i.e., “robustness” against signal value fluctuation. Note that while genericity refers to the availability (presence or absence) of signal *modes*, robustness refers to maintaining the system accuracy against *value* changes of the sampled signal modes.

Although many works have been done on fusion localization, they are highly customized and specialized for only a few (two or three) signal modes and require the modes to be fully available at the time of location estimation [21], [22], [23], [24], [25], [26], [27], [28], [29], [30]. Therefore, they cannot be applied in a highly varying signal environment for pervasive localization. In addition, these algorithms are meticulously designed and engineered, and hence cannot support flexible addition and removal of signal modes, and extension to other emerging or future signals. While some works can handle dynamic signal combination, they are not robust against environmental variations because they are tuned for a particular setting. This makes them difficult to cover more general, wide range, and dynamic environments [31], [32].

Designing a generic multimodal fusion platform robust against environmental changes is challenging, because the platform must not exploit the particular characteristics and bias/error of the signal mode. Moreover, because of environmental dynamics in its running, exhaustively considering all variations in the training data is not practical.

We propose SiFu, a novel, simple, generic, and robust multimodal signal fusion platform for pervasive localization. We illustrate in Fig. 2 a mobile phone running SiFu. With the sensors on, SiFu fuses any set of signal modes sampled at that time to estimate the location. Our work makes the following contributions.

- 1) *A Novel Generic Multimodal Platform for Multimodal Signals* To achieve genericity, SiFu does not devise any localization algorithm specific to signal mode. Instead, it is an original *framework* leveraging upon the existing localization algorithm for each signal mode. Treating the algorithm as a black box, it assumes the general case that each of them outputs independently the spatial likelihood of the user. Based on the sampled modes, SiFu processes and fuses the spatial likelihoods of the modes to estimate user location. The multimodal platform is flexible, accommodating arbitrary combinations of signal modes resulted from dynamic environments and heterogeneous signal sampling rates. It hence enables seamless pervasive roaming over wide signal variation, achieving high elasticity and scalability in signal modes. Thus, it can be easily deployed in any new environments with new signals.
- 2) *High Accuracy With Bayesian Deep Learning*: The spatial likelihood outputted by the algorithm of each mode may be noisy or biased. SiFu employs Bayesian deep

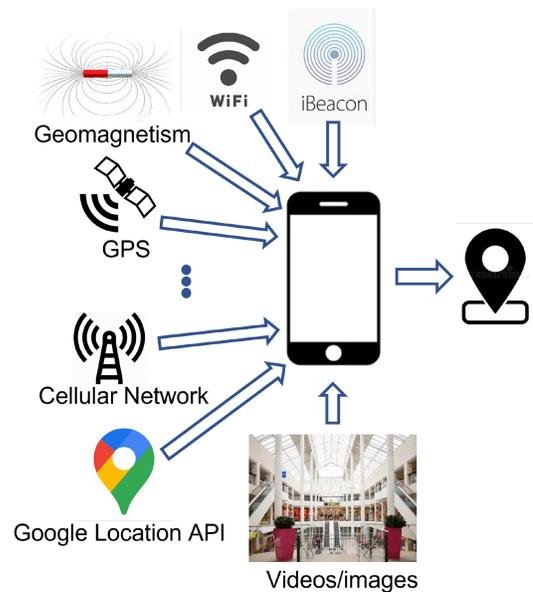


Fig. 2. SiFu, a generic fusion platform for highly accurate and robust localization.

learning models to correct that, leading to a more accurate predicted likelihood for each mode. The predicted likelihoods are then combined by a simple but effective weighted scheme, which seeks to optimize the weight of each mode to minimize the localization error. Finally, a particle filter is employed to integrate the result with user movement as obtained from inertial navigation system (INS) signals to further enhance the accuracy.

- 3) *Robustness Using Data Augmentation Training*: In training our Bayesian deep learning model, we apply data augmentation to simulate signals that may be encountered in reality. Such augmentation approaches unavoidably require the knowledge of the signal characteristics, and we provide examples on how this is done for WiFi and geomagnetic field. The augmentation factors in different environmental changes in the training process, therefore greatly relieving our model from overfitting to a specific environment. As such, this makes SiFu robust to a wide range of run-time environments.

SiFu is easy to implement and deploy. It is training efficient, without extra data collection process beyond those for individual signal modes. Its genericity naturally extends to any emerging or future modes with only incremental training on the newly introduced mode without the need for global training.

We have implemented SiFu in Android phones and conducted extensive experiments in the university campus, a subway station, and a shopping mall, given WiFi, magnetic field, Bluetooth, and GPS as our fusion signals. Our results show that SiFu achieves substantially lower localization error as compared to other state-of-the-art schemes (lower by more than 20% in our experiments). Despite heterogeneous devices, perturbed signals and missing APs, SiFu still maintains its high accuracy with significantly lower localization error as compared with other schemes (by around 30%).

TABLE I  
RELATED WORKS ON FUSION LOCALIZATION

Related Works	[4], [6]–[16]	[21], [22], [27], [29], [33]–[35]	[23]–[25], [28]	[31], [32]	[22], [24], [25], [27], [33]–[34]	—
<b>Approach</b>	Single-modal systems	Fusion on specific signal modes	Fusion on different signal modes	Handling missing signal modes	Improving localization estimation	SiFu
<b>Signals</b>	One single signal (e.g., Wi-Fi, Bluetooth, geomagnetic field, UWB)	Multiple fixed signals	Multiple fixed signals	Multiple fixed signals	—	Multiple signals supporting arbitrary addition, removal, and combination
<b>Performance</b>	Cannot support pervasive localization due to each single signal' limitation	Cannot work properly when missing some signal modes	Tailor-made for a fixed set of signal modes	Use a probabilistic framework, susceptible to environmental variations	Combine users' movement information with signals in highly customized manners	Support any changes in the signal modes, robust against environmental deviation and be generic to support any signal modes

The remainder of this article is organized as follows. We discuss related works in Section II and overview SiFu in Section III. Sections IV and V present the details of the offline training and online prediction processes, respectively. We evaluate SiFu with illustrative experimental results in Section VI, and conclude in Section VII.

## II. RELATED WORKS

We introduce the related works in following two aspects: Section II-A fusion localization and Section II-B mitigating environmental variations.

### A. Fusion Localization

As GPS signals cannot penetrate indoors, researchers have been studying alternative signal modes for indoor localization. Among them, radio-frequency (RF) signals, such as Wi-Fi [4], [6], [7], [8] and Bluetooth [9], [10], and geomagnetic field [11], [12] are popular due to their high availability. Other signal modes, including images/videos, Lidar, and UWB have also been explored [13], [14], [15], [16]. While encouraging results have been reported, each of the signals has its own limitations. Consequently, each of these single-modal systems cannot be used to support pervasive localization alone, but may serve as the building block in our multimodal fusion platform.

Recently, there have been studies on signal fusion. Some works engineer the scheme according to the specific signal modes. For instance, the works in [22], [33], and [30] are designed based on WiFi while [34], [35] builds on UWB. Without the availability of these modes, these works would not be possible. Another typical example is the coarse-to-fine method, which aims to reduce the search space gradually by considering signal modes sequentially. For example, the work [21] constructs heterogeneous filters for WiFi, sound, and motion, which are applied one by one to prune away candidate location points. The works in [27] and Magfi [29] work in a similar manner where the matching space of the magnetic field is constrained by other signal modes. As a sequential structure is adopted in these works, each signal mode tightly couples with the others and the sequence affects the accuracy. Then, when some signal modes are missing, the system might not work properly.

Some other works jointly use different signal modes. In [23], WiFi and magnetic field fingerprints are hybridized in the input level. It stacks WiFi and magnetic field fingerprints into a single vector and estimates user location using kernel discriminant analysis with the KNN algorithm. Similarly, VMag [25] combines magnetic field measurement with the feature vector obtained from the captured image, and then use a neural network to extract the deep features for final location estimation. The proposed scheme in [28] also constructs bimodal images from the magnetic field and light signals as the inputs of its LSTM feature extraction network. The work WiMag [24] first separately processes WiFi, magnetic and inertial signals, and then selects a result according to the phone status. Although impressive, they are not applicable under heterogeneous signal sampling rates because all signal modes have to be available as the input. Also, they are tailor-made for a fixed set of signal modes, and hence cannot be extended to additional or other modes. In contrast, our fusion platform overcomes this problem by unifying heterogeneous signals independently into the same likelihood framework.

To handle the missing signal modes, the work of [31] introduces a maximum-likelihood-based algorithm that uses an F-score weighting scheme where the weights are fixed and based on historical data. Uniloc [32] also designs a probabilistic framework that combines the available localization schemes using the predicted errors estimated in the online phase. These works assume a static environment with the parameters tuned accordingly. By contrast, SiFu is more robust against environmental variations with its data augmentation and self-correcting neural networks for location estimation.

User movement is often utilized to enhance location estimation. Using pedestrian dead reckoning (PDR), stride lengths and orientations can be extracted from inertial sensors, such as the gyroscope, accelerometer, and magnetometer. Works in the area often combine such information with specific signals in a highly customized manner [22], [24], [25], [27], [30], [33], [34]. Nevertheless, SiFu employs a typical particle filter, and is generic to support any signal modes.

In Table I, we summarize the above discussion on SiFu with respect to fusion localization.

TABLE II  
RELATED WORKS ON MITIGATING ENVIRONMENTAL VARIATIONS

Related Works	[6], [36]–[45]	[22], [29], [46]–[49]	[50]	[36], [51]	—
Variation to mitigate	Wi-Fi	Magnetic field	RSSI measurement	Adaptive system parameters	SiFu
Approach	Apply specifically designed methods such as denoising autoencoder, RSSI mapping and transfer learning techniques	Use different filters, mean removal technique and other calibration techniques	RSSI measurements in different phone carriage states	Apply adaptive system parameters	Use arbitrary set of signals with robust performance under environmental variations
Performance	Mitigates signal noise and heterogeneity among devices	Alleviate heterogeneity among devices	Understand the relationship between RSSI and different phone carriage states for proximity detection	Address problem for UWB and PDR systems in particular environments	Not tuned for a specific set of signals or environments but robust against environmental variations

### B. Mitigating Environmental Variations

For WiFi, the work in [6] addresses signal noise and missing signal value problems using a denoising autoencoder. Tilejunction mitigates signal noise by mapping the received signal strength indicator (RSSI) of each AP to a tile for localization [36]. Others have utilized channel state information, that contains more detailed location information, to alleviate the impacts of signal noise [37], [38]. He *et al.* [39] used a subset sampling method to check if there is any altered WiFi fingerprint signal before localization. The works in [40] and [41] introduce transfer learning techniques to adapt RSS measurements by transferring the knowledge from the old model to the altered one, so that localization can still be done after fingerprint adaption. To minimize heterogeneity among devices, the works in [42] and [43] leverage a linear mapping among them. Furthermore, robustness against these environmental variations specifically for fingerprinting approaches is also studied in [44] and [45].

For magnetic field, [46] explores the use of different filters to extract magnetic field fingerprints from noisy measurements. To address the offset in magnetic field measurements caused by heterogeneous devices and carriage states, the mean removal technique is often used [22], [29]. Several other calibration techniques for magnetometers have also been proposed [47], [48], [49].

A few other works have also tackle environmental variation for other signal modes. In [50], RSSI measurements in different phone carriage states are taken to understand their relationship for proximity detection. Moreover, adaptive system parameters are introduced in [35] and [51] to address the nonline-of-sight problem for UWB, and the user heterogeneity and carriage state problems for PDR systems, respectively. Despite the above works, due to their highly customized nature, extending them to cover multimodal signals and a wide range of environmental conditions is not straightforward. As compared to them, SiFu is not designed and tuned for a specific set of signals or a particular environment, but applies to arbitrary set of signals with robust performance under environmental variation for pervasive localization.

TABLE III  
MAJOR SYMBOLS USED IN SiFu

Symbols	Definitions
$\mathbf{x}_i$	Location of the $i$ -th reference point
$p_{ij}$	Likelihood in the $i$ -th reference point for signal mode $j$
$\sigma_w$	Variance of Gaussian white noise for WiFi data augmentation
$\alpha$	Fraction of masked APs for WiFi data augmentation
$\beta$	Fraction of random RSSIs for WiFi data augmentation
$\delta$	Range of device offsets for WiFi data augmentation
$\sigma_m$	Variance of Gaussian white noise for magnetic field data augmentation
$\theta$	Range of additive offsets for magnetic field data augmentation
$\gamma$	Parameter for multiplicative noise for magnetic field data augmentation
$\lambda$	Fraction of masked readings for magnetic field data augmentation
$\mu$	Fraction of random segments for magnetic field data augmentation
$z_w$	Maximum noise level for WiFi data augmentation
$z_m$	Maximum noise level for magnetic field data augmentation

In Table II, we summarize the above discussion on SiFu with respect to mitigating environmental variations.

### III. SiFu OVERVIEW

SiFu is based on spatial likelihoods independently given by the localization algorithm of individual mode. As likelihoods at two physically close locations are expected to be similar, the area of interest is discretized into  $m$  reference points  $\mathbf{x}_1, \dots, \mathbf{x}_m$  where likelihoods will be computed. We list some of the important symbols and acronyms used in this article in Tables III and IV, respectively.

Fig. 3 overviews SiFu, which consists of two phases: 1) offline training phase and 2) online prediction phase. In the offline phase, a likelihood prediction network based on the Bayesian neural network is trained for every signal mode. The general flow of the training process is depicted in Fig. 3(a). Given the original collected data of a signal mode, SiFu augments them to synthesize signals from different localization environments in the data augmentation module. Note that some signal modes such as GPS and cellular may not need this step as they directly provide location results instead of raw signal

TABLE IV  
ACRONYMS USED IN SiFu

Acronyms	Full names
AP	Access Point
BLE	Bluetooth Low Energy
DTW	Dynamic Time Warping
FSW	F-Score-Weighted
GPS	Global Positioning System
INS	Inertial Navigation System
KNN	K-nearest Neighbors Algorithm
LSTM	Long Short-term Memory
NN	Neural Network
PDR	Pedestrian Dead Reckoning
RF	Radio-frequency
RNN	Recurrent Neural Network
RP	Reference Point
RSSI	Received Signal Strength Indicator
UWB	Ultra-wideband

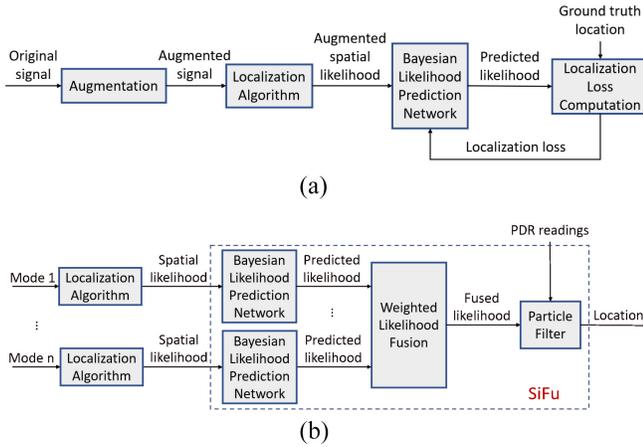


Fig. 3. SiFu overview. (a) Offline training phase. (b) Online prediction phase.

values. Using the black-box localization algorithm on these augmented signals, augmented spatial likelihoods that resemble the one received in a range of localization environments are obtained for training the Bayesian likelihood prediction network. The only requirement for the black-box localization algorithm is giving a likelihood prediction. The network learns to correct these likelihoods from any bias of the localization algorithm and then outputs more accurate predicted likelihoods by using a localization error computed from the ground truth location labels in the collected data for backpropagation.

For the online prediction phase, the major procedures are illustrated in Fig. 3(b). At the localization time, SiFu processes all the signal modes that are sampled within a window of size, say, 2 s. Each available signal mode will be unified into a spatial likelihood using the given localization algorithm. Suppose there are  $n$  observed signal modes  $s_1, \dots, s_n$  in the sampling window, likelihoods  $p_{ij} := p(x_i | s_j)$  for all  $i$  and  $j$  would be computed. Then, the spatial likelihood of signal mode  $j$  would be represented as a vector containing all the likelihoods for the mode as  $[p_{1j}, \dots, p_{mj}]^T$ .

For each signal mode, the Bayesian likelihood prediction network corrects the spatial likelihood into a more accurate predicted likelihood. From the independently computed likelihoods, a weight based on entropy is extracted for each signal mode and is used to fuse the likelihoods together in the

weighted likelihood fusion module. Finally, the final location result is estimated through a particle filter, where the fused likelihood is integrated with the user movement information provided by INS signals with feasible areas of the map as a constraint.

#### IV. OFFLINE TRAINING PHASE

In this section, we discuss the offline training phase of SiFu. In Section IV-A, we overview the data augmentation process to train our Bayesian likelihood prediction network. Next, we provide concrete examples of data augmentation methods for WiFi and magnetic field in Sections IV-B and IV-C, respectively. We present in Section IV-D the details of the network design and the localization loss computation.

##### A. Data Augmentation

SiFu employs data augmentation techniques to train the Bayesian likelihood prediction network. The intuition of data augmentation is to simulate signals sampled under different localization environments. This enables the network to take into consideration the environmental variations during likelihood prediction, and thus, generalize under general operational environments. As the survey process is often performed in a better condition [52], for example, in the least busy time or with a steady carriage state and walking pattern, collected data is usually cleaner and accurate. As such, SiFu makes use of the collected data to create augmented data for training our neural network. It manipulates raw signals instead of directly adding perturbations to the spatial likelihood to make sure that the augmentation follows real-world situations according to the signal characteristics, which could possibly improve the effectiveness of the likelihood prediction network.

While numerous data augmentation techniques for images have already been discussed in previous literature [53], in the following, we provide examples of data augmentation methods for WiFi and magnetic field. Note that the actual data augmentation is not limited to the following methods.

##### B. Case of WiFi RSSI

WiFi data is often represented by an RSSI vector which contains the received signal strengths of the deployed APs. A combination of the following augmentation methods may be used to transform the collected RSSI vectors. In particular, as we would like to simulate signals under environments with different levels of variation, each time the data is augmented, parameters for augmentation (e.g., AP masking and arbitrary RSSI in the following) would be sampled from  $\mathcal{U}(0, z_w)$  where  $z_w$  can be adjusted according to the maximum variation level we are expecting in reality.

- 1) *Additive White Gaussian Noise*: As there is noise in the wireless channels and sensor chips as well as other interference from human activities and partitioning, there would be statistical fluctuations in the received signal strengths at the same location. To model this, the white Gaussian noise with variance  $\sigma_w$  is added to the received signal strengths of the detected APs in the RSSI vector.

- 2) *Offset Noise*: It is also well-known that the received signal strengths measured by different devices at the same location would differ by an offset [42]. Hence, we sample a random offset from  $\mathcal{U}(-\delta, \delta)$  and add it to every received signal strength of detected APs.
- 3) *AP Masking*: Due to the multipath fading effect or blockage of AP signals, APs that are detected can vary at the same location. Therefore, we mask the random APs in the RSSI vector such that only a subset of the APs is used to compute the spatial likelihood. The fraction of masked APs is determined by a parameter  $\alpha$ .
- 4) *Arbitrary RSSI*: Power adjustments of APs or changes in environmental settings would affect the signal distribution. While the number of affected APs usually would not be too large (less than half of the number of APs), the difference in the received signal strengths before and after the introduction of altered APs could be significant. Since how the signal distribution would change cannot be known in advance, arbitrary RSSI values (from  $-100$  to  $-30$  dBm) are set for a portion of APs determined by a parameter  $\beta$ .

### C. Case of Magnetic Field

Instead of a single reading, localization using a magnetic field utilizes magnetic field sequences that contain magnetic field readings within a period of time. Thus, our augmentation focuses on magnetic field sequences. Similar to WiFi signals, multiple augmentation methods would be applied and parameters for random masking and random segment augmentation are generated from  $\mathcal{U}(0, z_m)$ .

- 1) *Gaussian Noise*: Like sensors for RF signals, magnetometers also have intrinsic noise, which is modeled as an additive Gaussian white noise with a variance  $\sigma_m$ .
- 2) *Additive Noise*: In addition to sensor noise, there is an additive component affecting magnetometer readings, which is contributed by multiple factors: a) the hard-iron effect where ferromagnetic materials on the phone add a constant field to the readings; b) the factory calibration offset of the magnetometer; and c) the altitude of the phone [46]. Hence, a random offset sampled from  $\mathcal{U}(-\theta, \theta)$  would be added to every magnetic field reading.
- 3) *Multiplicative Noise*: Noise can be multiplicative as well due to the soft-iron effect where the magnetic field is distorted by external objects [46]. Thus, the magnetic field readings are multiplied by a random value sampled from  $\mathcal{U}(1 - \gamma, 1 + \gamma)$ .
- 4) *Random Masking*: With varying sampling rates of magnetometers in different devices, the number of samples to represent the magnetic field in the same path would be different. To emulate this effect, we could randomly mask a portion of magnetic field readings in the sequence, which is determined by a parameter  $\lambda$ .
- 5) *Random Segment*: Magnetometer readings could also be sensitive to the carriage state, for example, rotation and swinging that may greatly distort the magnetic field readings. Thus, we randomly choose a segment from

the sequence and replace it with a random sequence. The fraction of the random segment is determined by a parameter  $\mu$ .

### D. Bayesian Likelihood Prediction Network and Loss Function

SiFu adopts a Bayesian neural network for the likelihood prediction network. The motivation of using a Bayesian neural network is that we want to obtain an uncertainty estimation for the neural network outputs, which in turn can be used for better fusion.

Here, a Bayesian neural network based on the dropout technique is leveraged [54]. In our framework, it is a deep classification network that comprises of fully connected layers with a dropout layer inserted before every fully connected layer, and finally a Softmax output layer. This Bayesian neural network takes a fixed length vector that is determined by the detected APs in the survey process as inputs. The task of the network is to classify the user location given a spatial likelihood vector  $\mathbf{p}$  generated from the black-box localization algorithm for the sampled signal. As the Softmax layer gives a probability distribution, the output of the network is also a spatial likelihood. Its complexity is the same as traditional BNN that have computational complexity of  $\approx O(K^2)$  by using  $K$ -bit per weights [55] or  $K$  bases in total [56].

In the network, the dropout layers do not only serve as a regularization to avoid overfitting as usual, they also work as a Bayesian approximation of the Gaussian process [54]. With dropout layers, each input unit and weight unit becomes a Bernoulli variable, and it has been shown that this approximates the predictive distribution  $q(\tilde{\mathbf{p}}|\mathbf{p})$ , where  $\mathbf{p}$  and  $\tilde{\mathbf{p}}$  are the input and output spatial likelihoods, respectively.

In the offline training phase, the network is treated essentially the same as the traditional neural network. For the loss function, the cross entropy loss is used. To further reduce the impact of labeling errors in the data, the ground truth labels are smoothed using a Gaussian function into soft labels such that RPs closer to the location label will be assigned higher scores.

## V. ONLINE PREDICTION PHASE

In this section, we present the online localization process. We start by describing the inference process of the Bayesian likelihood prediction network in Section V-A, followed by the weighted likelihood fusion module in Section V-B. Finally, we discuss in Section V-C the integration with particle filter.

### A. Bayesian Likelihood Prediction Network

Softmax in the traditional NN does not provide uncertainty estimation, probability can be high for inputs that it does not know. Thus, a Bayesian NN is needed in the likelihood prediction. In the online phase, the dropout layers in the Bayesian likelihood prediction network are retained so that the input units and weights will be sampled from their respective Bernoulli distribution for inference. To obtain the predicted likelihood,  $T$  forward passes are performed. In other words,  $T$  Monte Carlo estimates are acquired from the approximate

predictive distribution using different sets of weights and input units. By averaging the results, the predicted likelihood for a signal mode is computed as follows:

$$\hat{p} = \mathbb{E}_{q(\tilde{p}|p)}(\tilde{p}) \approx \frac{1}{T} \sum_{t=1}^T \tilde{p}_t. \quad (1)$$

### B. Weighted Likelihood Fusion

There are many fusion technologies and some of them have been proved to be powerful in its field. For example, an equal-averaging fusion of the likelihoods in [57] shows a great performance. However, in our cases multiple competing models remain viable *a posteriori* and our goal is to combine them for reliable estimation. Inspired by [58], a Bayesian Model Averaging has a particularly better performance in this condition than other naive averaging fusion methods. Thus, SiFu adopts the idea of the Bayesian Model Averaging [59] to fuse the independently predicted spatial likelihoods of different signal modes together. Let  $M_1, \dots, M_n$  be the  $n$  models for the available signal modes, where each of them can give a posterior distribution  $p(\mathbf{x}|M_i, s)$  given data  $s$ . The Bayesian Model Averaging suggests that predictions of individual models can be aggregated to compute the final posterior distribution as follows:

$$p(\mathbf{x}|s) = \sum_{j=1}^n p(\mathbf{x}|M_j, s) \times p(M_j|s) \quad (2)$$

where  $p(M_i|s)$  is the posterior model probability.

The posterior model probability aims to account for the uncertainty about individual models. However, it is difficult to evaluate this probability in practice. Hence, we approximate it based on the uncertainty information provided by the Bayesian likelihood prediction network. Intuitively, predictions with less uncertainty should be given higher importance in the final decision.

For classification problems, entropy is a common measure to quantify this uncertainty information. The larger the entropy is, the more uncertain the result is. Given a spatial likelihood  $\hat{p}_j$  for a signal mode  $j$ , we first find the normalized entropy given by

$$H_j = -\frac{1}{\log m} \sum_{i=1}^m \hat{p}_{ij} \log \hat{p}_{ij} \quad (3)$$

where  $m$  is the number of reference points and  $\hat{p}_{ij}$  is the  $i$ th entry of  $\hat{p}_j$ . Note that the range of  $H_j$  is  $[0, 1]$ .

Once we have all the normalized entropies  $H_1, \dots, H_n$ , the weights for signal mode  $j$  are computed as follows:

$$w_j = \frac{1 - H_j}{\sum_{i=1}^n (1 - H_i)} \quad (4)$$

such that  $\sum_{i=1}^n w_i = 1$ .

As  $w_i$  approximates the posterior model probability  $p(M_i|s)$  in 2, the fused likelihood is given by

$$p(\mathbf{x}|s) = \sum_{j=1}^n p(\mathbf{x}|M_j, s) \times w_j. \quad (5)$$

### C. Particle Filter

After the fused likelihood is obtained, a particle filter is used to combine the user movement information obtained from INS signals. The orientation and stride length of the user are estimated from the readings of the gyroscope, magnetometer, and accelerometer based on the works in [60] and [61].

Here, a typical particle filter is considered. It consists of the following four steps: 1) particle prediction; 2) weight update; 3) location estimation; and 4) resampling.

The particle prediction step utilizes the estimated orientation and stride length to propagate the particles. In the weight update stage, the measurement likelihood is required. By Bayes' theorem, it can be computed using the fused likelihood as follows:

$$p(s|\mathbf{x}) = \frac{p(\mathbf{x}|s)p(s)}{p(\mathbf{x})} \propto p(\mathbf{x}|s). \quad (6)$$

While likelihoods are only computed at the RPs and particles do not necessarily locate exactly at them, a particle will take the likelihood value of its closest RP. Furthermore, if a particle violates the map constraints such as moving across a wall, its weight is set to 0. All particle weights are then normalized.

Given a set of  $N$  particles, each with a position  $\mathbf{l}_i$  and a weight  $w_i$ , the user location is estimated by the weighted average of their position as follows:

$$\hat{\mathbf{l}} = \frac{1}{N} \sum_{i=1}^N w_i \mathbf{l}_i. \quad (7)$$

Finally, the resampling step corrects the set of samples based on the evidence where particles will be more likely to be resampled in a region that has a higher probability density.

## VI. ILLUSTRATIVE EXPERIMENTAL RESULTS

We have conducted extensive experiments in various sites to validate the performance of SiFu. In each site, we walk casually like normal users during the positioning, covering most regions of the sites. Thus, we can collect data that are more similar to those in the real cases. In this section, we first describe our experiment settings, comparison schemes, and evaluation metrics in Section VI-A, and then present the influence of system parameters in Section VI-B. We also discuss the fusion algorithm and signal combination in Section VI-C and methods to improve the system performance in Section VI-D. Finally, we illustrate the influence of environmental variations on the fusion localization in Section VI-E.

### A. Experimental Settings and Comparison Schemes

We have evaluated our work in three representative testbeds as shown in Fig. 4 and Table V provides some details about them.

Our testbeds include: 1) a 1800-m<sup>2</sup> university campus area shown in Fig. 4(a) covering indoor, semi-indoor, and outdoor environments; 2) a shopping mall of size 2000 m<sup>2</sup> shown in Fig. 4(b) which is mostly irregular open space; and 3) a subway station shown in Fig. 4(c), consisting of several long

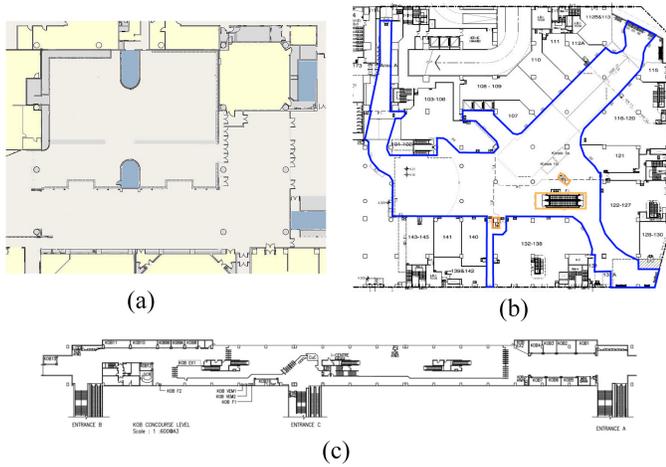


Fig. 4. Maps of the sites in our experiments. (a) University campus. (b) Shopping mall. (c) Subway station.

TABLE V  
DETAILS OF THE EXPERIMENT SITES

Site	Area ( $m^2$ )	Signal modes used in the baseline
University campus	1800	WiFi, magnetic field, GPS
Shopping mall	2000	WiFi, magnetic field
Subway station	2250	WiFi, magnetic field, Bluetooth

corridors and a couple of open space in 2250- $m^2$  area. A self-designed Android App has been installed in users' phone to collect signal values with timestamps. We have conducted two tests for each site and each test lasted for 2–3 min.

To demonstrate the versatility of our platform, we consider several signal modes of different characteristics, including WiFi, magnetic field, Bluetooth, Google API (geolocation result provided by Google), and GPS. Using these modes as examples, we demonstrate how they can be applied in our platform for multimodal localization. We have exhibited the use of different combinations of signal modes in different test sites. In the university campus area, WiFi, magnetic field, and GPS are used in our baseline model while we utilize only WiFi and magnetic field in the shopping mall. For the subway station, WiFi, magnetic field, and Bluetooth are used. As for Wi-Fi APs, they are all are COTS Wi-Fi devices. We notice that the number of APS varies as user walks in different sites. Roughly speaking, ust: 30–60; subway station: 5–20; and shopping mall: 100–150. Many APs are weak in ust and shopping mall, so not all would be useful. After collection, Wi-Fi data is around tens of kB to hundreds of kB, and magnetic field data is around a few to tens of MB. The subway station has a bit less data while ust and shopping mall have more. We use about hundreds of MB for all signals for all devices each site for testing. It should be noted that our platform does not require all signal modes to be available at the localization time. For example, during the experiments in our university campus, GPS is not always received as we roam across indoor, semi-indoor, and outdoor environments. Our results are by no means limited to these signal modes, and may be equally and simply extended to others.

For the black-box localization algorithms, we adopt a cosine similarity-based algorithm for WiFi [5], and a dynamic time warping (DTW)-based algorithm for matching magnetic field sequences [11]. In addition, the geolocation results, including GPS and Google API, are transformed into probability based on the reported location and accuracy. In our experiments, the Bayesian likelihood prediction networks for Google API and GPS have been omitted.

To study the impacts of the Bayesian likelihood prediction network for each signal mode, we first compare with the works that leverage a single signal. For WiFi, we use the following comparison schemes.

- 1) *WKNN* [5] which finds the top  $k$  RPs whose fingerprint RSSI vectors have the highest cosine similarity with the sampled one and computes a weighted average of the positions of the RPs.
- 2) *WiDeep* [6] which applies stacked denoising autoencoders to denoise WiFi RSSI vectors. Noise is injected to the fingerprint data for training. For every reference point, an autoencoder is trained correspondingly. User location is estimated from the likelihoods computed based on how well the autoencoders recover the sampled WiFi RSSI vector.
- 3) *Laafu* [39] which specifically addresses the problem of altered WiFi signals caused by AP power adjustment, environmental changes, etc. It first uses a subset sampling method to identify the APs that have been altered. After filtering out these APs, the user location is estimated by a WKNN algorithm.

For the magnetic field, we use the following for comparison.

- 1) *DTW* [11] which stretches or compresses the time dimension of sequences and finds an optimal warping path between two sequences. Based on the distance of warp paths between fingerprints and the sampled sequence, location is estimated.
- 2) *Recurrent Neural Network (RNN)* [12] which applies RNN to predict locations from magnetic field sequences. To address the ambiguity problem, training data are obtained from a magnetic field map by generating million traces of randomly generated walking paths.

Then, we compare our work with another three state-of-the-art fusion localization algorithms. The parameters settings are tuned to the best performance in our work.

- 1) *Magicol* [22] which fuses WiFi, magnetic field, and INS signals. Likelihoods are computed for WiFi and magnetic field, and a modified particle filter is designed based on the property of WiFi signals to fuse the likelihoods with INS signals.
- 2) *Uniloc* [32] which is a unified scheme to fuse different localization schemes. An error model is trained for each scheme to estimate the errors during the online phase. Based on the estimated errors, location results from different schemes are fused. In this scheme, INS signals are used as one of the spatial signal modes to provide spatial likelihood.
- 3) *F-Score-Weighted (FSW)* [31] which measures the weights of WiFi and magnetic field at different RPs through F-score. The final location is estimated by

TABLE VI  
DEFAULT PARAMETERS

Parameter	Default value
$\sigma_w$	25
$\delta$	10
$\sigma_m$	1
$\gamma$	0.9
$\theta$	5
$z_w$	0.4
$z_m$	0.4
$T$	50
Number of layers	2
Number of neurons in hidden layers	64
Dropout rate	0.3
Grid size	$2m$
Window size	$2s$
Number of particles	1000

minimizing the weighted sum of log-likelihoods of two modes, which are calculated using the Gaussian probability density function.

The localization error, which is the Euclidean distance between the estimated location and the ground truth, serves as the performance metric for comparison. It is the most direct and common measure to evaluate the overall performance of localization systems. Error distributions will also be examined for a more comprehensive assessment. For robustness, we focus on the change of localization errors due to different environmental variations.

Each model was trained for up to a few hours depending on the signal mode and the size of the site. We have implemented the system and tested in Android phones of three different models. The sampling frequency is around one sample every few seconds for WiFi and Bluetooth sensors, and tens of samples every second for magnetic field and other INS sensors. Furthermore, Google Fused Location API returns a location every few seconds while the sampling rate of GPS is generally faster. Our system supports offline localization in real time. It gives a localization result every 0.5 s although the computation time is much lower. The parameter settings in our experiments are shown in Table VI.

### B. Impact of System Parameters

We first study the *effectiveness of our proposed Bayesian likelihood prediction network*. Fig. 5 shows the performance of SiFu and other schemes under different environmental variations for WiFi. In Fig. 5(a), the mean errors of different WiFi localization schemes with different fractions of altered APs are shown. With altered APs, the received RSSIs would be deviated from the fingerprints. However, it can be observed that SiFu is less sensitive to this situation as compared to other schemes. The errors of SiFu increase less rapidly with increasing fraction of altered APs and it gains more than 20% improvements in the accuracy over the other schemes when the fraction becomes more significant. Naive WKNN is not robust to altered APs since it simply uses the deviated RSSIs for localization. Laafu, being a scheme to address altered APs, can work reasonably well when altered APs are not too many since it would identify and filter out these altered APs before localization. If the fraction of altered APs is high, identifying these

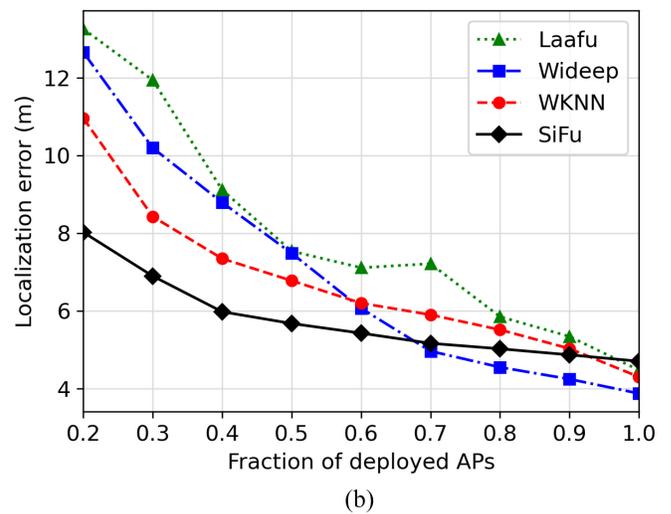
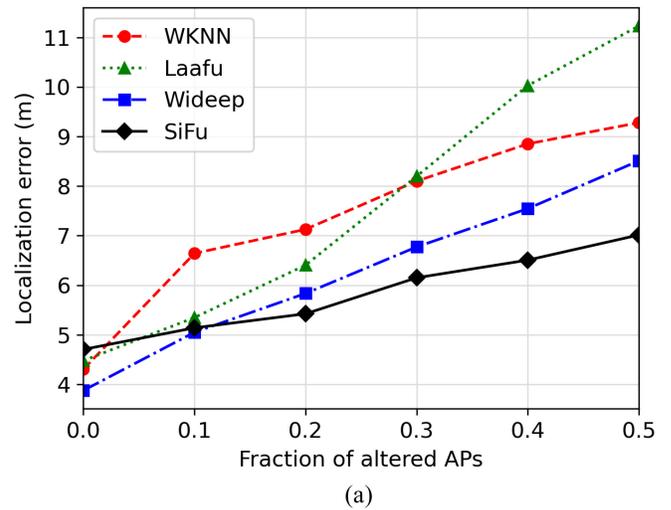


Fig. 5. Mean errors of WiFi localization schemes under different signal environments. (a) Mean errors of WiFi localization schemes with different fractions of altered APs. (b) Mean errors of WiFi localization schemes with different fractions of deployed APs.

APs becomes more difficult using the random subset sampling method, leading to a worse performance. The performance of WiDeep also sharply degrades with altered APs as the representations learnt from the denoising autoencoders may deviate from the actual situation. In comparison, the injection of random RSSI measurements in the data augmentation process of SiFu brings it the capability to work robustly under altered APs. Despite the robustness, we can see a little tradeoff for SiFu in accuracy when there are no altered APs at all. Its performance is slightly worse than the other schemes, probably because our model needs to take into account the uncertainty from the environmental variations, and thus, it sometimes gives more ambiguous and less accurate predictions.

Fig. 5(b) plots the performance of different schemes when only a subset of deployed APs is used for localization. Under this situation, SiFu outperforms the other schemes since its accuracy does not greatly deteriorates as the other schemes do. This is not surprising since the training data have been augmented by masking random APs. The results should be accurate even when some APs are not detected. Similarly, the

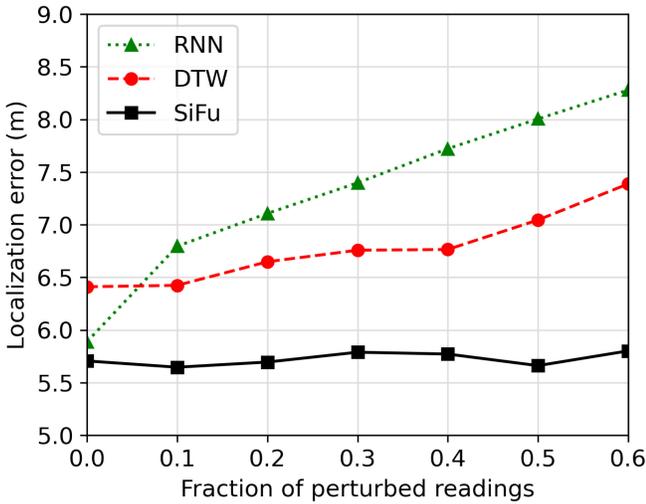


Fig. 6. Mean errors of magnetic field localization schemes with different fractions of perturbed readings.

training of WiDeep also involves AP masking, so its accuracy can be maintained when only the majority of deployed APs are used. However, since it only masks a small and fixed fraction in the training process to maintain its accuracy, its performance drops quickly when using less and less deployed APs. In the meantime, missing APs may not greatly influence the computation of cosine similarity, so the performance of WKNN is still acceptable. For Laafu, reducing the number of used APs may affect the subset sampling process, so it has a worse performance than the naive WKNN. In this figure, we also observe a tradeoff between accuracy and robustness against missing APs. SiFu is relatively less accurate than the others when all APs are used as the model may need to generalize to a range of environments instead of specializing a particular one. These two figures validate the use of our Bayesian likelihood prediction network for WiFi under the situation of altered APs and missing APs.

For the magnetic field, we plot in Fig. 6 the performance of different magnetic field localization schemes under different fractions of perturbed magnetic field readings. Perturbations are done by adding random noise and injecting random segments to emulate distorted measurements caused by unusual carriage states and external ferromagnetic objects. It is obvious that SiFu is both accurate and robust to perturbations in the magnetic field readings compared to other works. On the one hand, it manifests that our network is able to correct the bias in the likelihoods given by the DTW algorithm as it cuts the errors of the naive DTW algorithm by around 20%. On the other hand, using augmentation methods to train our network, our system becomes less vulnerable to unexpected perturbations in reality. On the contrary, the naive DTW algorithm and the RNN model do not train with perturbed data in the offline phase, so they may not work so well in such cases.

We further scrutinize how different *augmentation techniques* in our training process affects the localization results of SiFu. For WiFi, Fig. 7 plots the mean errors when using different combinations of data augmentation methods under different environmental variations. In Fig. 7(a), we show that arbitrary

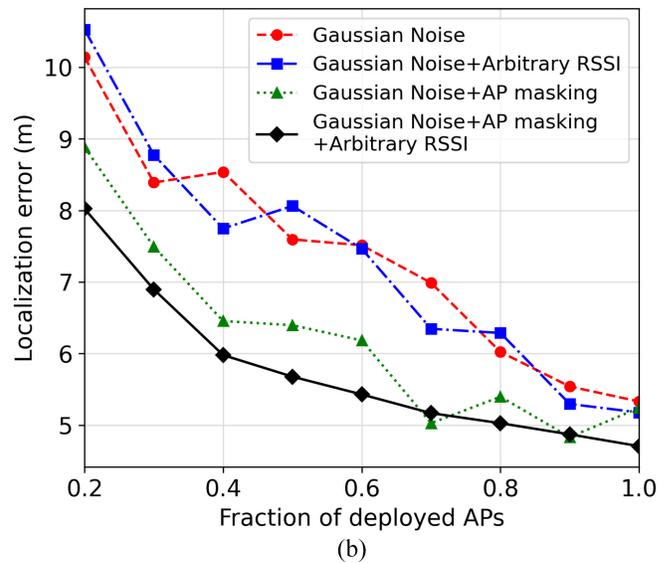
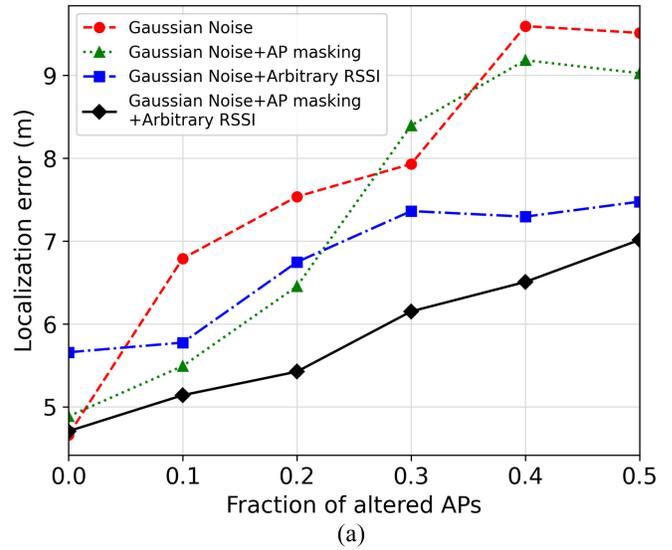


Fig. 7. Mean errors of SiFu under different signal environments using different data augmentation techniques in the training process. (a) Mean errors with different fraction of altered APs. (b) Mean errors with different fraction of deployed APs.

RSSI injection is vital for a good performance under altered APs. We perturb the data that already collected with random noise to emulate power control upon hardware adjustment. Without using it in the data augmentation process, localization errors would be much larger and the accuracy drop is more significant when there are more altered APs. As in Fig. 7(b), it can be seen that AP masking improves the accuracy and robustness under the situation with missing APs. Both models that use AP masking to augment the training data suffer less decline in accuracy than the other two that do not mask any AP in the training process. For the magnetic field, we illustrate in Fig. 8 the mean errors under perturbed magnetic field readings when using different combinations of data augmentation methods in the training process. With no augmentation, the accuracy is worse and obviously drops with perturbed readings. In contrast, noise injection and adding random segments in the training process indeed help reach a higher accuracy

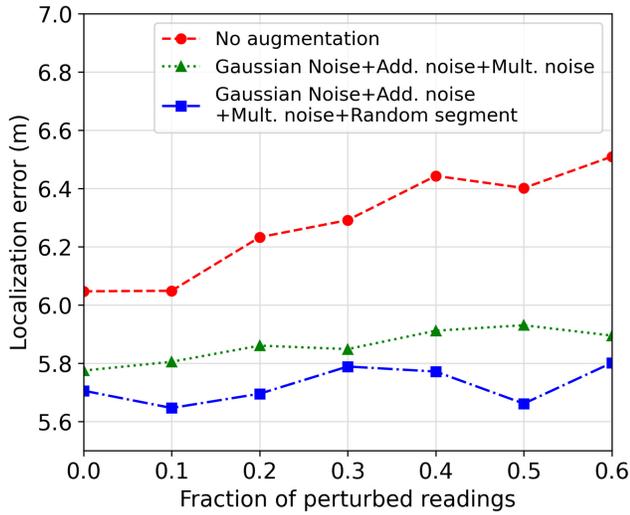


Fig. 8. Mean errors of SiFu with different fractions of perturbed readings using different data augmentation techniques in the training process.

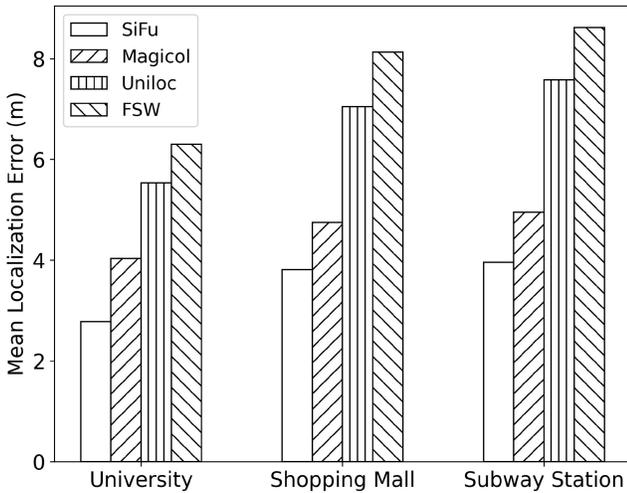
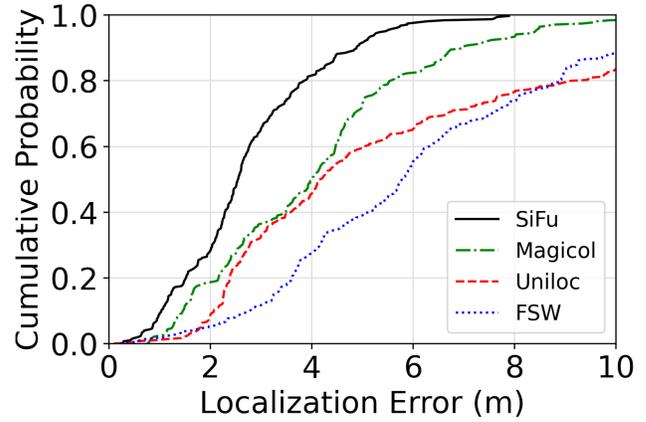


Fig. 9. Mean errors of different schemes in different sites.

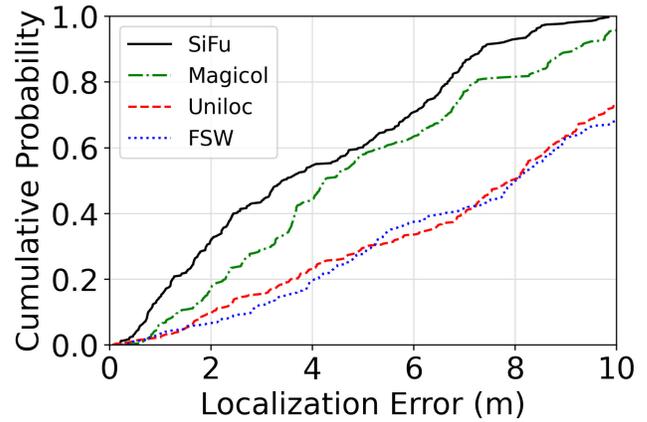
and robustness to perturbed measurements. In other words, data augmentation for the likelihood prediction network is effective to boost the accuracy and robustness under different environmental variations.

### C. Fusion Algorithm and Signal Combination

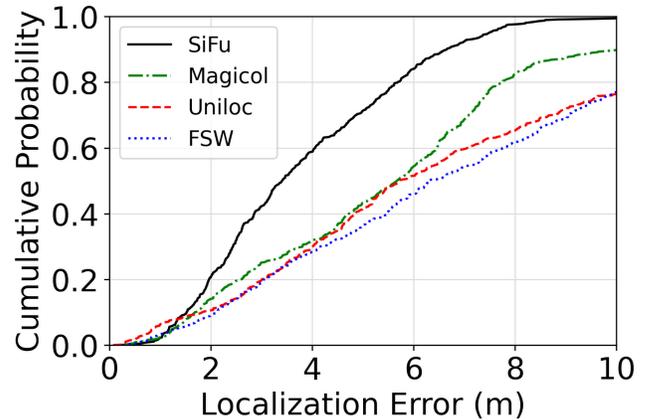
After investigating on single signals, we analyse SiFu in the *fusion perspective*. Fig. 9 presents the mean localization errors of SiFu and the compared schemes over three test sites. We notice notable improvements of SiFu over the other state-of-the-art algorithms in all three sites. The localization errors are cut by at least 20%. FSW and Uniloc perform relatively worse than the other two. This could be owing to the fact that INS signals have not been well utilized in these two schemes. FSW does not fuse them while Uniloc uses them only for spatial location estimation, which will be erroneous due to the accumulated errors in the INS sensors. In contrast, SiFu and Magical adopt particle filters to incorporate temporal information from INS signals, leading to a better performance.



(a)



(b)



(c)

Fig. 10. Cumulative distribution of localization errors in different sites. (a) University campus. (b) Shopping mall. (c) Subway station.

As compared with Magical, our scheme corrects any bias in the given localization algorithms using neural networks and does not specifically depend on WiFi signals. Hence, it would be less negatively affected in areas with weak WiFi signals, especially in the shopping mall and the subway station.

Furthermore, the error distributions of different schemes in various sites are examined as plotted in Fig. 10. It is clear that SiFu achieves the best performance among all compared schemes in all three sites. In general, the maximum error is greatly reduced and the majority of the results are satisfactory.

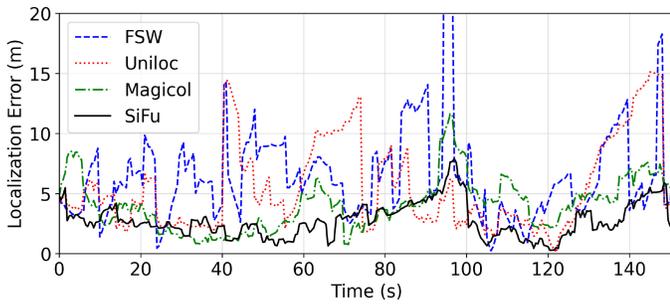


Fig. 11. Real-time localization errors of different schemes.

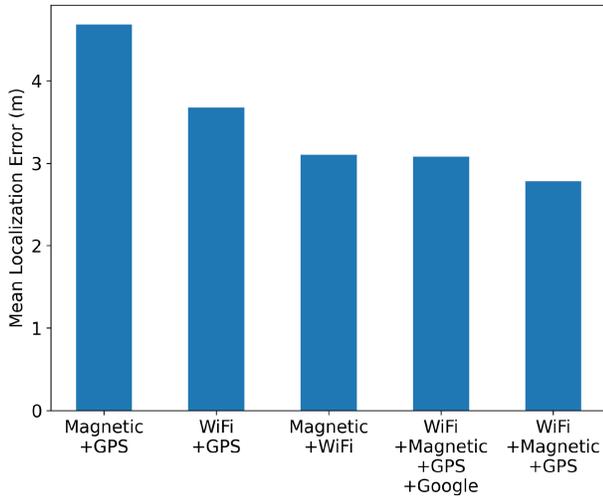


Fig. 12. Mean errors using different combinations of signal modes.

In Fig. 10(a), we observe that the localization errors have remarkable improvements in the university campus. Large errors are observed for FSW and Uniloc and the performance of Magicol is reasonable, but our system further improves the accuracy. Our spatial likelihoods are probably more accurate because of the likelihood prediction networks and the weighted likelihood fusion scheme. In the shopping mall as plotted in Fig. 10(b), the performance of SiFu and Magicol is quite close as they both fuse WiFi, magnetic field and INS signals in a similar way. Still, SiFu is slightly better as our likelihood prediction networks play a role in correcting the spatial likelihood under highly dynamic environments in the shopping mall. Without these networks, Magicol that specifically relies on WiFi may particularly suffer in such an environment with weak or noisy WiFi signals. For Fig. 10(c), all the comparison schemes do not work very well as the weak signals and noisy environments in the subway station makes it challenging for localization. Still, our scheme can attain a good accuracy by incorporating Bluetooth.

Fig. 11 shows the real-time localization errors of different schemes in the university campus. In this path, we performed several indoor–outdoor transitions. As observed in the figure, these transitions can hardly be noticeable for SiFu. Despite some sudden increase in the error, the performance can quickly recover. Although the performance of Magicol is similar to ours, large errors can be observed toward the end of the path and they are less stable when compared to ours. For FSW and

Uniloc, they do not exploit user movement information from INS signals, thus their performance badly fluctuates.

We also assess the mean errors of SiFu in the university campus using different *signal combinations* in Fig. 12. It clearly demonstrates the genericity of our framework as it is possible to fuse different combinations with decent accuracy. In this site, we found that the magnetic field fused with GPS does not work so well because the site covers both indoor and outdoor areas. When GPS signal cannot be received in indoor areas, we can only rely on the magnetic field. It is not ideal due to the low global discernibility of the magnetic field. In contrast, WiFi with GPS is better since WiFi can usually provide rough location estimations with reasonable accuracy, contributing to a more stable performance. When a magnetic field is combined with WiFi, we can indeed alleviate the global ambiguity of magnetic field by the rough location estimates provided by WiFi signals, enabling us to make use of the locally unique magnetic field for better localization. In addition, the performance can be even better if we further include GPS since it can offer a rather accurate clue outdoors. For the Google API, it provides the same location result as GPS when GPS can be received, and becomes not so accurate in indoor areas. Hence, including the Google API may introduce noise in the decision process and slightly reduce the performance. In a nutshell, signal modes that can more consistently yield an acceptable localization result would be more important, for example, WiFi/Bluetooth in indoor areas and GPS in outdoor areas. When these signal modes are available, others can contribute to more fine-grained localization. Nevertheless, additional signal modes do not always help if they are too noisy.

#### D. Performance Improvement

Although many previous works have demonstrated the effect of particle filtering in improving the localization accuracy, we want to prove the effectiveness of using *particle filter* combine the user movement from INS signals in our work for rigour. We have conducted comparison experiments in these three sites. We show the mean localization errors of SiFu with or without using a typical particle filter method in Fig. 13. It illustrates that the accuracy of localization can be improved by about 3–5 m in each site by using particle filtering. Thus, we validate the use of particle filtering in our system.

In Fig. 14, we compare the localization errors of SiFu with each individual signal mode across time. Note that the signal modes are sampled in different rates and are not always fully available at the localization time. This demonstrates that our framework is capable of handling dynamic and arbitrary combinations of signal modes during localization time. In terms of localization performance, although the localization errors of some signal modes are sometimes large, the errors of SiFu do not follow. Instead, SiFu keeps a rather *consistent performance*. In particular, its performance is closer to the signal whose performance is better as our weighting scheme possibly allows us to trust more on the better signal modes. Yet, it is expected that our scheme cannot always trace the best envelope of the signal modes as it only assigns weights

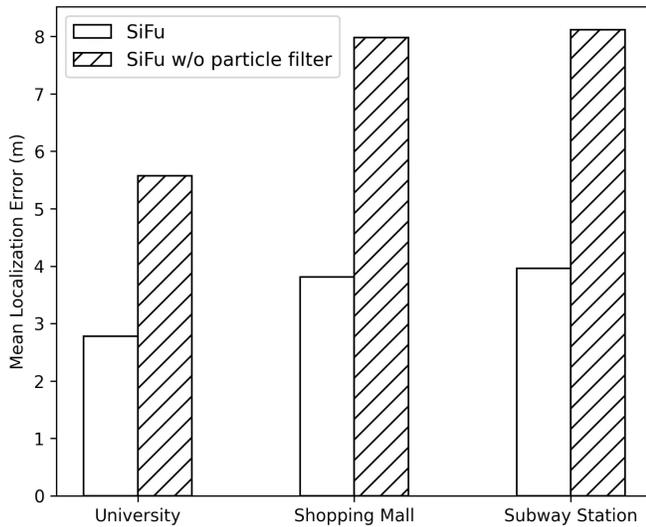


Fig. 13. Mean localization errors of SiFu with/without particle filtering.

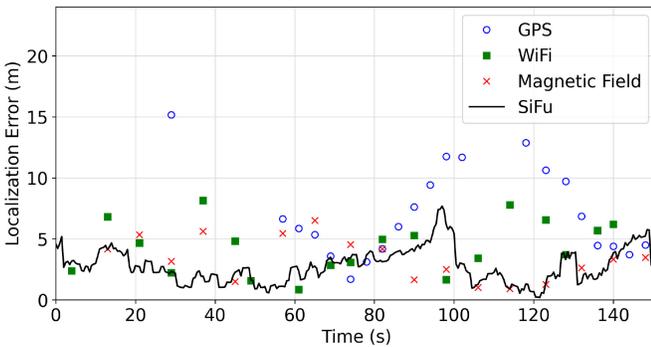


Fig. 14. Real-time errors of SiFu and individual signal modes.

based on the uncertainty of the modes instead of choosing the best one. As a result, information from the less reliable signal modes will still be included into consideration.

### E. Influence of Environmental Variations on the Fusion Localization

Similar to the single signal case, we are interested in how different environmental variations affect the fusion localization schemes.

Fig. 15 demonstrates the performance of SiFu and other fusion works under different *noise levels*. We define the noise level as the level of various environmental variations (fraction of altered APs and fraction of missing APs for WiFi, and the fraction of perturbed readings for magnetic field) such that both WiFi and magnetic field are affected. The result reveals that SiFu can achieve a highly accurate and robust performance despite these environmental variations. Over 30% improvements can be observed. While its accuracy is the highest among different schemes in the baseline situation, the localization error does not drastically rise either as the noise level increases. Contrastingly, the performance of other schemes is more adversely affected. This reinforces that our Bayesian likelihood prediction networks are able to generalize and correct likelihood estimations under environmental variations.

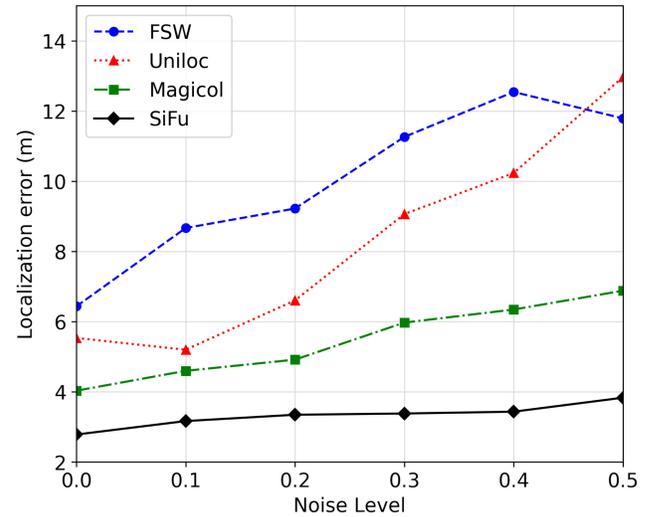


Fig. 15. Mean errors of fusion schemes under different noise levels.

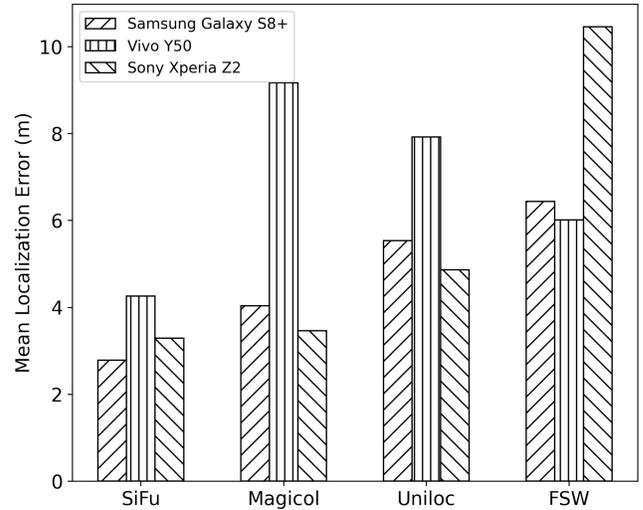


Fig. 16. Mean errors of fusion schemes using different devices.

Since the performance of WiFi and INS sensors can be device dependent, we evaluate the ability of SiFu to work in *heterogeneous devices*. Fig. 16 plots the mean localization errors of different schemes using different models of smartphones. We can see that SiFu's performance is consistent among different devices, while other schemes may perform badly for some phone models. SiFu is unsurprisingly better in handling device heterogeneity because emulating signals from different devices is one of our data augmentation methods for training our likelihood prediction networks, allowing SiFu to work under deviations across devices.

Furthermore, we investigate the impact of different parameters in our system on the localization performance. We start with the parameters related to the Bayesian likelihood prediction network.

In Fig. 17, the mean localization errors of likelihood prediction networks for WiFi and magnetic field with different *numbers of hidden layers* are plotted. It is shown that our networks could be simple that without requiring many layers. While more layers may be helpful for the networks to learn

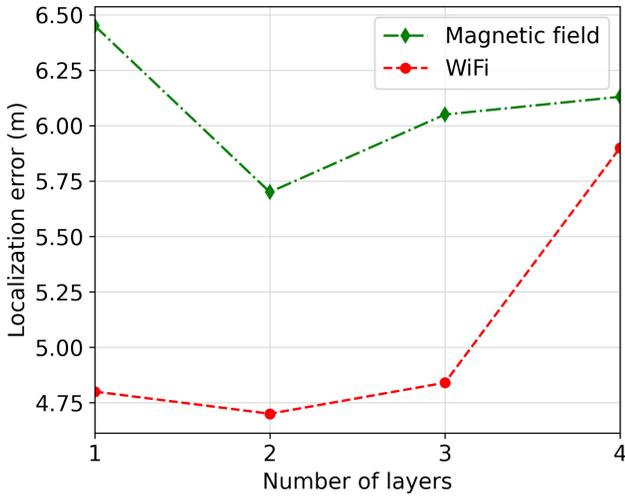


Fig. 17. Mean errors of the likelihood prediction network with a different number of layers.

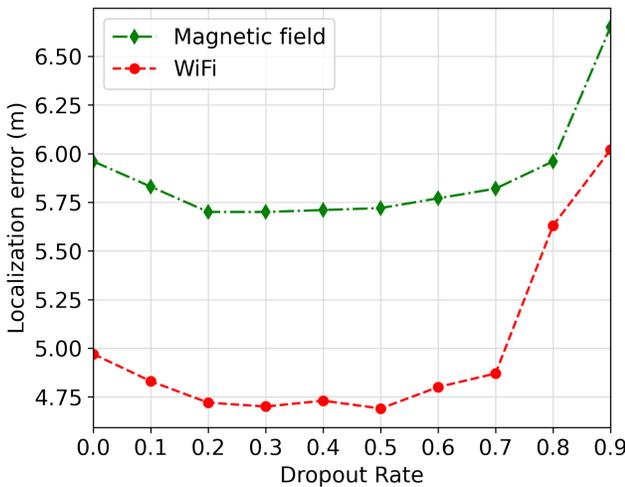


Fig. 18. Mean errors of the likelihood prediction network with different dropout rates.

more information, networks with too many layers can be over-fitted to the training data and cannot generalize at the test time.

We also illustrate in Fig. 18 the mean localization errors of likelihood prediction networks for WiFi and magnetic field using different *dropout rates*. Similar trends can be observed for both signals. With increasing dropout rates, the errors generally first decrease, and then remain constant. As the dropout rate becomes larger, the errors increase again and grow more rapidly with a large dropout rate. When the dropout rate is zero, the networks are simply conventional neural networks which ignore any uncertainty in the weights or the inputs. This can lead to overconfident but incorrect predictions. Nevertheless, a large dropout rate drops weight units in the network too often and the results can be inaccurate if too few weight units are used for prediction. With a suitable dropout rate, the network can reach a better performance.

In addition to the neural network parameters, Fig. 19 plots the mean localization errors in different sites with different

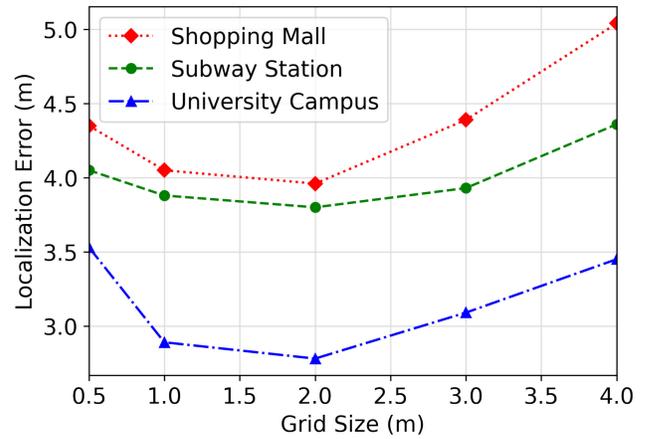


Fig. 19. Mean errors in different sites with different grid sizes.

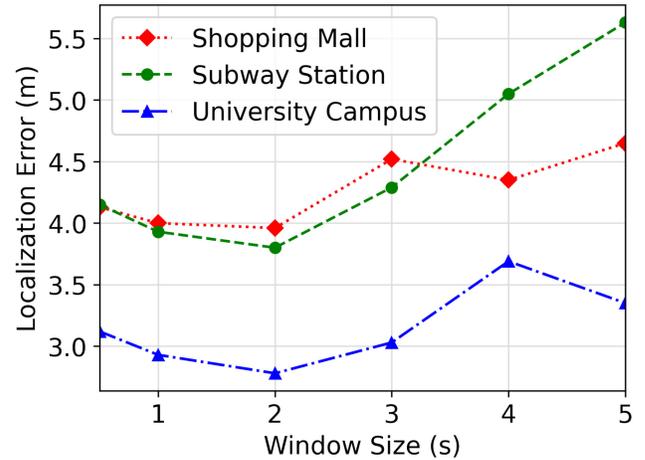


Fig. 20. Mean errors in different sites with different window sizes.

*grid sizes*, which we define as the distance between adjacent reference points. In general, we find that there is an optimal grid size. With an increasing grid size, grid classification accuracy of the likelihood prediction networks increases since a smaller number of classes and denser training data ease the training. This leads to better localization accuracy. On the contrary, localization accuracy decreases with grid size due to grid granularity. Likelihood at a reference point may not accurately reflect the probability of the user location within the neighborhood of the grid. Therefore, an appropriate grid size is essential for the best performance.

In Fig. 20, we analyse the impacts of the *sampling window size* on the localization errors in different sites. In general, the curves follow a V-shape. If the window size is small, there will often be at most one signal mode sampled within the window. In this case, our weighting scheme cannot bring into play as the weighted likelihood will exactly equal the likelihood of the only signal mode. This is not desirable when the sole likelihood is not accurate. On the other hand, setting the window size too big can harm the accuracy since the window size determines how frequently the likelihood will be computed and used to correct the particles in the particle filter. In other words, we should find a window size such that multiple signal modes can be sampled within the window

while being small enough to allow timely update of particles' weights.

## VII. CONCLUSION

As a user roams across different indoor and outdoor environments, the sampled data exhibit markedly varying signal modes and values. To support pervasive localization, such variation has to be accounted for with robust accuracy. Signal fusion has been shown to achieve high accuracy by leveraging the strengths while mitigating the weaknesses of individual signal modes. Prior works in the area are usually highly customized for a few (two or three) signal modes, and targeted to work in a specific environment. To address this for the pervasive environment, we propose SiFu, a novel, simple, highly accurate, and generic multimodal signal fusion platform which supports arbitrary addition, removal, and combination of signal modes, and achieves robustness against environmental variations, such as signal noise, device heterogeneity, missing signal values, phone carriage states, etc. SiFu is a framework, and hence supports any existing, emerging, and future signals based on their individual localization algorithms as black boxes.

To the best of our knowledge, SiFu is the first piece of work achieving both genericities in signal modes and robustness in performance for pervasive localization. It adopts a probabilistic fusion formulation. Signal modes are first independently processed, and then unified as likelihoods into the platform. Furthermore, it utilizes Bayesian neural networks to yield more accurate likelihoods, and combines them in a weighted fashion to minimize the localization error. To generalize its deep learning model to a wide range of environments, SiFu employs data augmentation in the training phase. We have implemented SiFu in Android mobile phones and conducted extensive experiments in a university campus, a shopping mall, and a subway station. Our results show that SiFu achieves significantly higher accuracy and robustness in online localization. It outperforms other state-of-the-art schemes by a large margin (more than 20% in terms of localization accuracy). In spite of environmental variations, its localization error is still significantly lower than state-of-the-art schemes (by around 30%).

We discuss some of the possible directions for future work here. Our current work studies traditional signals, such as Wi-Fi, GPS, etc. One may extend it to emerging signals, such as image and/or video. In this regard, the processing and memory capability of the handheld device needs to be considered. Another emerging signal is 5G, which should be incorporated into our framework to achieve pervasive localization. UWB, CSI, and Lidar are emerging signals worth studying as well.

## REFERENCES

- [1] A. Billa, I. Shayea, A. Alhammadi, Q. Abdullah, and M. Roslee, "An overview of indoor localization technologies: Toward IoT navigation services," in *Proc. IEEE 5th Int. Symp. Telecommun. Technol. (ISTT)*, 2020, pp. 76–81.
- [2] B. Barazandeh, M. Rafieisakhaei, A. Moosavi, and K. Bastani, "Effect of localization on the car market under intense sanctions; a system dynamics approach," in *Proc. 34th Int. Conf. Syst. Dyn. Soc.*, 2016, pp. 1–7.
- [3] J. Joshi, A. Patil, M. Yogesh, and A. Karwa, "GNL: GeoFencing based smart outdoor navigation and localization," *ADBU J. Eng. Technol.*, vol. 11, no. 1, pp. 1–6, 2022.
- [4] P. Bahl and V. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. 19th Annu. Joint Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, 2000, pp. 775–784.
- [5] S. Han, C. Zhao, W. Meng, and C. Li, "Cosine similarity based fingerprinting algorithm in WLAN indoor positioning against device diversity," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2015, pp. 2710–2714.
- [6] M. Abbas, M. Elhamshary, H. Rizk, M. Toriki, and M. Youssef, "WiDeep: WiFi-based accurate and robust indoor Localization system using deep learning," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, 2019, pp. 1–10.
- [7] M. T. Hoang, B. Yuen, X. Dong, T. Lu, R. Westendorp, and K. Reddy, "Recurrent neural networks for accurate RSSI indoor Localization," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10639–10651, Dec. 2019.
- [8] X. Song *et al.*, "A novel convolutional neural network based indoor localization framework with WiFi fingerprinting," *IEEE Access*, vol. 7, pp. 110698–110709, 2019.
- [9] C. Xiao, D. Yang, Z. Chen, and G. Tan, "3-D BLE indoor localization based on denoising autoencoder," *IEEE Access*, vol. 5, pp. 12751–12760, 2017.
- [10] T. Wu, H. Xia, S. Liu, and Y. Qiao, "Probability-based indoor positioning algorithm using iBeacons," *Sensors*, vol. 19, no. 23, p. 5226, 2019. [Online]. Available: <https://www.mdpi.com/1424-8220/19/23/5226>
- [11] K. P. Subbu, B. Gozick, and R. Dantu, "Indoor localization through dynamic time warping," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, 2011, pp. 1639–1644.
- [12] H. J. Jang, J. M. Shin, and L. Choi, "Geomagnetic field based indoor localization using recurrent neural networks," in *Proc. IEEE Global Commun. Conf.*, 2017, pp. 1–6.
- [13] M. Li, N. Liu, Q. Niu, C. Liu, S.-H. G. Chan, and C. Gao, "SweepLoc: Automatic video-based indoor localization by camera sweeping," *Proc. ACM Interact. Mobile Wearable Ubiquitous Technol.*, vol. 2, no. 3, p. 120, Sep. 2018. [Online]. Available: <https://doi.org/10.1145/3264930>
- [14] Q. Niu, M. Li, S. He, C. Gao, S. H. G. Chan, and X. Luo, "Resource-efficient and automated image-based indoor localization," *ACM Trans. Sens. Netw.*, vol. 15, no. 2, p. 19, Feb. 2019. [Online]. Available: <https://doi.org/10.1145/3284555>
- [15] R. W. Wolcott and R. M. Eustice, "Robust LIDAR localization using multiresolution Gaussian mixture maps for autonomous driving," *Int. J. Robot. Res.*, vol. 36, no. 3, pp. 292–319, 2017. [Online]. Available: <https://doi.org/10.1177/0278364917696568>
- [16] K. Gururaj, A. K. Rajendra, Y. Song, C. L. Law, and G. Cai, "Real-time identification of NLOS range measurements for enhanced UWB localization," in *Proc. Int. Conf. Indoor Position. Indoor Navigation (IPIN)*, 2017, pp. 1–7.
- [17] I. Ashraf, M. Kang, S. Hur, and Y. Park, "MINLOC: Magnetic field patterns-based indoor localization using convolutional neural networks," *IEEE Access*, vol. 8, pp. 66213–66227, 2020.
- [18] I. Fahmy, S. Ayman, H. Rizk, and M. Youssef, "MonoFi: Efficient indoor localization based on single radio source and minimal fingerprinting," in *Proc. 29th Int. Conf. Adv. Geographic Inf. Syst.*, 2021, pp. 674–675.
- [19] D. Ciuonzo, P. S. Rossi, and P. K. Varshney, "Distributed detection in wireless sensor networks under multiplicative fading via generalized score tests," *IEEE Internet Things J.*, vol. 8, no. 11, pp. 9059–9071, Jun. 2021.
- [20] R. Niu, A. Vempaty, and P. K. Varshney, "Received-signal-strength-based localization in wireless sensor networks," *Proc. IEEE*, vol. 106, no. 7, pp. 1166–1182, Jul. 2018.
- [21] M. Azizyan, I. Constandache, and R. R. Choudhury, "SurroundSense: Mobile phone localization via ambient fingerprinting," in *Proc. 15th Annu. Int. Conf. Mobile Comput. Netw.*, 2009, pp. 261–272.
- [22] Y. Shu, C. Bo, G. Shen, C. Zhao, L. Li, and F. Zhao, "Magicol: Indoor localization using pervasive magnetic field and opportunistic WiFi sensing," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 7, pp. 1443–1457, Jul. 2015.
- [23] M. Zhang, W. Shen, and J. Zhu, "WiFi and magnetic fingerprint positioning algorithm based on KDA-KNN," in *Proc. Chin. Control Decis. Conf. (CCDC)*, 2016, pp. 5409–5415.
- [24] X. Guo, W. Shao, F. Zhao, Q. Wang, D. Li, and H. Luo, "WiMag: Multimode fusion localization system based on magnetic/WiFi/PDR," in *Proc. Int. Conf. Indoor Position. Indoor Navigation (IPIN)*, 2016, pp. 1–8.

- [25] Z. Liu, L. Zhang, Q. Liu, Y. Yin, L. Cheng, and R. Zimmermann, "Fusion of magnetic and visual sensors for indoor localization: Infrastructure-free and more effective," *IEEE Trans. Multimedia*, vol. 19, no. 4, pp. 874–888, Apr. 2017.
- [26] H. Xu, Z. Yang, Z. Zhou, L. Shangguan, K. Yi, and Y. Liu, "Indoor localization via multi-modal sensing on smartphones," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, 2016, pp. 208–219.
- [27] Y. Li, Y. Zhuang, P. Zhang, H. Lan, X. Niu, and N. El-Sheimy, "An improved inertial/WiFi/magnetic fusion structure for indoor navigation," *Inf. Fusion*, vol. 34, pp. 101–119, Mar. 2017.
- [28] X. Wang, Z. Yu, and S. Mao, "DeepML: Deep LSTM for indoor localization with smartphone magnetic and light sensors," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2018, pp. 1–6.
- [29] H. Wu, Z. Mo, J. Tan, S. He, and S.-H. G. Chan, "Efficient indoor localization based on geomagnetism," *ACM Trans. Sen. Netw.*, vol. 15, no. 4, p. 42, Aug. 2019. [Online]. Available: <https://doi.org/10.1145/3342517>
- [30] X. Liu *et al.*, "Kalman filter-based data fusion of Wi-Fi RTT and PDR for indoor localization," *IEEE Sensors J.*, vol. 21, no. 6, pp. 8479–8490, Mar. 2021.
- [31] S. B. Keser, A. Yazici, and S. Gunal, "An F-score-weighted indoor positioning algorithm integrating WiFi and magnetic field fingerprints," *Mobile Inf. Syst.*, vol. 2018, pp. 1–8, May 2018.
- [32] W. Du, P. Ton, and M. Li, "UniLoc: A unified mobile localization framework exploiting scheme diversity," in *Proc. IEEE 38th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2018, pp. 818–829.
- [33] F. Hong, Y. Zhang, Z. Zhang, M. Wei, Y. Feng, and Z. Guo, "WaP: Indoor localization and tracking using WiFi-assisted particle filter," in *Proc. 39th Annu. IEEE Conf. Local Comput. Netw.*, 2014, pp. 210–217.
- [34] W. You, F. Li, L. Liao, and M. Huang, "Data fusion of UWB and IMU based on unscented Kalman filter for indoor localization of quadrotor UAV," *IEEE Access*, vol. 8, pp. 64971–64981, 2020.
- [35] D. Yuan, J. Zhang, J. Wang, X. Cui, F. Liu, and Y. Zhang, "Robustly adaptive EKF PDR/UWB integrated navigation based on additional heading constraint," *Sensors*, vol. 21, no. 13, p. 4390, 2021. [Online]. Available: <https://www.mdpi.com/1424-8220/21/13/4390>
- [36] S. He and S.-H. G. Chan, "Tilejunction: Mitigating signal noise for fingerprint-based indoor localization," *IEEE Trans. Mobile Comput.*, vol. 15, no. 6, pp. 1554–1568, Jun. 2016.
- [37] X. Wang, L. Gao, S. Mao, and S. Pandey, "DeepFi: Deep learning for indoor fingerprinting using channel state information," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2015, pp. 1666–1671.
- [38] 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. 18066–18074, 2017.
- [39] S. He, W. Lin, and S.-H. G. Chan, "Indoor localization and automatic fingerprint update with altered AP signals," *IEEE Trans. Mobile Comput.*, vol. 16, no. 7, pp. 1897–1910, Jul. 2017.
- [40] Z. Sun, Y. Chen, J. Qi, and J. Liu, "Adaptive localization through transfer learning in indoor Wi-Fi environment," in *Proc. 7th Int. Conf. Mach. Learn. Appl.*, 2008, pp. 331–336.
- [41] M. M. Atia, A. Noureldin, and M. J. Korenberg, "Dynamic online-calibrated radio maps for indoor positioning in wireless local area networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 9, pp. 1774–1787, Sep. 2013.
- [42] S. He, T. Hu, and S.-H. G. Chan, "Toward practical deployment of fingerprint-based indoor localization," *IEEE Pervasive Comput.*, vol. 16, no. 2, pp. 76–83, Apr.–Jun. 2017.
- [43] Y. Li, S. Williams, B. Moran, and A. Kealy, "A probabilistic indoor localization system for heterogeneous devices," *IEEE Sensors J.*, vol. 19, no. 16, pp. 6822–6832, Aug. 2019.
- [44] X. Guo, N. R. Elikplim, N. Ansari, L. Li, and L. Wang, "Robust WiFi localization by fusing derivative fingerprints of RSS and multiple classifiers," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3177–3186, May 2020.
- [45] N. Alikhani, V. Moghtadaiee, and S. A. Ghorashi, "Fingerprinting based indoor localization considering the dynamic nature of Wi-Fi signals," *Wireless Personal Commun.*, vol. 115, no. 2, pp. 1445–1464, Nov. 2020. [Online]. Available: <https://doi.org/10.1007/s11277-020-07636-0>
- [46] W. Shao *et al.*, "Location fingerprint extraction for magnetic field magnitude based indoor positioning," *J. Sensors*, vol. 2016, Dec. 2016, Art. no. 1945695.
- [47] M. Hehn, E. Sippel, C. Carlowitz, and M. Vossiek, "High-accuracy localization and calibration for 5-DoF indoor magnetic positioning systems," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 10, pp. 4135–4145, Oct. 2019.
- [48] A. Poullose, J. Kim, and D. S. Han, "Indoor localization with smartphones: Magnetometer calibration," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, 2019, pp. 1–3.
- [49] B. Siebler, S. Sand, and U. D. Hanebeck, "Localization with magnetic field distortions and simultaneous magnetometer calibration," *IEEE Sensors J.*, vol. 21, no. 3, pp. 3388–3397, Feb. 2021.
- [50] G. F. Hatke *et al.*, "Using Bluetooth low energy (BLE) signal strength estimation to facilitate contact tracing for COVID-19," 2020, *arXiv:2006.15711*.
- [51] X. Teng *et al.*, "ARPDR: An accurate and robust pedestrian dead reckoning system for indoor localization on handheld smartphones," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2020, pp. 10888–10893.
- [52] P. Roy and C. Chowdhury, "A survey on ubiquitous WiFi-based indoor localization system for smartphone users from implementation perspectives," *CCF Trans. Pervasive Comput. Interact.*, vol. 4, pp. 298–318, Jan. 2022.
- [53] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, p. 60, Jul. 2019. [Online]. Available: <https://doi.org/10.1186/s40537-019-0197-0>
- [54] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. PMLR*, 2016, pp. 1050–1059.
- [55] S. Zhou, Y. Wu, Z. Ni, X. Zhou, H. Wen, and Y. Zou, "Dorefa-Net: Training low bitwidth convolutional neural networks with low bitwidth gradients," 2016, *arXiv:1606.06160*.
- [56] X. Lin, C. Zhao, and W. Pan, "Towards accurate binary convolutional neural network," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–9.
- [57] G. Aceto, D. Ciunzo, A. Montieri, and A. Pescapè, "MIMETIC: Mobile encrypted traffic classification using multimodal deep learning," *Comput. Netw.*, vol. 165, Dec. 2019, Art. no. 106944.
- [58] M. Hinne, Q. F. Gronau, D. van den Bergh, and E.-J. Wagenmakers, "A conceptual introduction to Bayesian model averaging," *Adv. Methods Pract. Psychol. Sci.*, vol. 3, no. 2, pp. 200–215, 2020.
- [59] J. A. Hoeting, D. Madigan, A. E. Raftery, and C. T. Volinsky, "Bayesian model averaging: A tutorial with comments by M. Clyde, David Draper and E. I. George, and a rejoinder by the authors," *Stat. Sci.*, vol. 14, no. 4, pp. 382–417, 1999. [Online]. Available: <https://doi.org/10.1214/ss/1009212519>
- [60] Q. Tian, Z. Salcic, K. I. Wang, and Y. Pan, "A multi-mode dead reckoning system for pedestrian tracking using smartphones," *IEEE Sensors J.*, vol. 16, no. 7, pp. 2079–2093, Apr. 2016.
- [61] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," in *Proc. IEEE Int. Conf. Rehabil. Robot.*, 2011, pp. 1–7.