Mobile Networks and Applications (2020) 25:819–832
https://doi.org/10.1007/s11036-019-01302-x

Indoor Localization Using Smartphone Magnetic and Light Sensors: a Deep LSTM Approach

Xuyu Wang[1] · **Zhitao Yu**[2] · **Shiwen Mao**[2]

Published online: 14 June 2019

Abstract
With the increasing demand for location-based services, indoor localization has attracted great interest. In this paper, we present DeepML, a deep long short-term memory (LSTM) based system for indoor localization using magnetic and light sensors on smartphones. We experimentally verify the feasibility of using bimodal data from magnetic and light sensors for indoor localization for closed environments where there is no ambient light. We then design the DeepML system, which first builds bimodal images by data preprocessing, and then trains a deep LSTM network in the offline phase. Newly received magnetic field and light data are then exploited for estimating the location of the mobile device using a probabilistic method. The extensive experiments verify the effectiveness of the proposed DeepML system.

Keywords Indoor localization · Deep long short-term memory (LSTM) · Magnetic and light sensors · Visible light positioning · Fingerprinting

1 Introduction

With the rapid development of mobile technology, accurate indoor localization in the Internet of Things (IoT) is of high demand for enriching human daily life [1–3]. Many location based services emerged, such as finding an boarding gate in an airport, locating a car in a parking lot, or navigating to the nearest ATM machine in a shopping mall [4, 5]. Unlike outdoor GPS navigation systems, there are no robust indoor localization systems in the market up to now. Therefore, indoor localization has become a hot research area. Most researchers focus on WiFi based fingerprinting methods using received signal strength (RSS) [6–9] or Channel State information (CSI) [10, 11]. These methods can achieve robust meter-level accuracy but may not be effective when the signals are weak or not available in some scenarios, such as an underground parking area.

In contrast, the geomagnetic field is *omnipresent* and thus can be considered as a ubiquitous signature for indoor localization. In the past, geomagnetism basically needs to be used with special equipments for robot tracking [12] and navigation [13]. In [13], researchers employ the leader-follower model in a navigation system, where customized magnetic sensing devices are used for blind people. On the other hand, for magnetic field based localization with smartphones, the authors in [14] use mobile phones to measure magnetic field intensity and use them as magnetic signatures for identifying locations and rooms. However, according to its strategy, this system depends heavily on pillars in the building and only achieves room-level accuracy. Recently, the Magicol system combines magnetism and WiFi RSSI to build a fingerprint map, which is designed with a particle-filtering based inertial measurement unit (IMU) engine for localization and tracking [15]. Other systems based on magnetic sequences matching are proposed for improving tracking accuracy [16]. The above magnetic field based, smartphone localization systems require the user to walk around for data collection and online localization.

In addition, visible light is also omnipresent and has been exploited for indoor localization, due to the density

✉ Shiwen Mao
smao@ieee.org

Xuyu Wang
xuyu.wang@csus.edu

Zhitao Yu
zzy0021@auburn.edu

[1] Department of Computer Science, California State University, Sacramento, CA 95819, USA

[2] Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849-5201, USA

and stability of lighting infrastructures. For example, visible light intensity in an underground parking area usually does not change over time, and is not influenced by the outdoor sunlight, which can be thus leveraged for indoor localization. Existing visible light localization systems, such as polarized LEDs [17] and collocated LEDs [18], require customized LED drivers to emit identity beacons, which increases the system cost. To eliminate the need for customized LEDs, LiTell system [20] extracts high-frequency features from fluorescent light for localization. Other visible light localization systems for smartphones are based on particle-filtering and light intensity data sequence, for which there is still room for improvement by exploiting movement sensors [21, 22].

In this paper, we exploit bimodal magnetic field and ambient light data for indoor localization with a deep learning approach. The proposed scheme is motivated by the following observations. First, the magnetic field and light intensity at each location are highly stable and robust over time. Second, magnetic field and light intensity are complementary to each other at many locations. For example, magnetic field does not perform well at some locations, while these locations may have different light intensities, which can be used to distinguish them. Using the bimodal data can enhance magnetic field based indoor localization schemes. Third, using bimodal data with magnetic field and light intensity can increase the size and dimension of input data, thus improving location diversity and recognition performance. Moreover, we incorporate a deep long-short term memory (LSTM) network to train the bimodal data, which is a popular recurrent neural network (RNN) to handle time series data [23, 24]. The deep LSTM network has been successfully employed for speech recognition [25] and human activity recognition [26]. Compared to conventional fingerprinting based methods, the deep LSTM network only requires one group of weights trained for all training locations, instead of creating a database for each training location. This feature can accelerate location prediction with lower data storage requirement.

In particular, we present DeepML, a Deep LSTM network based indoor localization system using smartphone Magnetic and Light sensors. The proposed DeepML system includes a data preprocessing module for collecting magnetic field and light intensity data, and to create bimodal image data with a sliding window method. DeepML also has an offline training phase that includes feature extraction, the deep LSTM network, and a softmax classifier. A fully connected layer is incorporated to extract features from bimodal image data. The deep LSTM network consists of multiple layers of LSTM networks to achieve a stronger learning and representation ability. The softmax classifier employs the cross-entropy to measure the difference between true labeled data and the normalized output data, as well as the L2 regularization hyperparameter to avoid over-fitting. The back propagation through time (BPTT) algorithm, which is a gradient-based technique for training certain types of RNNs, is used for training the deep LSTM network. For online location prediction, an improved probabilistic method is leveraged for estimating the location of the target smartphone using newly received magnetic field and light intensity data.

The main contributions of this paper are summarized as follows.

- We experimentally validate the feasibility of using magnetic field and light intensity data for indoor localization. We show that both data are stable over time, and fusion of magnetic field and light intensity data can improve location diversity and accuracy. To the best of our knowledge, this is the first work to employ bimodal magnetic field and light intensity data for indoor localization with a deep LSTM network approach.
- We present the DeepML system design, which first builds bimodal images to train the deep LSTM network, and then employs newly received magnetic field and light data for estimating the location of the target mobile device.
- We implement the proposed DeepML system with Android smartphones, and validate its performance in two typical indoor environments with extensive experiments. DeepML outperforms the baseline scheme that uses magnetic field data only with considerable gains in all the experiments.

In the remainder of this paper, we present the preliminaries and motivation in Section 2. We discuss the DeepML design in detail in Section 3 and our performance evaluation in Section 4. Related work is discussed in Section 5 and Section 6 concludes this paper.

2 Preliminaries, motivation, and feasibility

In this section, we present the characteristics of magnetic field and light intensity data, especially on the time stability and location diversity of both data types. Then, we consider the fusion of magnetic field and light intensity data for location features.

2.1 Magnetic field characteristics

Generally, the smartphone magnetometer measures the magnetic field strength in the vector form with three elements (m_x, m_y, m_z), which describe the magnetic field data for north, east, and vertical directions, respectively. To

study the time stability and location diversity of magnetic field data, we collect magnetic field data (m_x, m_x, m_z) at 10 different locations in a 20-meter corridor in the Broun Hall in the Auburn University Campus. We also collect five different datasets for the 10 locations at five different times. Figure 1 shows the characteristics of magnetic field data (m_x, m_y, m_z) for different locations and times. The x-axis includes indexes for different locations, while the error bars in the figures indicate the variation over the five measurement times. First, we find that for any of the 10 locations, all the three elements (m_x, m_y, m_z) of magnetic field data exhibit very small changes, as indicated by the negligible error bars in the plots. This shows that magnetic field data is stable over time. This feature guarantees the reliability of fingerprinting based indoor localization using magnetic field data.

On the other hand, we investigate the diversity of magnetic field data over different locations. In Fig. 1, it is noticed that magnetic field data values are different at different local locations. For example, each of three elements (m_x, m_y, m_z) of magnetic field data are slightly different from location 1 to location 2. Specifically, we can see that at least one element of the magnetic field data is different from that of other locations for most of the 10 locations. Thus the indoor magnetic field exhibits local anomalies due to local disturbances. This is because modern buildings usually have many ferromagnetic structures. Such ambient magnetic field leads to geomagnetic anomalies, which can be leveraged for indoor localization.

2.2 Light intensity characteristics

Modern buildings nowadays deploy several types of light bulbs for indoor lighting, such as compact fluorescent lamp (CFL) and light-emitting diode (LED) [27]. Smartphones can capture light intensity from such bulbs. In fact, light propagates in the air from the bulbs to the light receiver in the smartphone. The measured radiant intensities are usually different for various locations, because of the impact of indoor environments such as shadowing, scattering, and reflection on different surfaces. Such diversity motivates the technique on light intensity based fingerprinting localization.

To study the time stability and location diversity for light intensity at different time and locations, we also measure the light intensities at the 10 different locations in the 20-meter corridor in the Broun Hall in the Auburn University campus. Similarly, we collect five different datasets of light intensity measurements a five different times. Figure 2 presents the characteristics of light intensity data at different locations and times. It is noticed that light intensity data for all locations are stable over time, as indicated by the small error bars (variations) in the figure for all the 10 locations. On the other hand, the location diversity of light intensity data is not always strong. Light intensity measurements take different values for some different locations, e.g., see neighboring locations 1 and 2, 4 and 5, and 9 and 10. But for some other neighboring locations, e.g., 3 and 4, and 7 and 8, the light intensity values are very close. Thus, unlike magnetic field data, it is harder to leverage light intensities only as fingerprints for indoor localization.

2.3 Fusion of magnetic field and light intensity data

The main motivation for fusing magnetic field data and light intensity data is that they both exhibit good time stability and different location diversity. We can exploit different light intensities at different locations to improve a magnetic field based indoor localization scheme. By fusing the magnetic field and light intensity data, the dimension of input data for the deep LSTM network is increased, thus strengthening the uniqueness of location features.

To assess the location diversity achieved by magnetic field, light intensity, and fusion of magnetic field and light

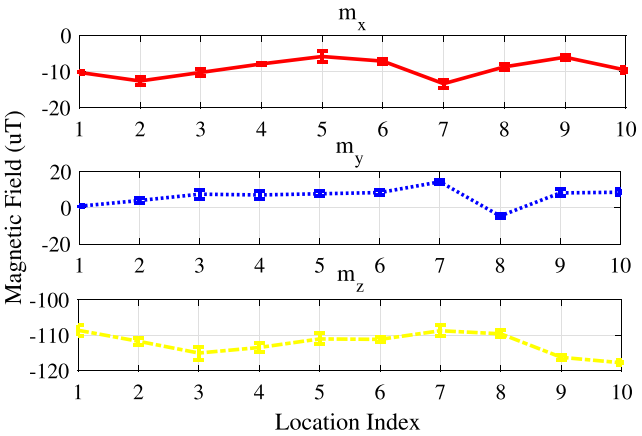


Fig. 1 Characteristics of the magnetic field data

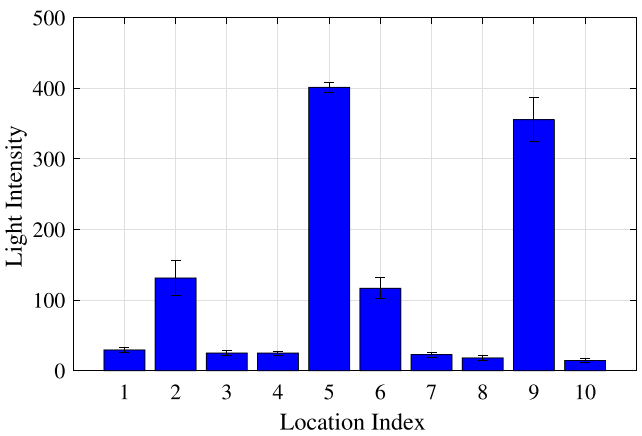


Fig. 2 Characteristics of the light intensity data

intensity, we define a confusion matrix for N different locations, given by

$$D = \begin{bmatrix} d_{11} & d_{12} & d_{13} & \dots & d_{1N} \\ d_{21} & d_{22} & d_{23} & \dots & d_{2N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{N1} & d_{N2} & d_{N3} & \dots & d_{NN} \end{bmatrix}, \quad (1)$$

where d_{ij} denotes the Euclidean distance between the signal vectors for location i and j, which can be computed as in Eq. 2, where $\mathbf{s}_i$ denotes the signal vector collected at location i (i.e., the magnetic field vector, the light intensity, or both). In particular, the fusion of the magnetic filed vector and the light intensity is that $\mathbf{s} = [m_x, m_y, m_z, l]$, where l is the light intensity sample. To measure the location diversity for the three types of data, we need to normalize the confusion matrix for each of the three different methods into the same range.

$$d_{ij} = |\mathbf{s}_i - \mathbf{s}_j|_2 = \sqrt{(m_{1x} - m_{2x})^2 + (m_{1y} - m_{2y})^2 + (m_{1z} - m_{2z})^2 + (l_1 - l_2)^2}. \quad (2)$$

Figure 3 presents the three confusion matrices for data collected at the 10 locations in the corridor, including (i) Magnetic field vector, (ii) Light intensity, (iii) Magnetic field vector & Light intensity. From Fig. 3a, we find that the magnetic field data for most location pairs exhibit some variations, but the magnitude of change in the confusion matrix is usually small. From Fig. 3b, we can see that the light intensity data for some locations have great variations, but most of the location pairs have similar measurements. These two confusion matrices lead to the same conclusion as that in the previous study of magnetic field and light intensity (see Sections 2.1 and 2.2). In Fig. 3c, the confusion matrix for the fused magetic field and light intensity data is plotted. It can be seen that the fusion of magnetic field and light intensity has achieved great location diversity; there is a large distance range for most location pairs. The fusion of two types of sensory data can thus enhance the training and location estimation for deep LSTM based indoor localization.

3 The DeepML system design

3.1 System architecture

We design and prototype the DeepML system with Samsung Galaxy S7 Edge smartphone on the Android 7.0 platform. An Android application is generated by Android Studio 2.3.3 on the phone for data collection and preprocessing. Moreover, the proposed DeepML system employs the ambient light and magnetic field data for two main reasons. First, according to the study in the previous section, both magnetic field data and light intensity data exhibit strong time stability. Therefore they are both suitable for fingerprinting based localization (i.e., no big change from the training phase to the testing phase). Second, the location diversities of light intensity and magnetic field are both limited. But these two types of data are complementary to each other for most locations, as shown in Fig. 3c. Fusing these two types of data could lead to great improvements for indoor fingerprint.

The architecture of the proposed DeepML system is presented in Fig. 4. DeepML first collects magnetic field data and light intensity data from the smartphone sensors. It then executes data preprocessing to process the collected magnetic field and light data, to build a bimodal image data using a sliding window based method. Then, Offline training and online position prediction are the next to components in the DeepML system. At the offline stage (see Section 3.3), feature extraction is performed for the bimodal data, for effectively training the deep LSTM network. Compared to conventional fingerprinting based methods, DeepML does not need to set up a database for all training locations, which is used to store either features or raw

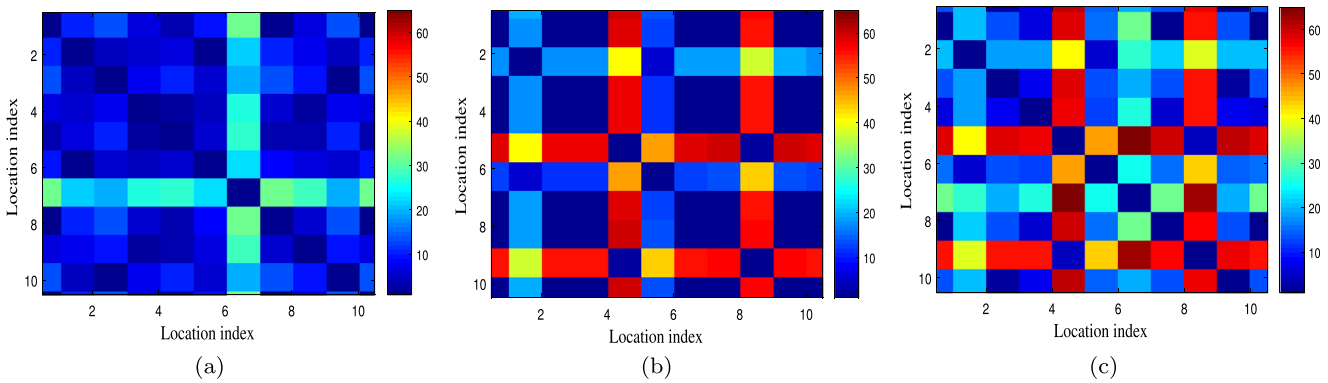


Fig. 3 Three confusion matrices for 10 locations in a corridor: **a** magnetic field vector, **b** light intensity, **c** magnetic field vector & light intensity

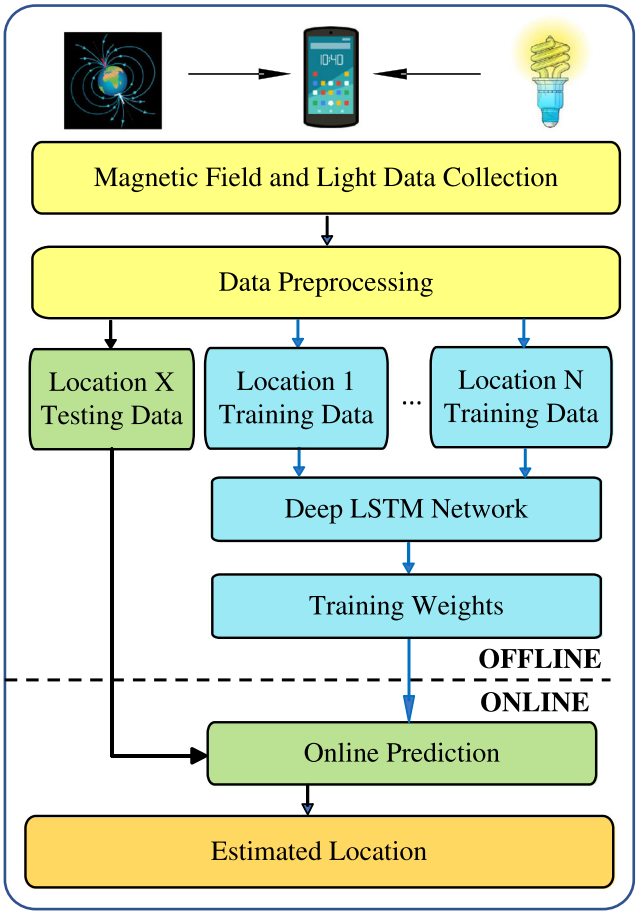


Fig. 4 The DeepML system architecture

data as fingerprints. However, DeepML only requires one group of weights trained for all training locations. At the online stage, new magnetic field and light intensity data are collected by a smartphone, which are then processed and fed into the trained deep LSTM network to identify the location of the smartphone. We propose an improved probabilistic approach for location estimation in the online stage, which will be presented in detail in Section 3.4.

3.2 Data preprocessing

In the data preprocessing module, the ambient light and magnetic field sensors are activated to collect real-time sensory data. Due to the requirement on the input sequence size of the LSTM network, we need to reduce the sampling rate of the magnetic field sensor so that it can synchronize with the ambient light sensor. Specifically, we collect 1,500 rows of light intensity and magnetic field samples for each training location (each row consists of $\mathbf{s} = [m_x, m_y, m_z, l]$). As will be discussed in Section 4.1, we carried our experiments with DeepML in a computer lab environment and a corridor environment. Therefore, the size of training data is 15,000 rows in total for the 10 training locations in the corridor, and 18,000 rows in total for the 12 train locations in the lab. For location estimation in the online prediction stage, we collect 400 rows of sensory data for each test location.

Once the sensory data are collected, we apply a sliding window to construct bimodal images from the collected magnetic field and light intensity data. For both training and testing, we set the size of the sliding window to 20. Specifically, we construct bimodal images with size 20×4, with 20 rows and each row consists of four elements m_x, m_y, m_z, and l.

3.3 Offline training

We propose a deep LSTM to learn the location features from collected bimodal sensory data. In the offline training phase, we use the bimodal data collected from the known training locations, to train the deep LSTM network. The offline training module includes feature extraction, deep LSTM network and Softmax classifier. The last two modules are shown in Fig. 5.

3.3.1 Feature extraction

For better training the deep LSTM network, we first incorporate a neural network of one fully connected layer to extract features from the raw magnetic field and light intensity data, which is formulated by

$$z_t = \text{ReLU}(\mathbf{W}\mathbf{x}_t + \mathbf{b}), \tag{3}$$

where $\mathbf{x}_t$ and z_t are the input and output of the fully connected layer, respectively, $\mathbf{W}$ and $\mathbf{b}$ are the weight and bias vectors of the fully connected layer, and $\text{ReLU}(\cdot)$ is the

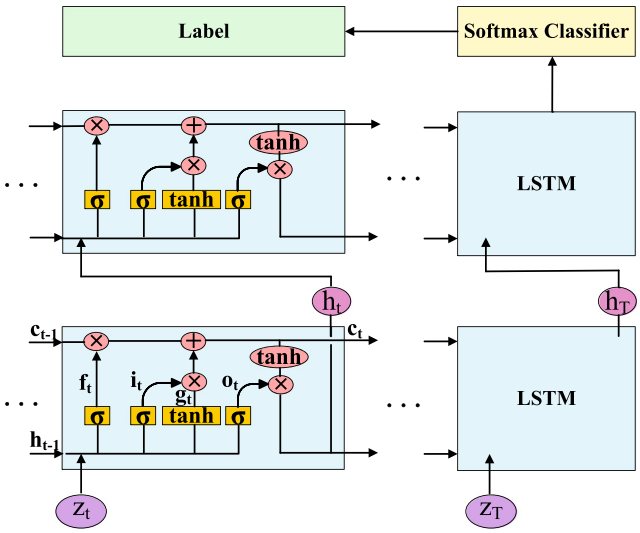


Fig. 5 The Deep LSTM network model for offline training

rectified linear unit, which is the activation function defined as

$$\mathrm{ReLU}(x) = \max(x, 0). \tag{4}$$

The ReLU$(\cdot)$ function has several advantages such as sparse representation, efficient gradient propagation, and low computational complexity.

3.3.2 Deep LSTM network

After extracting features from magnetic field and light intensity sensory data, we feed the output from the fully connected layer to a deep LSTM network to train it weights. LSTM is a popular recurrent neural network (RNN) that are suited to handle time series data and capture the long-range dependencies in the data series. LSTM can solve the problems of exploding or vanishing gradients found in RNNs. Moreover, LSTM can exploit temporal information of the magnetic field and light intensity data through recursively mapping the input sequence to output label using the hidden LSTM units. Each LSTM unit has a built-in memory cell to store information over time with non-linear gate units, which can control the changing values and memory contents.

We utilize a deep LSTM network to determine a mapping from a feature based on slot scheme $\mathbf{z} = (z_1, z_2, ..., z_T)$ to an output label y by computing the network unit activations from $t = 1$ to $t = T$:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{ix}\mathbf{z}_t + \mathbf{W}_{im}\mathbf{h}_{t-1} + \mathbf{b}_i) \tag{5}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{fx}\mathbf{z}_t + \mathbf{W}_{fm}\mathbf{h}_{t-1} + \mathbf{b}_f) \tag{6}$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{ox}\mathbf{z}_t + \mathbf{W}_{om}\mathbf{h}_{t-1} + \mathbf{b}_o) \tag{7}$$

$$\mathbf{g}_t = \tanh(\mathbf{W}_{cx}\mathbf{z}_t + \mathbf{W}_{cm}\mathbf{h}_{t-1} + \mathbf{b}_c) \tag{8}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t \tag{9}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \tag{10}$$

where the $\mathbf{W}$ terms denote the matrices of weights, $\mathbf{b}$ terms denote the bias vecotrs, $\sigma(x) = 1/(1 + e^{-x})$ is the sigmoid function with outputs in [0,1], $\tanh(x) = (e^x - e^{-x})/(e^x + e^{-x})$ is the hyperbolic tangent function with outputs in [-1,1], $\mathbf{i}$, $\mathbf{f}$, $\mathbf{o}$, $\mathbf{g}$, and $\mathbf{c}$ are the input gate, forget gate, output gate, candidate values, and cell activation, respectively, $\odot$ denotes the element-wise product of vectors, and $\mathbf{h}$ is the cell output activation vector.

In LSTM, the input gate $\mathbf{i}$ determines how much new information will be used in the current memory cell, the forget gate $\mathbf{f}$ controls how much information will be removed from the old memory cell, and the output gate $\mathbf{o}$ decides how much data will be output based on the current memory cell $\mathbf{c}$. Moreover, the sigmoid function $\sigma(\cdot)$ can control how much information should be updated and the hyperbolic tangent function $\tanh(\cdot)$ can create new candidate values $\mathbf{g}$. Compared with RNN, LSTM can handle long-term dependency and has better data representation ability.

For the proposed DeepML system, we propose to stack two or more layers of the LSTM network to construct a deep LSTM network, for a higher learning and representation ability for magnetic and light sensor data. A two-layer example is shown in Fig. 5, while in our experimental study, we find a four-layer deep LSTM network can produce the smallest localization error.

3.3.3 Softmax classifier

In DeepML, the output of the final cell's hidden state in the second LSTM network is connected to another fully connected layer, which is a basic neural network with one hidden layer to train the output data using a softmax classifier. The input data to the softmax classifier is an N-element vector $\mathbf{q} = [q_1, q_2, ..., q_N]$, where N is the number of training locations. The ith input data element to the softmax classifier can be obtained by $q_i = \mathbf{h}_f^T \mathbf{w}_i$, where $\mathbf{h}_f$ is the output vector of the final cell's hidden state in the second LSTM network, and $\mathbf{w}_i$ is the weighting vector for the fully connected layer. Then, the softmax function maps the N-element vector $\mathbf{q}$ to a normalized vector $\mathbf{p} = [p_1, p_2, ..., p_N]$, i.e.,

$$p_i = \frac{e^{q_i}}{\sum_{n=1}^{N} e^{q_n}} = \frac{e^{\mathbf{h}_f^T \mathbf{w}_i}}{\sum_{n=1}^{N} e^{\mathbf{h}_f^T \mathbf{w}_n}}, \text{ for } i = 1, 2, ..., N. \tag{11}$$

Let $J(\mathbf{W})$ denote the loss function with weight parameter set $\mathbf{W}$. We adopt the cross-entropy to measure the difference between the label (i.e., the known training locations) and the output. We also use the L2 regularization hyperparameter to avoid over-fitting. To obtain the optimal weights, the training process is to minimize the loss function, as

$$\min_{\mathbf{W}} J(\mathbf{W}) = -\sum_{i=1}^{N} y_i \log(p_i) + \frac{\lambda}{2} \|\mathbf{W}\|_2^2, \tag{12}$$

where y_i is the label for location i, and λ is the L2 regularization hyperparameter. Then, we can train the deep LSTM network using BPTT to determine the weights. We also apply the Adam Optimizer for improving the efficiency of optimization [28].

3.4 Online testing

In the online testing phase, newly collected magnetic field and light intensity data from a mobile device is first processed in the data preprocessing mobule. Then M bimodal images are then constructed with the processed data, each of which has the same size as training images. The newly constructed bimodal image data are then fed into the trained deep LSTM network to extract location features.

We propose a probabilistic method for estimating the location of the mobile device after passing the M newly collected bimodal images through the trained deep LSTM network. Generally, the probabilistic method can obtain better accuracy of indoor localization than the deterministic method [7], which can leverage data statistical characteristic for the location estimation. Let $\boldsymbol{\Xi}$ denote the output results from softmax classifier using deep LSTM network for N training locations with M bimodal images, given by

$$\boldsymbol{\Xi} = \begin{bmatrix} \xi_{11} & \xi_{12} & \xi_{13} & \cdots & \xi_{1M} \\ \xi_{21} & \xi_{22} & \xi_{23} & \cdots & \xi_{2M} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \xi_{N1} & \xi_{N2} & \xi_{N3} & \cdots & \xi_{NM} \end{bmatrix}, \tag{13}$$

where each column j is the softmax classifier output for the jth input bimodal image. We then compute the row-averge for $\boldsymbol{\Xi}$, each for a training location, to reduce the variance of the output results. Let $\bar{\xi}_i$ denote the mean for the softmax classifier output vector $[\xi_{i1}, \xi_{i2}, ..., \xi_{iM}]$. The mean vector can be obtained by $\bar{\xi} = [\bar{\xi}_1, \bar{\xi}_2, ..., \bar{\xi}_N]^T$, where $\bar{\xi}_i$ is the probability for the mobile device to be located at training location i, $i = 1, 2, ..., N$.

Finally, the position of the mobile device, denoted by $\hat{\mathcal{L}}$, can be estimated as a weighted average of all the N known training locations, which is given by

$$\hat{\mathcal{L}} = \sum_{i=1}^{N} \mathcal{L}_i \cdot \bar{\xi}_i, \tag{14}$$

where $\mathcal{L}_i$ is the ith training position.

4 Experimental study

4.1 Experiment configuration

We prototype the DeepML system with the Samsung Galaxy S7 Edge smartphone, which is based on the Android 7.0 platform. Moreover, we implement an android application with Android Studio 2.3.3 on the smartphone for data collection and preprocessing. For data collection, the ambient light sensor and the magnetic field sensor in the smartphone are used to collect real-time smaples. For offline training, we collect 15,000 samples for 10 training locations in the corridor experiment and 18,000 samples for 12 training locations in the lab experiment as training data. For online test, we collect 400 samples from a smartphone put at an unknown location, which is always different from a training location. We develop deep LSTM model using Tensorflow on a computer with a 4720HQ CPU, and then combine it with the data collection android application, thus achieving indoor localization in real-time.

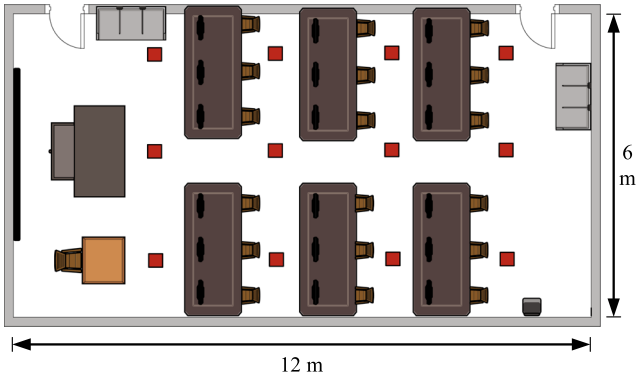


Fig. 6 Layout of the computer laboratory: training locations are marked as red squares

For performance comparison, we implement a simplified version of DeepML that only uses magnetic field data, termed "Magnetic only." To guarantee a fair comparison, we implement these two approaches with the same magnetic field dataset and the same deep LSTM parameters. We experiment with DeepML and Magnetic only in two different indoor scenarios as follows.

Computer laboratory We choose a 6 × 12 m^2 computer laboratory in the Broun Hall in the Auburn University campus. The laboratory is a cluttered environment, with many computers, wooden tables and chairs. Most of the line-of-sight (LOS) paths are blocked in this environment. The floor plan is shown in Fig. 6. We choose 12 training locations, which are marked as red squares in the floor plan. The distance between two neighboring training locations is 1.6 m. We collect 1,500 rows of light intensity and magnetic field combined data (in the form of $[m_x, m_y, m_z, l]$) for each training location, and 400 rows of data for each test location.

Corridor This is a long, empty corridor in Broun Hall with dimension 2.4 × 20 m^2. As shown in Fig. 7, we choose 10 training locations (also marked as red squares) along a straight line. We also set the distance between two adjacent training locations to 1.6 m. The LOS component is dominant in this case due to absence of obstacles. Similarly, 1,500 samples are collected for each training location and 400 samples are collected for each test location.

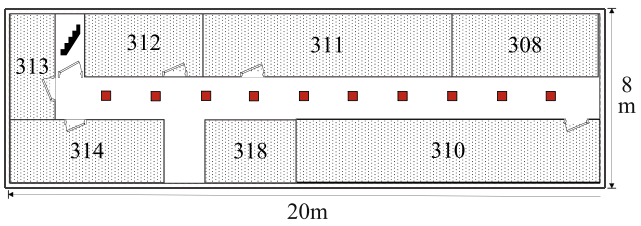


Fig. 7 Layout of the corridor: training locations are marked as red squares

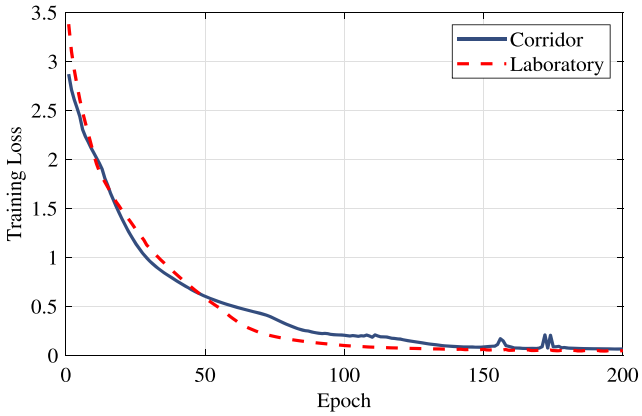


Fig. 8 Training losses for the laboratory and corridor experiments

4.2 Accuracy of location estimation

Figure 8 presents the training loss over epochs of deep LSTM for the corridor and computer laboratory scenarios. Obviously, it can be seen that the training loss curve of laboratory experiments starts to converge after 100 epochs and eventually descends to less than 0.01 after 200 epochs. For the corridor experiments, even though with a little fluctuation, we can see that the training loss curve starts to converge after 120 epochs and finally drops to lower than 0.01 after 200 epochs.

Figure 9 presents the cumulative distribution function (CDF) of localization errors of DeepML and Magnetic only for the laboratory experiment. For this environment with complex light intensity and magnetic field distribution, DeepML successfully leverages the bimodal light-magnetic features to predict the location of the mobile device, which are different for various positions. Figure 9 shows that 58% of the location errors for the proposed DeepML system are under 0.5 m. For Magnetic only, only 20% of the location errors are less than 0.5 m. Moreover, DeepML has 82% of the test locations with a location error less than or equal to 2 m. For Magnetic only, only 50% of the test locations have a location error smaller than or equal to 2 m. The maximum error of DeepML is 3.7 m, while the maximum error for Magnetic only is 5m. Apparently, the proposed DeepML system is more suitable for the complex laboratory environment.

Figure 10 shows the CDF of localization errors of the two schemes in the corridor scenario. There are about 65% of test locations that have an estimation error less than or equal to 0.4 m for DeepML, while the magnetic only scheme has 25% of test locations having errors less than or equal to 0.4 m. DeepML has 87% of test locations achieving an error under 3 meters, compared to 78% for the magnetic only scheme. Moreover, for the long corridor scenario, the maximum location errors for DeepML and the magnetic only scheme are 6.5 meters and 8.2 meters, respectively. Obviously, the proposed DeepML system exhibits higher robustness than the baseline scheme. These two experiments demostrate that light intensity and magnetic field are really complementary to each other; fusing these two types of data improves the indoor localization performance.

4.3 Impact of different design parameters

4.3.1 Impact of test data size

We evaluate the proposed DeepML system with test data size of 120, 180, 240, 300, and 360, to examine how test data size impacts the accuracy of indoor localization. Considering variable control, we fix all other parameters for every test data size. The epoch, window size, hidden units, and batch size are set to 200, 10, 40, and 1500, respectively. As shown in Fig. 11, distance errors of both scenarios are almost the same for different test data sizes. Specifically, the mean distance errors in the lab and corridor settings are 0.98 m and 1.36 m, respectively, which indicate that the localization performance of DeepML is sufficiently robust

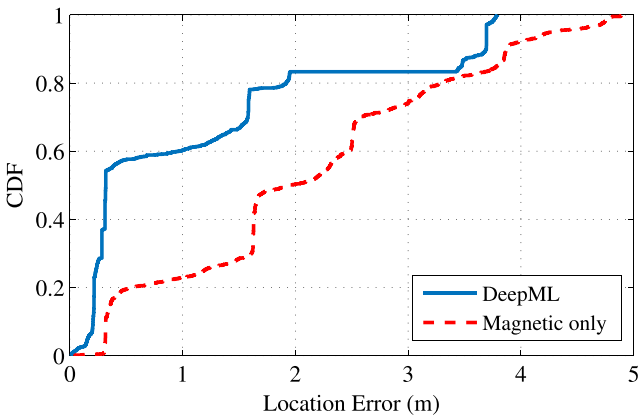


Fig. 9 CDF of localization errors for the laboratory experiment

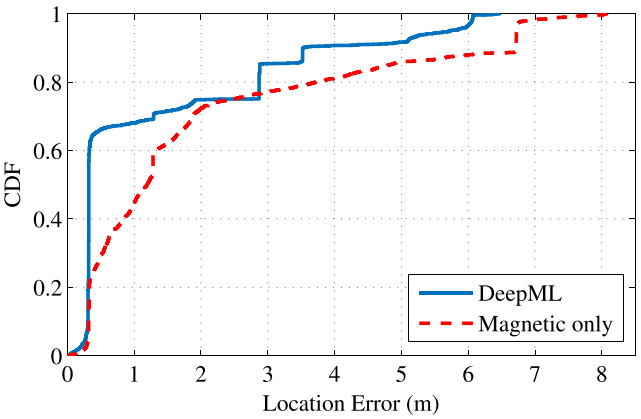


Fig. 10 CDF of localization errors for the corridor experiment

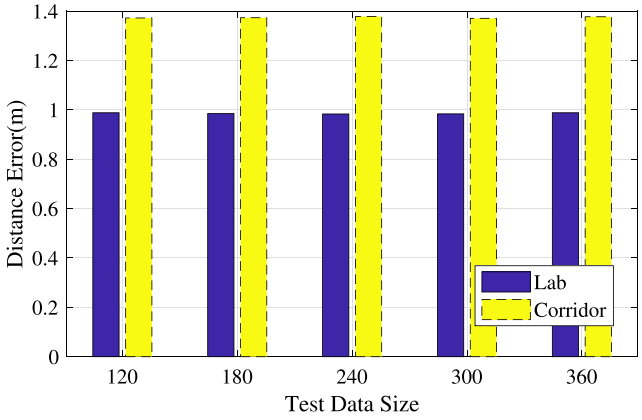


Fig. 11 Mean distance errors for different sizes of test data

for various test data sizes. DeepML could also achieve higher precision if more data samples and training locations are used. The computation time is less than 0.1 s for each test data size, which is sufficient for real-time localization (e.g., tracking and navigation).

4.3.2 Impact of the number of LSTM layers

To evaluate how the number of LSTM layers affects the mean distance error, we build six deep LSTM models that have 1, 2, 3, 4, 5, and 6 layers, respectively, and evaluate their performance under the same setting (i.e., the lab and corridor scenarios). The mean distance errors achieved by the six deep LSTM models are presented in Fig. 12. We find in general the mean distance error decreases as more layers are used. The smallest mean errors, i.e., 1.07 m for the lab setting and 1.39 m for the corridor setting, are achieved when the number of LSTM layers is 5 and 4, respectively. It can been seen that the biggest reduction in mean error happens when the number of layers is increased from one to two. Beyond two layers, there are still some small improvements as more layers are added.

Figure 13 presents the training time across all datasets with different numbers of LSTM layers. It clearly shows that the number of LSTM layers has a great impact on the training time. The training time for the lab case is always longer than that for the corresponding corridor case. Considering the localization performance shown in Fig. 12, we choose to have a model with 4 layers for the best location accuracy and a moderate training time. Another reason to consider is the possibility of overfitting. It is well known that a very large number of neural network layers may lead to overfitting. Note the slight increasing trend of the mean distance error when the number of layers is increased beyond 4 in Fig. 12, which may be caused by overfitting. To maintain the robustness of DeepML, setting the number of LSTM layers to 4 should be a good and safe choice.

4.3.3 Impact of hidden units

The number of hidden units represents the dimensionality of the "hidden state" in the LSTM network. In order to examine the effect of hidden units on localization accuracy, we implement the DeepML model with five different numbers of hidden units. Keeping all the other settings the same as in previous sections, we compare the localization errors achieved by the five deep LSTM models and the corresponding training times.

From Fig. 14, it can be seen that the mean distance error decreases as the number of hidden units is increased. The best localization performance is achieved with 40 hidden units for both lab and corridor scenarios, which are 1.02 m for the lab setting and 1.33 m for the corridor setting. In addition, the largest mean distance errors, i.e., 1.39 m for the lab setting and 1.93 m for the corridor setting, are achieved with 5 hidden units. The localization accuracy gap between the lab and corridor settings is from 0.31m to 0.54 m. This is because the lab environment is more complicated so that the light intensity and magnetic field data have more diversity

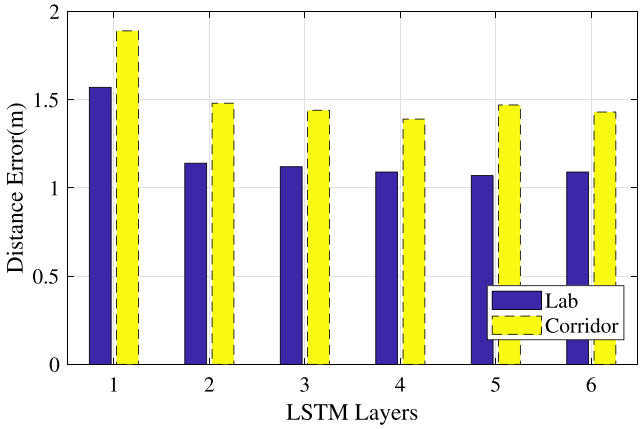


Fig. 12 Mean distance errors for different numbers of LSTM layers

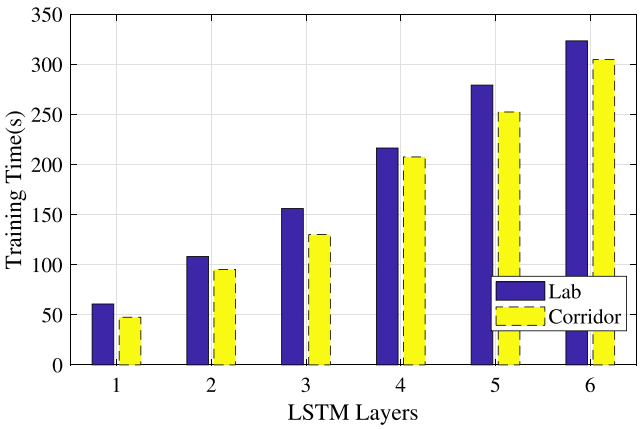


Fig. 13 Mean training time for different numbers of LSTM layers

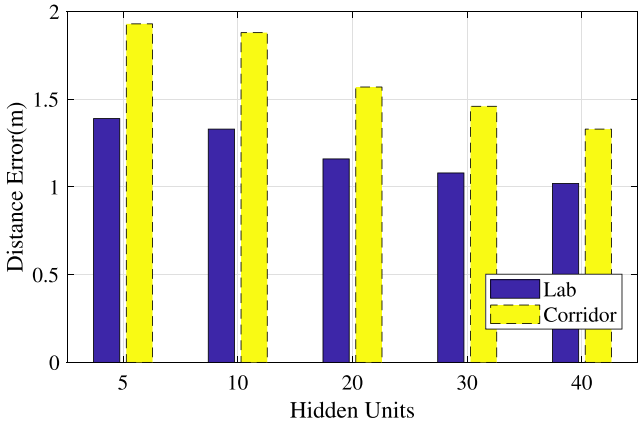


Fig. 14 Mean distance errors for different numbers of hidden units

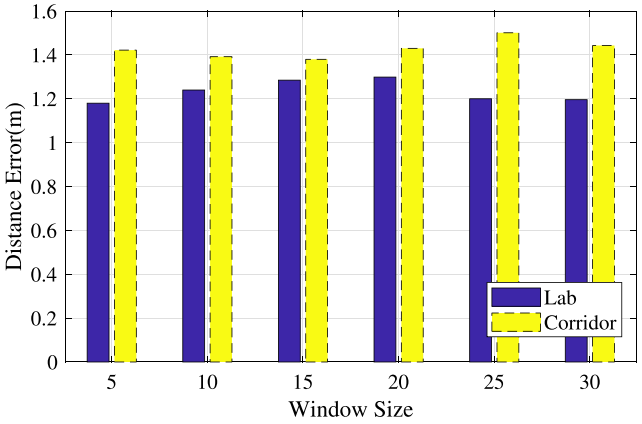


Fig. 16 Mean distance errors for different window sizes

than the corridor case. The proposed DeepML system can exploit such diversity for better performance.

The corresponding training times are presented in Fig. 15 for different numbers of hidden units. Obviously, the training time is positively correlated with the number of hidden units. Although the longest training time has reached 520.72 s, we still choose the design with 40 hidden units for its best localization accuracy.

4.3.4 Impact of window size

We next examine the impact of window size on DeepML's performance. Window size here is the length of a sliding window that defines a time sequence of data. Figure 16 shows that similar results are achieved for different window sizes. Although there are small fluctuation of mean distance error with six different window sizes, the difference between the best and worst localization errors are only 0.108 m for the corridor setting and 0.103 m for the lab setting. Considered that the lab and the corridor are 6×12 m^2 and 2.4×20 m^2, respectively, a 0.1 meter-level difference is negligible. We conclude that the window size does not greatly affect the localization performance of DeepML.

Figure 17 presents the mean training time with different window sizes. The training time quickly increases with the window size, and is over 500 s for the lab setting when the window size is 30. In fact, the training time does not affect the user experience much in practice it is a one-time cost at the offline stage. Therefore, localization accuracy should be considered first. The design with a window size of 5 is adopted in DeepML considering both localization accuracy and training time.

4.3.5 Impact of batch size

In DeepML, batch size is the size of the subset of training samples We examine the impact of batch size on localization accuracy under the two environments. Figure 18 illustrates the mean distance errors for increased batch size in the lab and corridor scenarios. The mean distance error become larger as the batch size is increased from 1,500 to 3,500. The best localization performance, i.e., 1.08 m for the lab

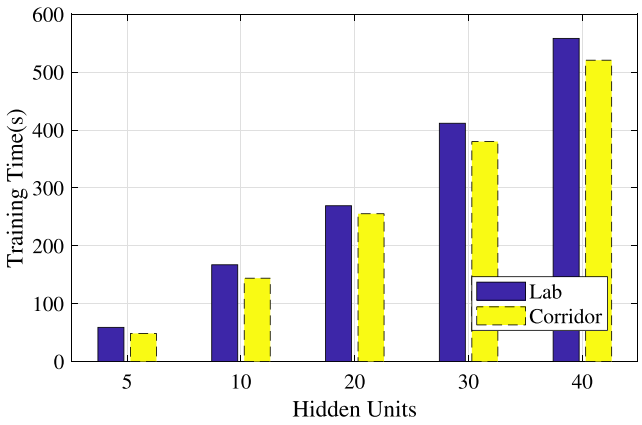


Fig. 15 Mean training time for different numbers of hidden units

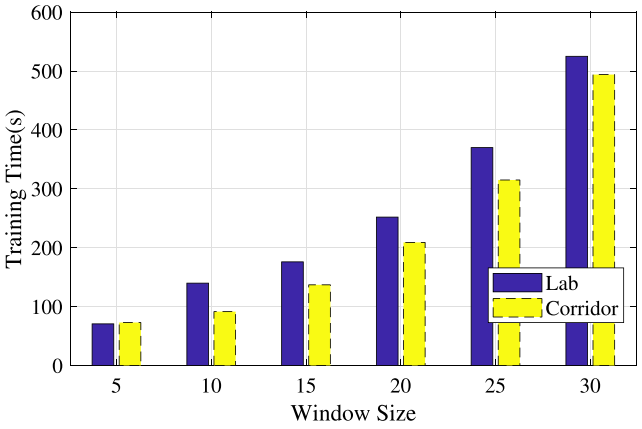


Fig. 17 Mean training time for different window sizes

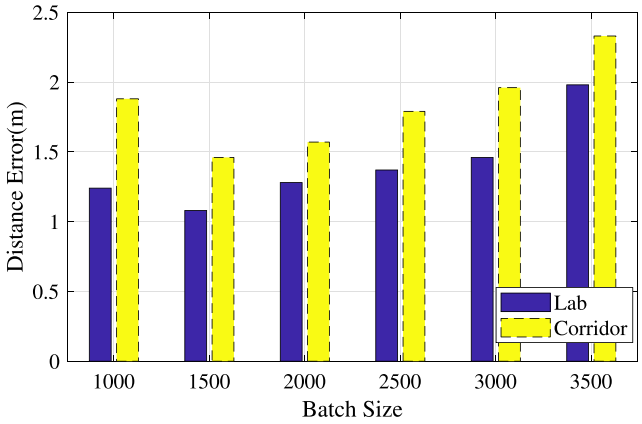


Fig. 18 Mean distance errors for different batch sizes

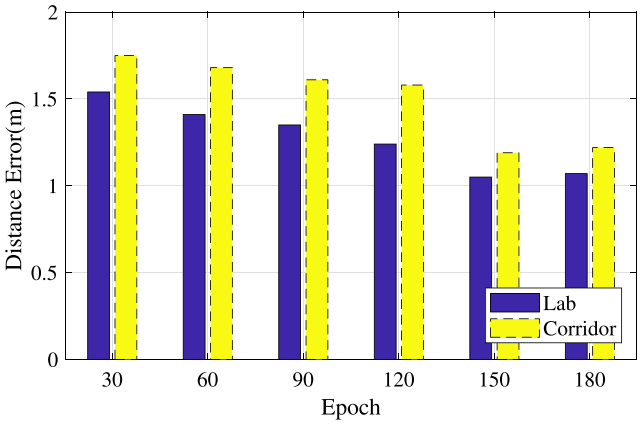


Fig. 20 Mean distance errors for different epochs

setting and 1.46 m for the corridor setting, is achieved when the batch size is 1500.

Figure 19 presents the training time for different batch sizes. The longest training times are 124.276 s and 128.64 s for the lab and the corridor setting, respectively, while the smallest training times are 106.70 s and 99.42 s for the lab and the corridor settings, respectively. This indicates that the batch size only has a small influence on the LSTM network's training time.

4.3.6 Impact of batch epoch

When training a neural network, an epoch means one pass of the full training dataset forward and backward through the neural network. Figure 20 shows the impact of the number of epochs on the mean distance errors. Obviously, The largest error is achieved when the number of epochs is 30 in both the lab and corridor environments. Moreover, the mean error keeps on decreasing as the number of epochs is increased. We can see that the error becomes steady when there are 150 or more epochs. The lowest error is achieved at 150 epochs from Fig. 20.

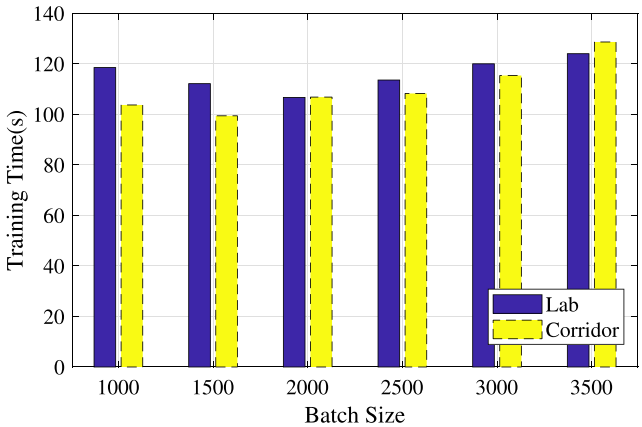


Fig. 19 Mean training time for different batch sizes

Figure 21 presents the training time for different numbers of epochs. The training time increases as the number of epochs is increased in both scenarios. The result is obvious because more epochs will lead to longer training time. To reach the lowest distance error, it takes 429.73 s and 408.51 s to train the deep LSTM network in the lab and corridor settings, respectively.

5 Related work

Indoor localization have been a hot research topic in recent years [29–32]. In this section, we mainly review two types of indoor localization systems that are directly related to the DeepML system, i.e., deep learning based and visible light based systems.

Deep learning based indoor localization systems leverage deep networks to better learn the location features for fingerprints in the offline training phase and then predict the location of a mobile device based on newly collected data in the online test phase [33]. Three types of deep networks have been used for indoor localization, including deep

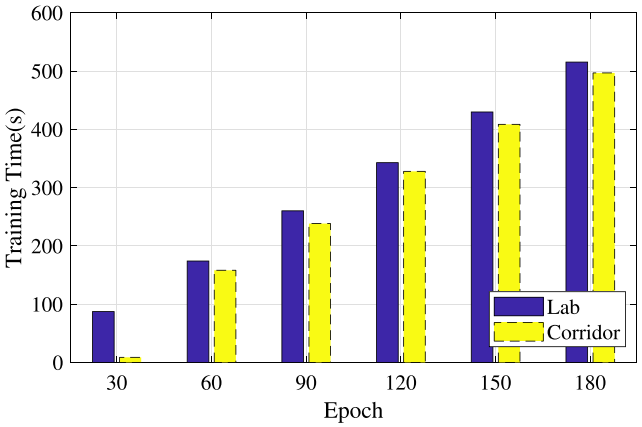


Fig. 21 Mean training time for different epochs

autoencoder networks, deep convolution neuron networks (CNN), and LSTM networks [33]. DeepFi is the first autoencoder based work for indoor localization utilizing CSI amplitudes [11, 34]. In addition, PhaseF [35, 36] and BiLoc [37] exploit calibrated CSI phase and bimodal data, respectively for improved performance. Moreover, deep autoencoder networks-based indoor localization systems with Bluetooth Low Energy (BLE) [38], dead reckoning data [39], and WiFi RSS data [40, 41] have also been proposed.

On the other hand, CiFi is the first system that incorporates a deep CNN for indoor localization, which employs CSI images created by estimated phase difference values of 5 GHz WiFi to train the deep CNN [42, 43]. To further improve the localization accuracy, the ResLoc system utilizes a deep residual sharing learning model to train the bimodal CSI tensor data, and achieves the smallest localization errors comparing to the several existing deep learning based approaches that utilize CSI data [44]. To the best of our knowledge, the proposed DeepML system is the first to employ deep LSTM for indoor localization, and exploits magnetic and light bimodal data for indoor localization [45]. In another recent work [46], the authors also adopts deep LSTM for WiFi RSS based indoor localization and test its performance with a WiFi fingerprint dataset.

Visible light based indoor localization systems can be roughly classified into three categories according to the beacon device: light-emitting diodes (LEDs) based, fluorescent lights (FLs) based, and pervasive light sources based schemes. For LED based systems, ByteLight leverages the rolling shutter effect of complementary metal oxide semiconductor (CMOS) camera to generate unique image patterns with customized LEDs for building fingerprinting maps [47]. Epsilon requires customized light sensors and assigns unique beacons to LEDs. Then, location can be estimated by using RSS-based methods [19]. In addition, Luxapose uses a camera as the receiver to calculate the location by using an angle-of-arrival (AOA) based method [18]. In [17], the authors propose polarization based modulation on LEDs to achieve sub-meter accuracy. Compared with LEDs, the biggest advantage of fluorescent lights is universality. The LiTell localization system employs distinguishable and stable characteristic frequency of unmodulated FLs as location landmarks and commercial off-the-shelf (COTS) smartphones cameras as light sensors [20].

Unlike the two types of aforementioned VLP systems, pervasive light sources based systems basically have no light source limitation and mostly use light intensity to create fingerprints, which means there is no need for customized devices on both sides. In [22, 48], the authors leverage the vector of multiple light intensity values collected by a smartphone when the user is walking around to create a fingerprint map. In this work, the data collected from inertial measurement units is also combined with light intensity to overcome the weakness of the low diversity of light intensity at different locations. Our DeepML system integrates magnetic field and ambient light data for indoor localization with a smartphone implementation, which does not require any customized devices. Moreover, the deep LSTM incorporated in DeepML is also highly effect in learning the location features from the relatively large bimodal data, and thus guaranteeing a superior localization performance.

6 Conclusions

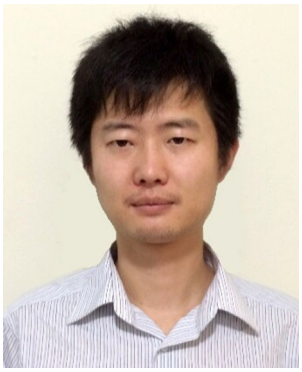
In this paper, we presented DeepML, a deep LSTM based system for indoor localization using magnetic and light sensors on smartphones. We experimentally verified the feasibility of using bimodal data from magnetic and light sensors for indoor fingerprinting. We then presented the DeepML system design, where data preprocessing was implemented to obtain bimodal image data, and a four-layer LSTM network was then trained. For online test, newly received magnetic field and light data were used for estimating the location of the mobile device. The extensive experiments demonstrated the effectiveness of the proposed DeepML system.

Acknowledgments This work is supported in part by the NSF under Grant CNS-1702957, and by the Wireless Engineering Research and Education Center (WEREC) at Auburn University.

References

1. Khelifi F, Bradai A, Benslimane A, Rawat P, Atri M (2018) A survey of localization systems in internet of things. Springer Mobile Networks and Applications Journal, pp 1–25
2. Kumarasiri R, Alshamaileh K, Tran NH, Devabhaktuni V (2016) An improved hybrid RSS/TDOA wireless sensors localization technique utilizing Wi-Fi networks. Springer Mobile Networks and Applications 21(2):286–295
3. Kan C, Ding G, Wu Q, Zhang T (2018) Robust localization with crowd sensors: a data cleansing approach. Springer Mobile Networks and Applications 23(1):108–118
4. Gu Y, Lo A, Niemegeers I (2009) A survey of indoor positioning systems for wireless personal networks. IEEE Commun Surveys Tuts 11(1):13–32
5. Wang X, Mao S, Pandey S, Agrawal P (2014) CA2T: Cooperative antenna arrays technique for pinpoint indoor localization. In: Proc MobiSPC 2014, Niagara Falls, Canada, pp 392–399
6. Bahl P, Padmanabhan VN (2000) Radar: an in-building RF-based user location and tracking system. In: Proc IEEE INFOCOM'00, Tel Aviv, Israel, pp 775–784
7. Youssef M, Agrawala A (2005) The Horus WLAN location determination system. In: Proc ACM MobiSys'05, Seattle, WA, pp 205–218

8. Caso G, De Nardis L (2017) Virtual and oriented WiFi fingerprinting indoor positioning based on multi-wall multi-floor propagation models. Springer Mobile Networks and Applications 22(5):825–833
9. Liu H-H (2017) The quick radio fingerprint collection method for a WiFi-based indoor positioning system. Springer Mobile Networks and Applications 22(1):61–71
10. Xiao J, Wu K, Yi Y, Ni L (2012) FIFS: Fine-grained indoor fingerprinting system. In: Proc IEEE ICCCN'12, Munich, Germany, pp 1–7
11. Wang X, Gao L, Mao S, Pandey S (2017) CSI-Based fingerprinting for indoor localization: a deep learning approach. IEEE Trans Veh Technol 66(1):763–776
12. Chung J, Donahoe M, Schmandt C, Kim I-J, Razavai P, Wiseman M (2011) Indoor location sensing using geo-magnetism. In: Proc ACM MobiSys'11, Bethesda, MD, pp 141–154
13. Storms W, Shockley J, Raquet J (2010) Magnetic field navigation in an indoor environment. In: Proc IEEE UPINLBS'10, Kirkkonummi, Finland, pp 1–10
14. Gozick B, Subbu KP, Dantu R, Maeshiro T (2011) Magnetic maps for indoor navigation. IEEE Trans Instrum Meas 60(12):3883–3891
15. Shu Y, Bo C, Shen G, Zhao C, Li L, Zhao F (2015) Magicol: Indoor localization using pervasive magnetic field and opportunistic WiFi sensing. IEEE J Sel Areas Commun 33(7):1443–1457
16. Ma Y, Dou Z, Jiang Q, Hou Z (2016) Basmag: an optimized HMM-based localization system using backward sequences matching algorithm exploiting geomagnetic information. IEEE Sensors J 16(20):7472–7482
17. Yang Z, Wang Z, Zhang J, Huang C, Zhang Q (2015) Wearables can afford: Light-weight indoor positioning with visible light. In: Proc ACM Mobisys'15, Florence, Italy, pp 317–330
18. Kuo Y-S, Pannuto P, Hsiao K-J, Dutta P (2014) Luxapose: Indoor positioning with mobile phones and visible light. In: Proc ACM MobiCom'14, Maui, HI, pp 447–458
19. Li L, Hu P, Peng C, Shen G, Zhao F (2014) Epsilon: a visible light based positioning system. In: Proc USENIX NSDI'14, Seattle, WA, pp 331–343
20. Zhang C, Zhang X (2016) LiTell: Robust indoor localization using unmodified light fixtures. In: Proc ACM MobiCom'16, New York, NY, pp 230–242
21. Xu Q, Zheng R, Hranilovic S (2015) IDyLL: Indoor localization using inertial and light sensors on smartphones. In: Proc ACM UbiComp'15, Osaka, Japan, pp 307–318
22. Zhao Z, Wang J, Zhao X, Peng C, Guo Q, Wu B (2017) NaviLight: Indoor localization and navigation under arbitrary lights. In: Proc IEEE INFOCOM'17, Atlanta, GA, pp 1–9
23. Gers FA, Schmidhuber J, Cummins F (2000) Learning to forget: Continual prediction with lstm. J Neural Comput 12(10):2451–2471
24. Greff K, Srivastava RK, Koutníik J, Steunebrink BR, Schmidhuber J (2017) Lstm: a search space odyssey. IEEE Trans Neural Netw Learn Syst 28(10):2222–2232
25. Graves A, Jaitly N, Mohamed A-r (2013) Hybrid speech recognition with deep bidirectional LSTM. In: Proc IEEE ASRU'13, Olomouc, Czech Republic, pp 273–278
26. Ordonez FJ, Roggen D (2016) Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. MDPI Sensors 16(1):115
27. Do T-H, Yoo M (2016) An in-depth survey of visible light communication based positioning systems. MPDI Sensors 16(5):678
28. Kingma D, Ba J (2014) Adam: a method for stochastic optimization. arXiv:1412.6980
29. He S, Chan S-HG (2016) Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons. IEEE Commun Surveys Tuts 18(1):466–490. First Quarter
30. Win MZ, Shen Y, Dai W (2018) A theoretical foundation of network localization and navigation. Proc IEEE 106(7):1136–1165
31. Sun Y, Peng M, Zhou Y, Huang Y, Mao S (2018) Application of machine learning in wireless networks: Key techniques and open issues. arXiv:1809.08707
32. Rapinski J, Cellmer S (2016) Analysis of range based indoor positioning techniques for personal communication networks. Springer Mobile Networks and Applications 21(3):539–549
33. Wang X, Wang X, Mao S (2018) RF Sensing in the Internet of Things: a general deep learning framework. IEEE Commun Mag 56(9):62–67
34. Wang X, Gao L, Mao S, Pandey S (2015) Deepfi: Deep learning for indoor fingerprinting using channel state information. In: Proc WCNC'15, New Orleans, LA, pp 1666–1671
35. Wang X, Gao L, Mao S (2015) Phasefi: Phase fingerprinting for indoor localization with a deep learning approach. In: Proc GLOBECOM'15, San Diego, CA, pp 1–6
36. Wang X, Gao L, Mao S (2016) CSI Phase fingerprinting for indoor localization with a deep learning approach. IEEE Internet of Things J 3(6):1113–1123
37. Wang X, Gao L, Mao S (2017) BiLoc: Bi-modality deep learning for indoor localization with 5 GHz commodity Wi-Fi. IEEE Access J 5(1):4209–4220
38. Xiao C, Yang D, Chen Z, Tan G (2017) 3-D BLE indoor localization based on denoising autoencoder. IEEE Access J 5:12751–12760
39. Gu F, Khoshelham K, Yu C, Shang J (2018) Accurate step length estimation for pedestrian dead reckoning localization using stacked autoencoders. IEEE Trans Instrum Meas, p 1
40. Khatab ZE, Hajihoseini A, Ghorashi SA (2018) A fingerprint method for indoor localization using autoencoder based deep extreme learning machine. IEEE Sens Lett 2(1):1–4
41. Abbas M, Elhamshary M, Rizk H, Torki M, Youssef M (2019) WiDeep: WiFi-based accurate and robust indoor localization system using deep learning. In: IEEE PerCom'19, Kyoto, Japan
42. Wang X, Wang X, Mao S (2017) CiFi: Deep convolutional neural networks for indoor localization with 5GHz Wi-Fi. In: Proc IEEE ICC 2017, Paris, France, pp 1–6
43. Wang W, Wang X, Mao S Deep convolutional neural networks for indoor localization with CSI images. IEEE Transactions on Network Science and Engineering, in press. https://doi.org/10.1109/TNSE.2018.2871165
44. Wang X, Wang X, Mao S (2017) ResLoc: Deep residual sharing learning for indoor localization with CSI tensors. In: Proc IEEE PIMRC 2017, Montreal, Canada, pp 1–6
45. Wang X, Yu Z, Mao S (2018) DeepML: Deep LSTM for indoor localization with smartphone magnetic and light sensors. In: Proc IEEE ICC 2017, Kansas City, MO, pp 1–6
46. Sahar A, Han D (2018) An LSTM-based indoor positioning method using Wi-Fi signals. In: Proc ACM international conference on vision, image and signal processing, Las Vegas, NV, p Article No 43
47. Bytelight Technology. [Online] Available: http://www.bytelight.com/
48. Hu Y, Xiong Y, Huang W, Li X-Y, Yang P, Zhang Y, Mao X (2018) Lightitude: Indoor positioning using uneven light intensity distribution. Proc ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 2(2):Artical 67



Xuyu Wang received the M.S. in Signal and Information Processing in 2012 and B.S. in Electronic Information Engineering in 2009, both from Xidian University, Xi'an, China. He received a Ph.D. in Electrical and Computer Engineering from Auburn University, Auburn, AL, USA in Aug. 2018. He is an Assistant Professor in the Department of Computer Science, California State University, Sacramento, CA. His research interests include indoor localization, deep learning, and big data. He received a Woltolsz Fellowship at Auburn University, and is a co-recipient of the Second Prize of Natural Scientific Award of Ministry of Education, China in 2013, the Best Demo Award of IEEE SECON 2017, and the Best Student Paper Award of IEEE PIMRC 2017.

Zhitao Yu received a B.S. degree in electrical engineering from Nanjing University of Posts and Telecommunication, Nanjing, China in 2016, and an MS degree in Electrical and Computer Engineering from Auburn University, Auburn, AL in 2018. He has been pursuing a PhD in ECE at Auburn University since Spring 2019. His research interests include indoor localization, deep learning and UAV indoor navigation.

Shiwen Mao received his Ph.D. in electrical and computer engineering from Polytechnic University, Brooklyn, NY in 2004 (now New York University Tandon School of Engineering). He was the McWane Associate Professor from 2012 to 2015, and is the Samuel Ginn Distinguished Professor and Director of the Wireless Engineering Research and Education Center (WEREC) at Auburn University, Auburn, AL.

His research interests include wireless networks, multimedia communications, IoT, and smart grid. He was a Distinguished Lecturer from 2014 to 2018, and is a Distinguished Speaker from 2018 to 2021 of the IEEE Vehicular Technology Society. He is on the Editorial Board of IEEE Transactions on Mobile Computing, IEEE Transactions on Multimedia, IEEE Internet of Things Journal, IEEE Networking Letters, IEEE Multimedia, and ACM GetMobile. He received the IEEE ComSoc TC-CSR Distinguished Technical Achievement Award in 2019, several service awards from the IEEE ComSoc, and the NSF CAREER Award in 2010. He is a co-recipient of the IEEE ComSoc MMTC Best Journal Paper Award in 2019, the IEEE ComSoc MMTC Best Conference Paper Award in 2018, the Best Paper Awards from IEEE GLOBECOM 2016, IEEE GLOBECOM 2015, IEEE WCNC 2015, and IEEE ICC 2013, the Best Demo Award from IEEE SECON 2017, and the 2004 IEEE Communications Society Leonard G. Abraham Prize in the Field of Communications Systems. He is a Fellow of the IEEE.