

RESEARCH

Open Access



A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting

Kyeong Soo Kim^{1,2*} , Sanghyuk Lee^{1,2} and Kaizhu Huang¹

An earlier version of this paper was presented in part at ICUMT/FOAN 2017, Munich, Germany, November 2017.

*Correspondence:

Kyeongsoo.Kim@xjtlu.edu.cn

¹ Department of Electrical and Electronic Engineering, XJTLU, 111 Ren'ai Road, 215123 Suzhou, China

² Centre for Smart Grid and Information Convergence, XJTLU, 111 Ren'ai Road, 215123 Suzhou, China

Abstract

Background: One of the key technologies for future large-scale location-aware services covering a complex of multi-story buildings is a *scalable indoor localization technique*. In this paper, we report the current status of our investigation on the use of deep neural networks (DNNs) for the scalable building/floor classification and floor-level position estimation based on Wi-Fi fingerprinting. Exploiting the hierarchical nature of the building/floor estimation and floor-level coordinates estimation of a location, we propose a new DNN architecture consisting of a stacked autoencoder for the reduction of feature space dimension and a feed-forward classifier for *multi-label classification* of building/floor/location, on which the multi-building and multi-floor indoor localization system based on Wi-Fi fingerprinting is built.

Results: We evaluate the performance of building/floor estimation and floor-level coordinates estimation of a given location using the UJIIndoorLoc dataset covering three buildings with four or five floors in the Jaume I University (UJI) campus, Spain. Experimental results demonstrate the feasibility of the proposed DNN-based indoor localization system, which can provide near state-of-the-art performance using a single DNN.

Conclusions: The proposed scalable DNN architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting can achieve near state-of-the-art performance with just a single DNN and enables the implementation with lower complexity and energy consumption at mobile devices.

Keywords: Multi-building and multi-floor indoor localization, Wi-Fi fingerprinting, Deep learning, Neural networks, Multi-label classification

Background

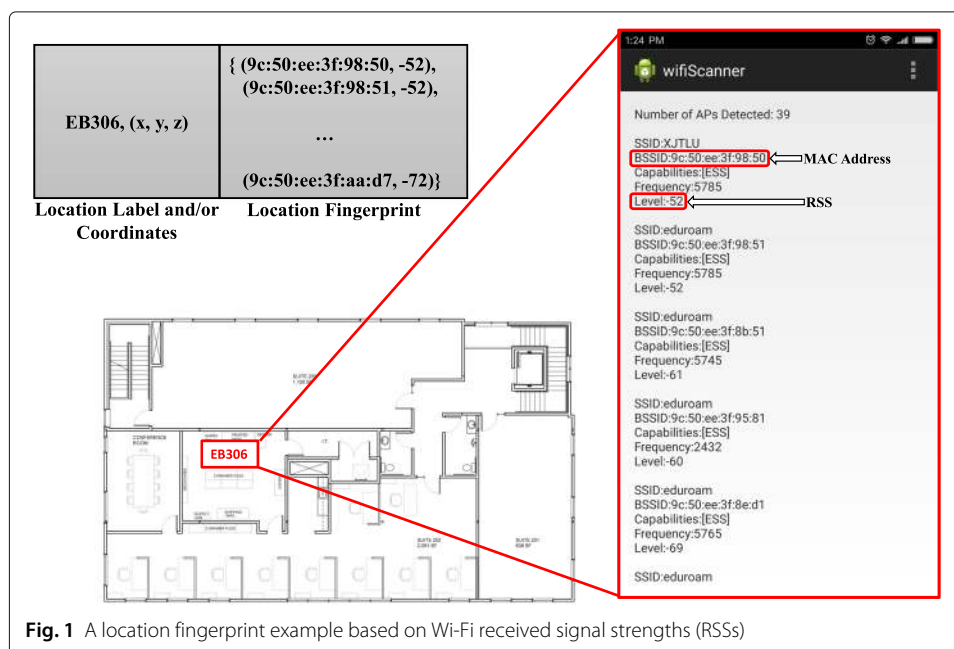
Location fingerprinting using received signal strengths (RSSs) from wireless network infrastructure is one of the most popular and promising technologies for localization in an indoor environment, where there is no line-of-sight signal from the global positioning system (GPS) available [1]: For example, a vector of pairs of a medium access control (MAC) address and an RSS for a Wi-Fi access point (AP) measured at a location can

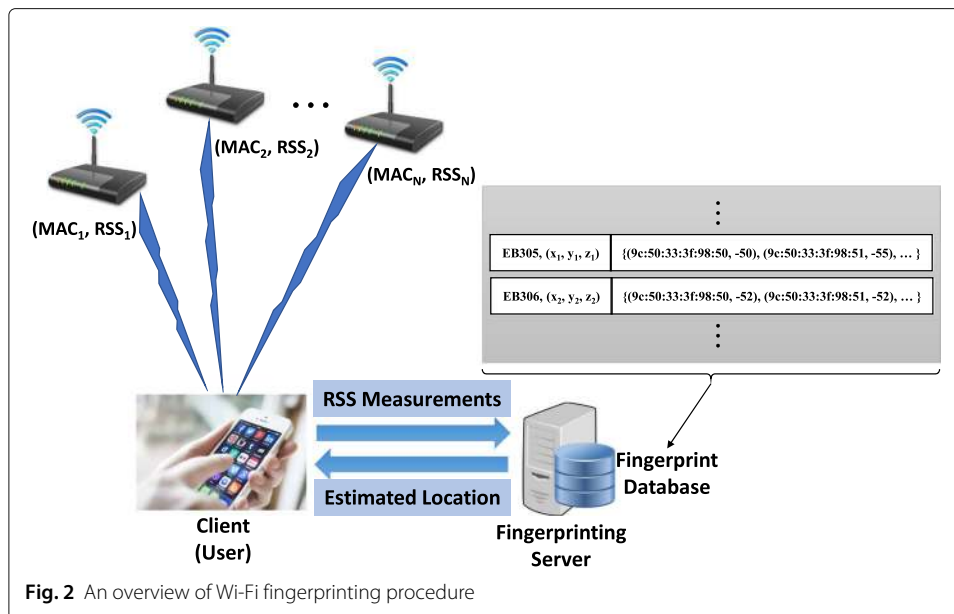
be its *location fingerprint*, which is illustrated in Fig. 1. A location of a user/device then can be estimated by finding the closest match between its RSS measurement and the fingerprints of known locations in a database [2] as shown in Fig. 2. Note that the location fingerprinting technique does not require the installation of any new infrastructure or the modification of existing devices but that it is just based on the existing wireless infrastructure, which is its major advantage over alternative techniques like triangulation based on time of arrival (TOA) requiring precise synchronization among all transmitters and receivers in the system and non-standard timestamp labeling for the measurement of distances between a target and reference points [3].

In this paper, we report the current status of our investigation on the use of deep neural networks (DNNs) in multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting. We propose a new DNN architecture consisting of a stacked autoencoder (SAE) [4] for the reduction of feature space dimension and a feed-forward *multi-label classifier* [5, 6] for the scalable building/floor classification and floor-level location estimation and evaluate its performance using the UJIIndoorLoc dataset [7].

Multi-building and multi-flow indoor localization

When the indoor localization is to cover a large *building complex* — e.g., a big shopping mall or a university campus — where there are lots of multi-story buildings under the same management, the scalability of fingerprinting techniques becomes an important issue. The current state-of-the-art Wi-Fi fingerprinting techniques assume a hierarchical approach to the indoor localization, where the building, floor, and position (e.g., a label or coordinates) of a location are estimated in a hierarchical and sequential way using a different algorithm tailored for each task. In [8], for instance, building estimation is done as follows: Given the AP with the strongest RSS in a measured fingerprint, we first build a subset of fingerprints where the same AP has the strongest RSS; then, we count the number of fingerprints associated to each building and set the estimated building to be the





most frequent one from the counting. Similar procedures are also proposed to estimate a floor inside the building. For the estimation of the coordinates of the location, we first build a subset of fingerprints belonging to the building and the floor estimated from the previous procedures. Then, take multiple fingerprints from the subset most similar to the measured one, and compute the centroid of the coordinates of the selected fingerprints as the estimated coordinates of the given location. According to the results in [8], the best building and floor hit rates achieved for the UJIIndoorLoc dataset [7] are 100% and 94%, respectively, and the mean error in coordinates estimation is 6.20m^1 .

Wi-Fi fingerprinting based on deep neural networks

One of the major challenges in Wi-Fi fingerprinting is how to deal with the random fluctuation of a signal, the noise from multi-path effects, and the device & position dependency in RSS measurements. Unlike traditional solutions relying on complex filtering and time-consuming parameter tuning specific to given conditions, the popular DNNs can provide attractive solutions to Wi-Fi fingerprinting due to their less parameter tuning and adaptability to a wider range of conditions with standard architectures and training algorithms [9–11]: In [9], a four-layer DNN generates a coarse positioning estimate, which, in turn, is refined to produce a final position estimate by a hidden Markov model (HMM)-based fine localizer. The performance of the proposed indoor localization system is evaluated in both indoor and outdoor environments which are divided into hundreds of square grids. In [10], the authors investigate the application of deep belief networks (DBNs) with two different types of Restricted Boltzmann Machines for indoor localization and evaluate the performance of their approaches using data from simulation in heterogeneous mobile radio networks using ray tracing techniques. In both cases, the authors focus only on the localization in a single plane and do not consider the hierarchical nature of multi-building and multi-floor indoor localization. In [11], on the other hand, a DNN consisting of an SAE and a feed-forward multi-class classifier is used for building/floor classification. This

work, too, does not take into account the hierarchical nature of building/floor classification, because the classification is done over flattened, one-dimensional labels of combined building and floor identifiers. Also, the floor-level location estimation is not considered at all. In this regard, to the best of our knowledge, the work presented in this paper is the first to apply DNNs for multi-building and multi-floor indoor localization, exploiting its hierarchical nature in classification.

A scalable DNN architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting

Location awareness is one of enabling technologies for future smart and green cities; understanding where people spend their times and how they interact with environments is critical to realizing the vision of smart and green cities [12]. One of the key technologies for future large-scale location-aware services covering a complex of multi-story buildings — e.g., a big shopping mall and a university campus — is a scalable indoor localization technique. Regarding the scalability of the indoor localization, consider the evolution of the Xi'an Jiaotong-Liverpool University (XJTLU) campus in Suzhou, China, where the authors are currently working: As shown in Fig. 3a, the XJTLU started with just one building in 2006. As of this writing, the XJTLU has two campuses, which are shown in Fig. 3b, and the number of buildings over two campuses has increased to around 20; this number

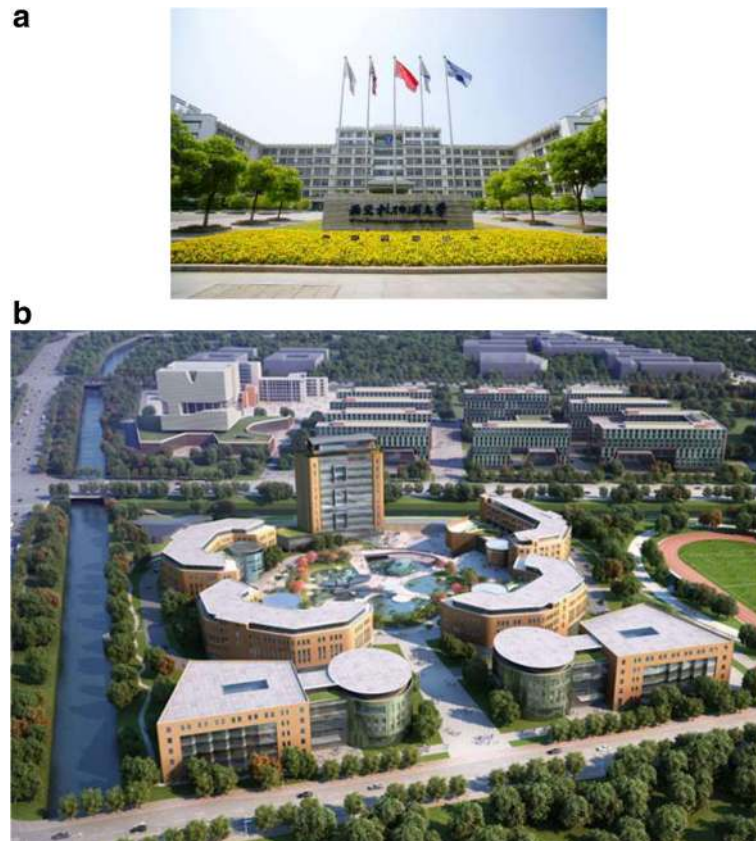
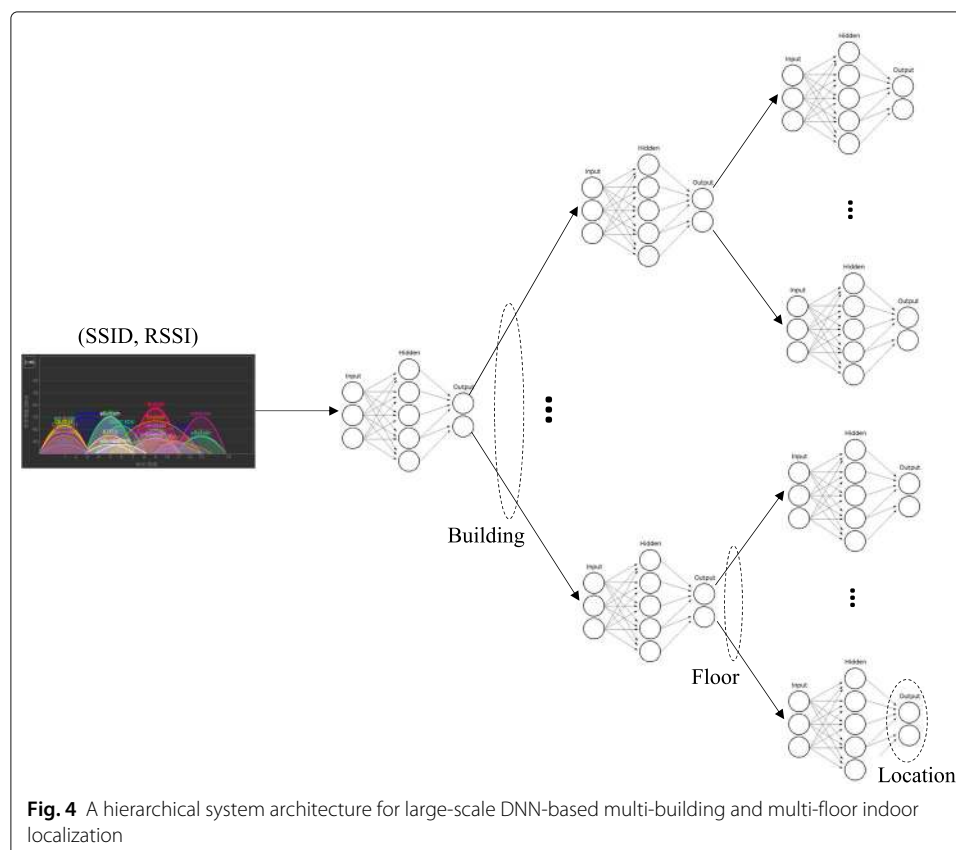


Fig. 3 XJTLU campus in (a) 2006 and (b) 2017

is still increasing as more buildings and sports facilities are being constructed. Considering all the floors within each building and the locations on each floor, the total number of distinct locations (e.g., offices, lecture rooms, and labs) is already on the order of thousands. If we adopt a grid-based representation of the localization area as in [9], the total number of locations would be even greater. The indoor localization system to cover such a large building complex, therefore, must be scalable.

Figures 4 and 5 show two alternative system architectures for large-scale DNN-based multi-building and multi-floor indoor localization. In the hierarchical architecture shown in Fig. 4, the task of building/floor/location classification is separated into multiple sub-tasks dedicated to the classification at each level of building, floor, and location. This architecture directly corresponds to the state-of-the-art hierarchical Wi-Fi fingerprinting methods (e.g., [8]), where DNNs replace traditional techniques for building, floor, and location estimation. Compared to the methods based on traditional techniques, a major disadvantage in this hierarchical DNN architecture is that the DNNs in the floor and the location levels of the system need to be trained separately with multiple sub-datasets derived from a common dataset (i.e., building-specific datasets for DNNs for floor estimation and building-floor-specific datasets for DNNs for location estimation), which poses significant challenges on the management of location fingerprint databases as well as the training of possibly a large number of DNNs. In this paper, therefore, we focus on the integrated architecture shown in Fig. 5 where a single DNN handles the classification of building, floor, and location in an integrated way with a common dataset.



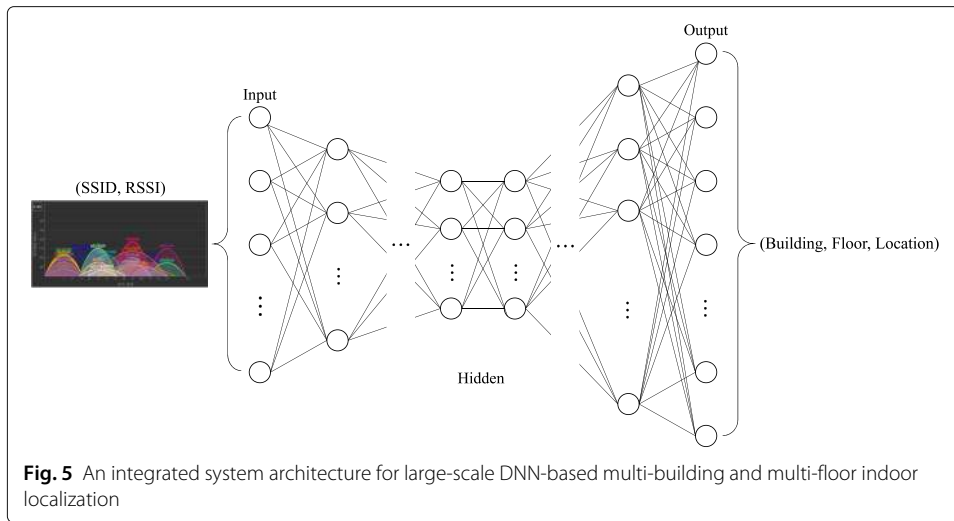
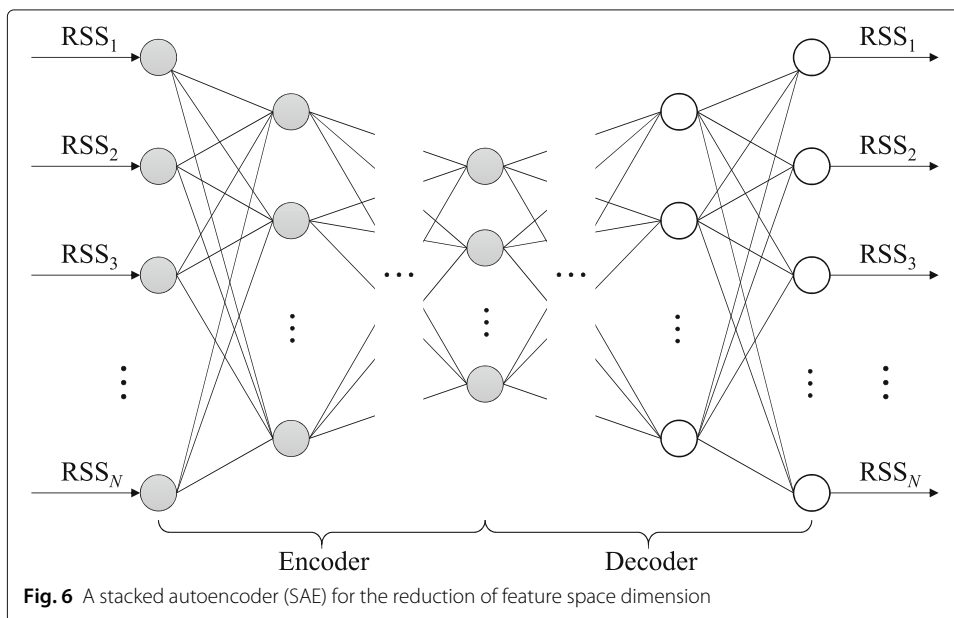
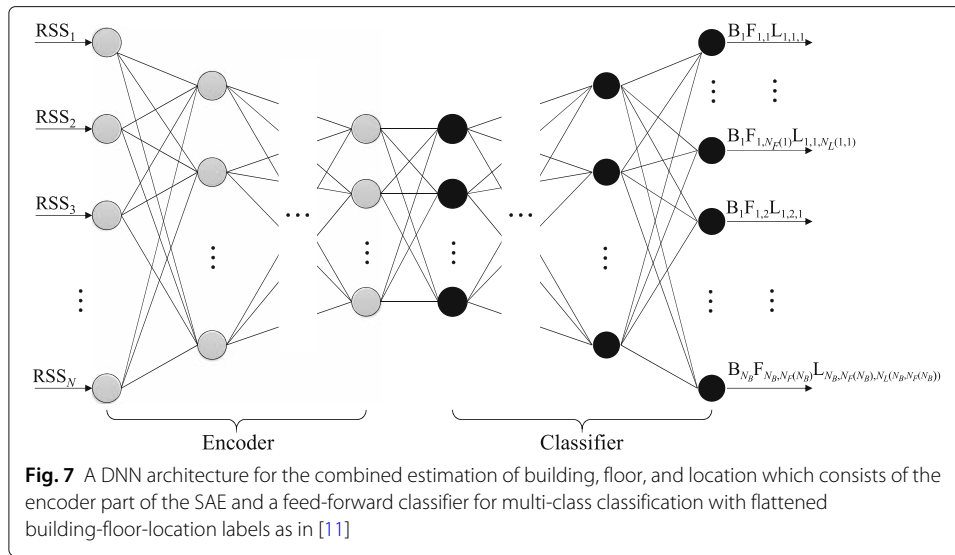


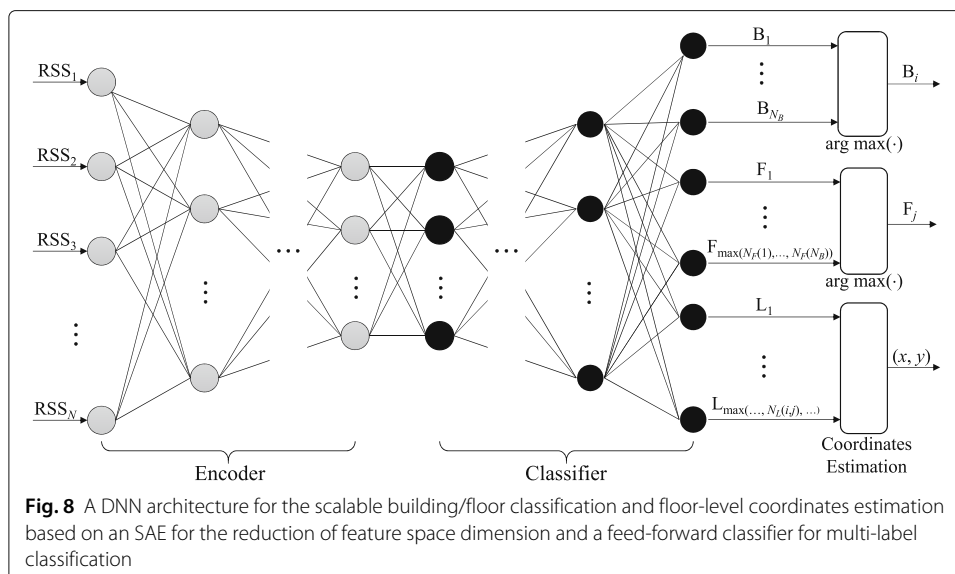
Figure 6 shows an SAE used for the reduction of feature space dimension; after training with RSSs as both input and output data as shown in the figure, only the gray-colored nodes are used as an encoder for feature space dimension reduction. Figure 7 shows a DNN architecture for the combined estimation of building, floor, and location consisting of the encoder part of the SAE and a feed-forward classifier for *multi-class classification* with flattened building-floor-location labels, which is a straightforward extension of the DNN system proposed in [11] for building/floor classification. This DNN architecture based on multi-class classification with flattened labels, however, has the scalability issue that the number of output nodes is equal to the number of locations over the building complex: In case of the UJIIndoorLoc dataset, the number of distinct locations (i.e., also called *reference points* in [7]) over three buildings with four or five floors is 933. It also does not take into account the hierarchical nature of the building/floor/location





classification problem due to its calculating the loss and the accuracy over *flattened building/floor/location labels*²; the misclassification of building, floor, or location has equal loss during the training phase. To reflect the hierarchical nature of the building/floor/location classification in a DNN classifier, one can use a hierarchical loss function — e.g., a loss function with different weights for building, floor, and location — with the existing multi-class classifier and flattened labels. Because the hierarchical loss function for flattened labels is quite complicated and does not provide a closed-form gradient function, however, training the DNN with the usual backpropagation procedure could be challenging.

To address the scalability issue of the DNN classifier based on multi-class classification and take into account the hierarchical nature of the building/floor/location classification, we propose a scalable DNN architecture based on *multi-label classification*³ shown in Fig. 8. The building/floor/location classification with the proposed architecture is done as



follows: First, building, floor, and location identifiers are mapped to sequential numbers, the latter two of which are meaningful only in combination with higher-level numbers; those numbers are one-hot encoded *independently* and combined together into a vector as a categorical variable for multi-label classification as illustrated in Table 1. Then, the output vector from the multi-label classifier is split into a building, a floor, and a location vector by indexes as shown in Fig. 8. Finally, we estimate the building and the floor of a location as the index of a maximum value of the corresponding vector through the *arg max* function. For the estimation of the location coordinates, we select κ largest elements from the location vector (i.e., $\mathbf{L} = (L_1, \dots, L_{\max(\dots, N_L(i,j))})$ in Fig. 8), filter out the elements whose values are less than $\sigma \times \max(\mathbf{L})$ ($\sigma \in [0, 1]$), and calculate the estimated coordinates of the location as either a normal or a weighted (with the values of the elements as weights) centroid of the remaining elements as described in detail in Fig. 9.

Note that there are two design parameters — i.e., κ and σ — in the location coordinates estimation procedure, the rationale of which is illustrated in Fig. 10: If we use only κ as a design parameter as in [8] and sets its value to 5 in Fig. 10a, we can include the reference points quite close to the new location (i.e., those inside the dotted circle) in the estimation procedure and can generate good estimation. In Fig. 10b, however, the same value of κ could result in poor estimation because the reference points 4 and 5 have to be considered during the estimation. With $\kappa = 3$, on the other hand, we can expect good estimation with Fig. 10b but not with Fig. 10a this time. If we can use both κ and σ as design parameters, however, we can include good reference points by properly setting σ for a threshold value. The actual effects of these design parameters on the location coordinates estimation are investigated in “Results” and “Discussion” sections.

Table 1 Label formation example for the multi-label classification of building, floor, and location with two multi-story buildings

| Building | Floor | Location | Sequential coding | One-hot coding | |
|----------|-------|-------------------|-------------------|----------------|--------------|
| A | 1st | Lecture Theater A | 0,0,0 | 01 0001 0001 | |
| | | Lecture Theater B | 0,0,1 | 01 0001 0010 | |
| | 2nd | Lab 1 | 0,1,0 | 01 0010 0001 | |
| | | Lab 2 | 0,1,1 | 01 0010 0010 | |
| | 3rd | A301 | 0,2,0 | 01 0100 0001 | |
| | | A302 | 0,2,1 | 01 0100 0010 | |
| | | A303 | 0,2,2 | 01 0100 0100 | |
| | B | 1st | Common Room | 1,0,0 | 10 0001 0001 |
| | | | Printing Room | 1,0,1 | 10 0001 0010 |
| 2nd | | Conference Room 1 | 1,1,0 | 10 0010 0001 | |
| | | Conference Room 2 | 1,1,1 | 10 0010 0010 | |
| 3rd | | B301 | 1,2,0 | 10 0100 0001 | |
| | | B302 | 1,2,1 | 10 0100 0010 | |
| | | B303 | 1,2,2 | 10 0100 0100 | |
| | | B304 | 1,2,3 | 10 0100 1000 | |
| 4th | | Gym | 1,3,0 | 10 1000 0001 | |


```

Input : Training/validation dataset  $\mathcal{D}$ 
Input : Estimated building ID  $B_i$ 
Input : Estimated floor ID  $F_j$ 
Input :  $\mathcal{L} = \{\mathcal{L}_1 \dots \mathcal{L}_\kappa\}$ , a set of  $\kappa$  largest elements of the location vector  $L$  from the DNN classifier
Output: Estimated location coordinates  $C_s = (x_s, y_s)$  (centroid)
Output: Estimated location coordinates  $C_w = (x_w, y_w)$  (weighted centroid)

 $sum_x \leftarrow 0$ 
 $wsum_x \leftarrow 0$ 
 $sum_y \leftarrow 0$ 
 $wsum_y \leftarrow 0$ 
 $sum_w \leftarrow 0$ 
 $n_c \leftarrow 0$  // number of candidate coordinates
for  $l \in \mathcal{L}$  do
  if  $l \geq \sigma \times \max(L)$  then // threshold checking
    if  $(B_i, F_j, l) \in \mathcal{D}$  then // i.e., a valid location of the dataset
       $x \leftarrow \text{coords}_x(B_i, F_j, l)$  // return the  $x$  coordinate of building/floor/location
       $y \leftarrow \text{coords}_y(B_i, F_j, l)$  // return the  $y$  coordinate of building/floor/location
       $sum_x \leftarrow sum_x + x$ 
       $wsum_x \leftarrow wsum_x + l \times x$ 
       $sum_y \leftarrow sum_y + y$ 
       $wsum_y \leftarrow wsum_y + l \times y$ 
       $sum_w \leftarrow sum_w + l$ 
       $n_c \leftarrow n_c + 1$ 
    end
  end
end
if  $n_c > 0$  then
   $C_s \leftarrow (\frac{sum_x}{n_c}, \frac{sum_y}{n_c})$ 
   $C_w \leftarrow (\frac{wsum_x}{sum_w}, \frac{wsum_y}{sum_w})$ 
end
else // i.e., no candidate coordinates
   $C_s, C_w \leftarrow$  centroid of all location coordinates belonging to building  $B_i$  and floor  $F_j$  of  $\mathcal{D}$ 
end

```

Fig. 9 Procedure for the estimation of location coordinates

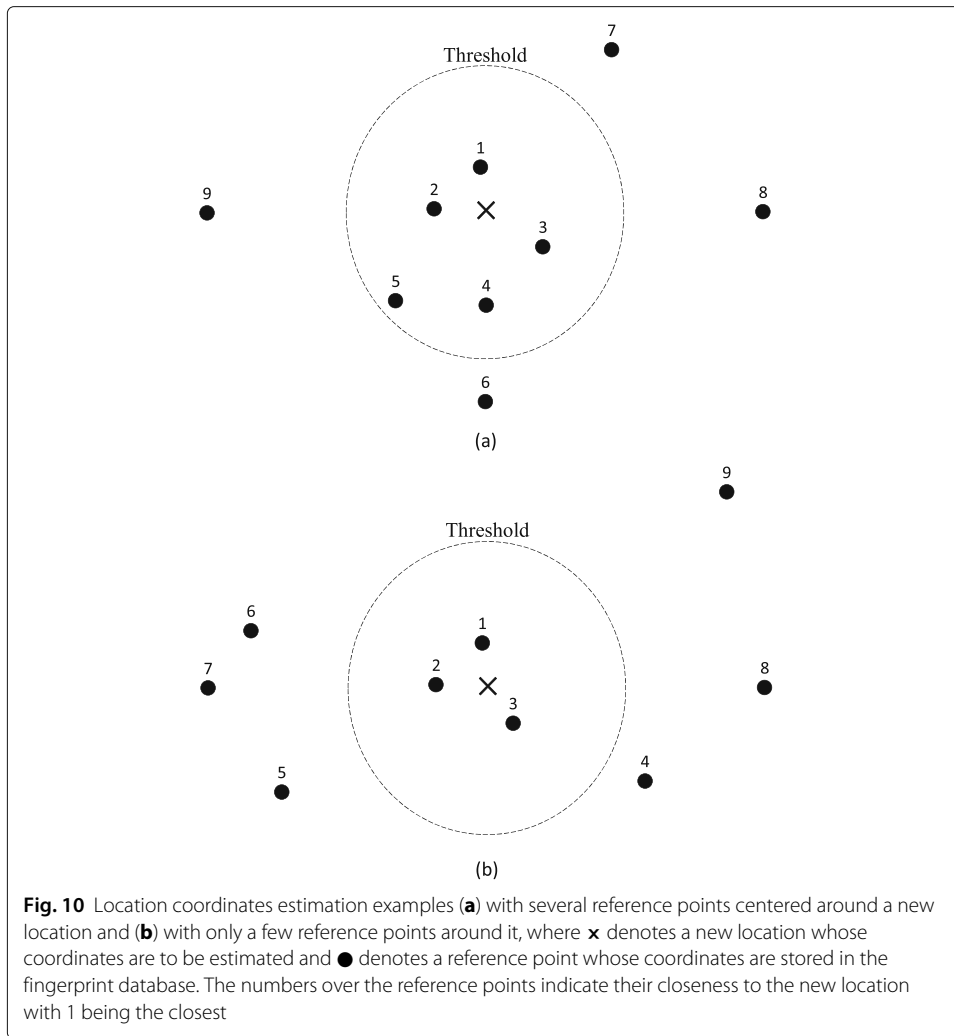
As for the scalability of the proposed DNN architecture, the number of output nodes becomes much smaller than that of the DNN architecture based on multi-class classification: The number of output nodes for multi-label building/floor/location classification is given by

$$N_B + \max(N_F(1), \dots, N_F(N_B)) + \max(N_L(1, 1), \dots, N_L(N_B, N_F(N_B))), \quad (1)$$

where N_B , $N_F(i)$, and $N_L(j, k)$ are the number of buildings in the complex, the number of floors in the i th building ($i = 1, \dots, N_B$), and the number of locations on the k th floor of the j th building ($j = 1, \dots, N_B$; $k = 1, \dots, N_F(j)$), respectively. Note that for multi-class building/floor/location classification, the number becomes

$$\sum_{i=1}^{N_B} \sum_{j=1}^{N_F(i)} N_L(i, j). \quad (2)$$

According to (1), the number of output nodes of the proposed DNN architecture for the publicly available UJIIndoorLoc dataset at the University of California, Irvine (UCI), Machine Learning Repository⁴ is given by 118 (i.e., the sum of the number of buildings (3), the maximum of the numbers of floors of the buildings (5), and the maximum of the numbers of locations⁵ on the floors (110)), which is smaller than the number of output nodes of the DNN architecture based on multi-class classification (i.e., 905⁶). Note that the difference could be much larger if the UJIIndoorLoc dataset covers all the buildings on the Jaume I University (UJI) campus where the data were collected.



Also, due to the clear mapping between building, floor, and location identifiers and its corresponding one-hot-encoded categorical variable for the DNN-based multi-label classifier, it is easy to carry out different processing for parts of DNN outputs specifically for building, floor, and location as illustrated in Fig. 8. Especially, the use of multiple elements in estimating location coordinates as described in Fig. 9 is a huge advantage in terms of computational complexity because trained DNNs can generate multi-dimensional output values in parallel; in traditional approaches, on the other hand, selecting nearest locations based on Euclidean distances are complex and time consuming. This flexibility in handling DNN outputs also makes it easy to apply different weights to the cost of building, floor, and location classification error during the training phase.

Methods

We carry out experiments using the UJIIndoorLoc dataset [7] to evaluate the performance of the proposed DNN-based multi-building and multi-floor indoor localization system. DNN models are implemented based on Keras [13] and TensorFlow [14], and their source code is available online [15]. We focus on the effects of the number of largest

elements from the output location vector (i.e., κ) and the scaling factor for a threshold (i.e., σ) in the location coordinates estimation procedure described in “A scalable DNN architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting” section. Table 2 summarizes DNN parameter values for the experiments, which are chosen experimentally and used throughout the experiments.

As indicated in [11], the publicly available UJIIndoorLoc dataset includes training and validation data, but not testing data which were provided only to the competitors at the Evaluating Ambient Assisted Living (EvAAL) competition at the International Conference on Indoor Positioning and Indoor Navigation (IPIN) 2015 [8]. Also, unlike the training data, the validation data do not include location information (i.e., *SpaceID* and *RELATIVEPOSITION* fields) because the measurements were taken at arbitrary points as would happen in a real localization system. In this regard, we split the training data into new training and validation data with the ratio of 70:30 for DNN training and validation with building/floor/location labels for both. During the evaluation phase, the output from the trained DNN are post-processed as described in “A scalable DNN architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting” section and compared with the building, floor, and coordinates of a given location. In this way, we can compare our results of multi-building and multi-floor indoor localization with the baseline and the best results from [7] and [8], respectively.

Results

Table 3 summarizes our experimental results, which show the effects of the number of largest elements from the output location vector (κ) and the scaling factor for a threshold (σ) on the performance of multi-building and multi-floor indoor localization.

Discussion

In general, σ in the range of 0.1–0.3 produces the best localization performance for $\kappa \leq 8$; once κ becomes larger than 8, however, higher values of σ (i.e., 0.4 for $\kappa = 9$ and 0.5 for $\kappa = 10$) generate better performance. Considering the coordinates location estimation examples shown in Fig. 10 with their explanations in “A scalable DNN architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting” section regarding the use of two design parameters, we can explain these results as follows:

Table 2 Parameter Values for Scalable DNN-Based Indoor Localization

| DNN parameter | Value |
|--|--------------------------|
| Ratio of training data to overall data | 0.90 |
| Number of Epochs | 20 |
| Batch size | 10 |
| SAE hidden layers | 256-128-256 |
| SAE activation | Rectified Linear (ReLU) |
| SAE Optimizer | ADAM [25] |
| SAE loss | Mean Squared Error (MSE) |
| Classifier hidden layers | 64-128 |
| Classifier activation | ReLU |
| Classifier optimizer | ADAM |
| Classifier loss | Binary Crossentropy |
| Classifier dropout rate | 0.20 |

Table 3 Effects of the number of largest elements from the output location vector (κ) and the scaling factor for a threshold (σ) on the Performance of multi-building and multi-floor indoor localization

| κ | σ | Building hit rate [%] | Floor hit rate [%] | Success rate [%] | Positioning error [m] | |
|----------|----------|-----------------------|--------------------|------------------|-----------------------|-------------------|
| | | | | | Centroid | Weighted centroid |
| 1 | N/A* | 99.82 | 91.90 | 91.81 | 11.40 | 11.40 |
| 2 | 0.0 | 99.37 | 92.44 | 91.81 | 10.62 | 10.54 |
| | 0.1 | 100.00 | 91.81 | 91.81 | 10.40 | 10.33 |
| | 0.2 | 99.82 | 92.62 | 92.44 | 9.74 | 9.66 |
| | 0.3 | 99.64 | 91.99 | 91.81 | 9.78 | 9.71 |
| | 0.4 | 99.73 | 91.54 | 91.45 | 10.29 | 10.21 |
| | 0.5 | 100.00 | 90.01 | 90.01 | 10.16 | 10.09 |
| 3 | 0.0 | 99.73 | 91.54 | 91.36 | 10.14 | 9.79 |
| | 0.1 | 99.91 | 90.91 | 90.82 | 9.92 | 9.76 |
| | 0.2 | 98.83 | 90.91 | 90.28 | 9.98 | 9.80 |
| | 0.3 | 99.55 | 92.08 | 91.90 | 10.13 | 10.01 |
| | 0.4 | 99.91 | 91.99 | 91.99 | 10.63 | 10.47 |
| | 0.5 | 99.82 | 90.37 | 90.37 | 9.94 | 9.89 |
| 4 | 0.0 | 99.82 | 90.91 | 90.91 | 10.27 | 9.66 |
| | 0.1 | 99.37 | 91.99 | 91.63 | 10.37 | 9.92 |
| | 0.2 | 99.64 | 92.08 | 91.90 | 10.26 | 10.09 |
| | 0.3 | 99.82 | 91.45 | 91.36 | 10.24 | 10.16 |
| | 0.4 | 99.91 | 92.26 | 92.17 | 10.35 | 10.23 |
| | 0.5 | 99.82 | 91.27 | 91.18 | 10.10 | 10.07 |
| 5 | 0.0 | 99.91 | 91.36 | 91.27 | 11.29 | 10.36 |
| | 0.1 | 99.91 | 91.63 | 91.63 | 9.90 | 9.62 |
| | 0.2 | 99.91 | 90.73 | 90.73 | 9.89 | 9.57 |
| | 0.3 | 99.82 | 90.91 | 90.82 | 10.27 | 9.99 |
| | 0.4 | 99.73 | 92.17 | 92.08 | 10.17 | 10.01 |
| | 0.5 | 99.82 | 92.98 | 92.89 | 10.59 | 10.54 |
| 6 | 0.0 | 99.82 | 91.90 | 91.72 | 10.84 | 9.71 |
| | 0.1 | 99.64 | 92.08 | 91.81 | 10.35 | 9.86 |
| | 0.2 | 100.00 | 91.99 | 91.99 | 9.85 | 9.56 |
| | 0.3 | 99.82 | 92.80 | 92.80 | 10.49 | 10.22 |
| | 0.4 | 99.37 | 91.09 | 91.00 | 10.32 | 10.17 |
| | 0.5 | 99.64 | 90.91 | 90.64 | 9.55 | 9.52 |
| 7 | 0.0 | 99.82 | 89.29 | 89.29 | 11.74 | 10.22 |
| | 0.1 | 99.82 | 90.19 | 90.01 | 10.43 | 9.82 |
| | 0.2 | 99.91 | 91.45 | 91.45 | 10.00 | 9.55 |
| | 0.3 | 99.91 | 91.63 | 91.54 | 9.75 | 9.53 |
| | 0.4 | 99.64 | 90.46 | 90.19 | 10.42 | 10.28 |
| | 0.5 | 99.55 | 91.45 | 91.36 | 9.83 | 9.73 |
| 8 | 0.0 | 99.91 | 90.19 | 90.10 | 11.32 | 9.27 |
| | 0.1 | 100.00 | 91.27 | 91.27 | 10.62 | 10.14 |
| | 0.2 | 99.82 | 91.27 | 91.18 | 9.76 | 9.29 |
| | 0.3 | 99.82 | 90.55 | 90.37 | 9.95 | 9.82 |
| | 0.4 | 99.91 | 90.37 | 90.28 | 10.21 | 10.14 |
| | 0.5 | 99.91 | 90.55 | 90.55 | 9.86 | 9.79 |

Table 3 Effects of the number of largest elements from the output location vector (κ) and the scaling factor for a threshold (σ) on the Performance of multi-building and multi-floor indoor localization (*Continued*)

| κ | σ | Building hit rate [%] | Floor hit rate [%] | Success rate [%] | Positioning error [m] | |
|----------|----------|-----------------------|--------------------|------------------|-----------------------|-------------------|
| | | | | | Centroid | Weighted centroid |
| 9 | 0.0 | 99.82 | 91.00 | 90.91 | 12.75 | 9.76 |
| | 0.1 | 99.82 | 90.28 | 90.19 | 10.17 | 9.55 |
| | 0.2 | 99.82 | 90.73 | 90.64 | 10.05 | 9.76 |
| | 0.3 | 99.82 | 91.45 | 91.36 | 10.36 | 10.23 |
| | 0.4 | 100.00 | 91.81 | 91.81 | 9.86 | 9.78 |
| | 0.5 | 99.91 | 92.80 | 92.71 | 10.26 | 10.23 |
| 10 | 0.0 | 99.91 | 90.91 | 90.82 | 13.21 | 9.75 |
| | 0.1 | 99.82 | 92.71 | 92.62 | 10.37 | 9.85 |
| | 0.2 | 99.91 | 91.72 | 91.72 | 10.33 | 10.04 |
| | 0.3 | 100.00 | 90.82 | 90.82 | 10.47 | 10.24 |
| | 0.4 | 100.00 | 91.09 | 91.09 | 10.10 | 9.96 |
| | 0.5 | 100.00 | 91.90 | 91.90 | 9.96 | 9.91 |

* N/A = not applicable; when $\kappa = 1$, the value of σ does not affect the selection of locations (i.e., reference points) included in the coordinates estimation

With a larger value of κ (i.e., 9 and 10), there could be a higher chance of including reference points relatively far from the given location as shown in Fig. 10b. In such a case, a tighter threshold (i.e., a larger value of σ^7) can filter out those reference points.

According to the results shown in Table 3, collectively the best results are achieved when $\kappa = 8$ and $\sigma = 0.2$. These results from the proposed DNN-based multi-building and multi-floor indoor localization system — i.e., 99.82% for building hit rate, 91.27% for floor hit rate, 91.18% for success rate and 9.29m for positioning error — are favorably comparable to the baseline results — i.e., 89.92% for success rate and 7.9m for positioning error — from [7] which are based on the distance-based k -Nearest Neighbors (k NN) algorithm [16]. As discussed in “A scalable DNN architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting” section, even though a direct comparison with the results from the EvAAL/IPIN 2015 competition is not possible due to the lack of testing samples in the public version of the UJIIndoorLoc dataset and a slightly different way of calculating the positioning error, our results are also comparable to the competition results summarized in Table 4.

Note that the results presented in this section are not optimized with DNN parameters, including the number of hidden layers and the number of nodes at each layer; we investigated the feasibility of the combined building/floor/location estimation using a single

Table 4 Best results from the four teams at the EvAAL/IPIN 2015 competition [8]

| | MOSAIC | HFTS | RTL SUM | ICSL |
|--------------------------------|--------|-------|---------|-------|
| Building hit rate [%] | 98.65 | 100 | 100 | 100 |
| Floor hit rate [%] | 93.86 | 96.25 | 93.74 | 86.93 |
| Positioning error (Mean) [m] | 11.64 | 8.49 | 6.20 | 7.67 |
| Positioning error (Median) [m] | 6.7 | 7.0 | 4.6 | 5.9 |

DNN based on multi-label classification framework with focus on the effects of the number of largest elements from the output location vector (κ) and the scaling factor for a threshold (σ) in the location coordinates estimation. This leaves much room for further optimization of the performance.

Open issues and areas for further research

There are several open issues raised by the current work and related areas for further research in multi-building and multi-floor indoor localization based on DNNs and Wi-Fi fingerprinting.

First, in the current work we focus on the estimation of the position of a *static* user/device based on location fingerprints consisting of MAC addresses and RSSs of Wi-Fi access points only, which could be extended to the case of a *moving* user/device. In fact, during the more recent IPIN competitions in 2016 and 2017 [17–19], the focuses have been shifted to more challenging trajectory estimation based on continuously-recorded data from multiple sensors of advanced smartphones — including inertial data, magnetic field, GPS, and pressure in addition to Wi-Fi RSS — and additional information like floorplan maps and map-based reference trajectories. Note that, while the static indoor localization based on a minimal set of information (i.e., Wi-Fi RSS) could be considered as the limit of the current work, it is also its strength in that this type of indoor localization makes it easy to build and maintain the fingerprint database and can be applicable to a variety of mobile devices including smartphones, smartwatches, tablets, laptops, wireless sensor nodes, and even Internet of Things (IoTs) devices; the location fingerprints based on Wi-Fi RSS are a common denominator of the information available from any Wi-Fi-capable devices.

Second, there is no direct match between the cost function used in DNN training/validation and the actual performance in the final evaluation for building/floor detection and the location coordinates estimation due to the additional processing of DNN output (i.e., the use of *arg max* function for the former and the rather complicated processing with multiple elements of the location vector for the latter). To fully take into account the actual performance of building/floor detection and location coordinates estimation during training/validation, we may consider heuristics like evolutionary algorithms (e.g., genetic algorithm (GA) [20] and particle swarm optimization (PSO) [21]), simulated annealing [22], and quantum annealing [23] for training DNN weights; due to its many tradeoffs between complexity and flexibility resulting from the use of heuristics in DNN weight training, this approach could be an interesting topic for long-term research.

Third, the issue of building and maintenance of a large-scale fingerprint database should be further investigated, too. For instance, it is a real challenge to gather the location fingerprints covering the vast area of a large building complex either directly from researchers and volunteers or through outsourcing to designated professionals or an organization. In this regard, *crowdsourcing* to a general public (e.g., students and staff members of a university or shoppers and visitors of a shopping mall) could be a viable solution [24]. Note that, however, crowdsourcing could raise other issues like malicious users intentionally submitting wrong data (e.g., RSSs with wrong location labels) in order to corrupt the fingerprint database. Also, how to update the fingerprint database with new measurement data, which could have different

statistics, is another important issue. In the current work based on the UJIIndoorLoc dataset, we already found that the statistics of its training and validation data, which were generated at different times and by different Android apps, show significant differences.

Conclusions

In this paper we have proposed a new scalable DNN architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting, which can cover a large-scale complex of many multi-story buildings under the same management. The proposed DNN architecture consists of an SAE for the reduction of feature space dimension and a feed-forward classifier for multi-label classification of building/floor/location. Reformulating the problem of building/floor/location classification based on the framework of multi-label classification, we can achieve better scalability (i.e., greatly reducing the number of DNN output nodes) and better exploit the hierarchical nature of the building/floor estimation and the floor-level location coordinates estimation through systematic label formation (i.e., providing straightforward mapping between a DNN categorical variable and building, floor & location vectors) compared to existing DNN architectures based on multi-class classification.

The experimental results using the UJIIndoorLoc dataset clearly demonstrate the feasibility of the proposed DNN-based multi-building and multi-floor indoor localization system, which can provide near state-of-the-art performance using a single DNN in an integrated way. Combined with the unique advantage of a DNN-based indoor localization system that, once trained, it does not need the fingerprint database any longer but carries the necessary information for localization in DNN weights, the scalable DNN architecture proposed in this paper could open a door for a future secure and energy-efficient indoor localization solution exclusively running on mobile devices without exchanging any data with the server.

Endnotes

¹ The mean error takes into account the building and floor estimation penalties; refer to [8, Eq. (2)] for details.

² For example, we can form a flattened label “ $B_i-F_{i,j}-L_{i,j,k}$ ” by combining a building, a floor, and a location label, where B_i , $F_{i,j}$, and $L_{i,j,k}$ denote the i th building, j th floor of the building, and k th location on the floor, respectively.

³ In multi-class classification (also called *single-label classification*), an instance is associated with only a single label from a set of disjoint labels; in multi-label classification, on the other hand, an instance can be associated with multiple labels [5].

⁴ <https://archive.ics.uci.edu/ml/datasets/ujiindoorloc>.

⁵ In the UJIIndoorLoc dataset, the position of a location is uniquely determined by four identifiers, i.e., *BuildingID*, *Floor*, *SpaceID*, and *RELATIVEPOSITION*. For convenience, we combine the *SpaceID* and the *RELATIVEPOSITION* into one and mention it as *location* throughout the paper so that the three identifiers for building, floor, and location uniquely determine the position of a location.

⁶ There are slight differences between the statistics of the UJIIndoorLoc dataset described in [7] and those of the publicly available dataset at the UCI Machine Learning Repository.

⁷When $\sigma = 0$, there is no filtering (i.e., including all κ reference points); when $\sigma = 1$, only the reference point with the largest value is considered during the location coordinates estimation.

Acknowledgements

Not applicable.

Funding

This work was supported in part by Xi'an Jiaotong-Liverpool University (XJTLU) Research Development Fund (under Grant RDF-14-01-25), Summer Undergraduate Research Fellowships programme (under Grant SURF-201739), Research Institute for Smart and Green Cities Seed Grant Programme 2016-2017 (under Grant RISGC-2017-4), Centre for Smart Grid and Information Convergence, National Natural Science Foundation of China (NSFC) (under Grant 61473236), Natural Science Fund for Colleges and Universities in Jiangsu Province (under Grant 17KJD520010), and Suzhou Science and Technology Program (under Grant SYG201712, SZS201613).

Availability of data and materials

All the data and source code for DNN models are available in the GitHub repository for research on indoor localization based on wireless fingerprinting techniques at XJTLU [15], https://github.com/kyeongsoo/indoor_localization.

Authors' contributions

KSK, SL, and KH initiated a seed project on the subject of scalable indoor localization based on DNNs and Wi-Fi fingerprinting, which this manuscript is based on. KSK proposed DNN architectures, carried out numerical experiments, and drafted the manuscript, which SL and KH checked and clarified. All authors read and approved the final manuscript.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 2 January 2018 Accepted: 3 April 2018

Published online: 19 April 2018

References

1. He S, Chan S-HG. Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Commun Surv Tuts.* 2016;18(1):466–90.
2. Bahl P, Padmanabhan VN. RADAR: An in-building RF-based user location and tracking system. In: *Proc. 2000 IEEE INFOCOM.* vol. 2. 2000. p. 775–84. <https://doi.org/10.1109/INFCOM.2000.832252>.
3. Liu H, Darabi H, Banerjee P, Liu J. Survey of wireless indoor positioning techniques and systems. *IEEE Trans. Syst. Man Cybern. C.* 2007;37(6):1067–80.
4. Vincent P, Laroche H, Lajoie I, Bengio Y, Manzagol P-A. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J Mach Learn Res.* 2010;11:3371–408.
5. Tsoumakas G, Katakis I. Multi-label classification: An overview. *Int J Data Warehousing Mining.* 2007;3(3):1–13. <https://doi.org/10.4018/jdwm.2007070101>.
6. Herrera F, Charte F, Rivera AJ, del Jesus MJ. Multilabel classification. In: *Multilabel Classification: Problem Analysis, Metrics and Techniques*, Chap. 2. Switzerland: Springer; 2016. p. 17–31.
7. Torres-Sospedra J, Montoliu R, Martínez-Uso A, Avariento JP, Arnau TJ, Benedito-Bordonau M, Huerta J. UJIIndoorLoc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems. In: *Proc. International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Piscataway: IEEE; 2014. p. 261–70.
8. Moreira A, Nicolau MJ, Meneses F, Costa A. Wi-Fi fingerprinting in the real world – RTLSUM at the EvAAL competition. In: *Proc. International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Piscataway: IEEE; 2015. p. 1–10.
9. Zhang W, Liu K, Zhang W, Zhang Y, Gu J. Deep neural networks for wireless localization in indoor and outdoor environments. *Neurocomputing.* 2016;194:279–87.
10. Félix G, Siller M, Álvarez EN. A fingerprinting indoor localization algorithm based deep learning. In: *Proc. ICUFN 2016*. Vienna; 2016. p. 1006–11.
11. Nowicki M, Wietrzykowski J. Low-effort place recognition with wifi fingerprints using deep learning. *ArXiv e-prints.* 2017. arXiv:1611.02049v2 [cs.RO].
12. Benevolo C, Dameri RP, D'Auria B. Smart mobility in smart city: Action taxonomy, ICT intensity and public benefits. In: Torre T, Braccini AM, Spinelli R, editors. *Empowering Organizations: Lecture Notes in Information Systems and Organisation*, 1st edn, Lecture notes in information systems and organisation. Chap. 2. Switzerland: Springer; 2016. p. 13–28.

13. Chollet F, et al. Keras. <https://github.com/keras-team/keras/>. Accessed 12 Apr 2018.
14. TensorFlow™. <https://www.tensorflow.org/>. Accessed 29 Dec 2017.
15. GitHub Repository for Research on Indoor Localization Based on Wireless Fingerprinting Techniques at XJTLU. https://github.com/kyeongsoo/indoor_localization. Accessed 29 Dec 2017.
16. Cover TM, Hart PE. Nearest neighbor pattern classification. *IEEE Trans Inf Theory*. 1967;13(1):21–7.
17. Torres-Sospedra J, Jiménez AR, Knauth S, Moreira A, Beer Y, Fetzer T, Ta V-C, Montoliu R, Seco F, Mendoza-Silva GM, Belmonte O, Koukofikis A, Nicolau MJ, Costa A, Meneses F, Ebner F, Deinzer F, Vaufreydaz D, Dao T-K, Castelli E. The smartphone-based offline indoor location competition at IPIN 2016: Analysis and future work. *Sensors*. 2017;17(3). <https://doi.org/10.3390/s17030557>.
18. Potortí F, Park S, Jiménez Ruiz AR, Barsocchi P, Girolami M, Crivello A, Lee SY, Lim JH, Torres-Sospedra J, Seco F, Montoliu R, Mendoza-Silva GM, Pérez Rubio MDC, Losada-Gutiérrez C, Espinosa F, Macias-Guarasa J. Comparing the performance of indoor localization systems through the EvAAL framework. *Sensors*. 2017;17(10). <https://doi.org/10.3390/s17102327>.
19. Torres-Sospedra J, Jiménez AR, Moreira A, Lungenstrass T, Lu W-C, Knauth S, Mendoza-Silva GM, Seco F, Pérez-Navarro A, Nicolau MJ, Costa A, Meneses F, Farina J, Morales JP, Lu W-C, Cheng H-T, Yang S-S, Fang S-H, Chien Y-R, Tsao Y. Off-line evaluation of mobile-centric indoor positioning systems: The experiences from the 2017 IPIN competition. *Sensors*. 2018;18(2). <https://doi.org/10.3390/s18020487>.
20. Goldberg DE. *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading: Addison-Wesley; 1989.
21. Kennedy J, Eberhart R. Particle swarm optimization. In: *Proc. IEEE International Conference on Neural Networks*. IV. Piscataway: IEEE; 1995. p. 1942–8.
22. Kirkpatrick S, Jr. CDG, Vecchi MP. Optimization by simulated annealing. *Science*. 1983;220(4598):671–80.
23. Finnila AB, Gomez MA, Sebenik C, Stenson C, Doll JD. Quantum annealing: A new method for minimizing multidimensional functions. *Chem Phys Lett*. 1994;219(5-6):343–8. [https://doi.org/10.1016/0009-2614\(94\)00117-0](https://doi.org/10.1016/0009-2614(94)00117-0).
24. Ali A, Qadir J, Rasool Ru, Sathiaselan A, Zwitter A, Crowcroft J. Big data for development: applications and techniques. *Big Data Analytics*. 2016;1(1):1–24. <https://doi.org/10.1186/s41044-016-0002-4>.
25. Kingma D, Ba J. ADAM: A method for stochastic optimization. *ArXiv e-prints*. 2017. arXiv:1412.6980v8 [cs.LG].

Submit your next manuscript to BioMed Central
and we will help you at every step:

- We accept pre-submission inquiries
- Our selector tool helps you to find the most relevant journal
- We provide round the clock customer support
- Convenient online submission
- Thorough peer review
- Inclusion in PubMed and all major indexing services
- Maximum visibility for your research

Submit your manuscript at
www.biomedcentral.com/submit

