Towards Sub-room Level Occupancy Detection with Denoising-Contractive Autoencoder

Pai Chet Ng†, James She† and Rong Ran‡
†Department of Electronic and Computer Engineering,
Hong Kong University of Science and Technology; Email: {pcng, eejames}@ust.hk
‡School of Electrical and Computer Engineering, Ajou University; Email: sunnyran@ajou.ac.kr

***Abstract*—Lately, there are many works exploited the radio frequency (RF) fingerprint for occupancy detection. However, most works suffer severe performance variations owing to the unreliable received signal strength (RSS). In this paper, we propose a deep learning approach to occupancy detection: 1) an unsupervised denoising-contractive autoencoder (DCAE) is built to learn a robust fingerprint representation from the raw RSS measurements, and 2) a supervised softmax function is added at the last layer for classification. A real testbed with Bluetooth Low Energy (BLE) beacons was built such that we can collect real-world RSS data for experiments. The data were collected via different devices at different times to better reflect environmental variations. The experimental results show that our proposed approach achieves a substantial performance gain in comparison to the conventional machine learning approaches. Specifically, our proposed DCAE is able to reconstruct the noisy and always changing data with less than 0.047 mean square error. Overall, our occupancy detection combining DCAE and softmax classifier achieves sub-room level accuracy for at least 99.3% of the time.**

I. Introduction

Bluetooth Low Energy (BLE) beacon is a very popular low power wireless device tailored to promote the development of the Internet of Things (IoT) applications . For example, many interesting research works have leveraged BLE beacon to deliver a better proximity sensing [1] [2], localization [3] [4], distance estimation [5] and occupancy detection [6]. While many of these works have reported to achieving a significant performance gain with radio frequency (RF) fingerprint approach, severe performance variations are observed especially when the environmental changes [7]. In other words, constant fingerprint updates and frequent calibrations are required to maintain a similar performance. In this paper, we propose autoencoder approach to learn a robust fingerprint for occupancy detection with BLE beacons.

Let $\mathcal{B} = \{b_j | 0 < j \leq M\}$ be the set of BLE beacons deployed in an indoor environment, then the RF fingerprint at a particular zone can be registered by scanning the received signal strength (RSS) from all M beacons. Specifically, the RF fingerprint is a vector $\mathbf{\Phi}^{(z_i)} \in \mathbb{R}^N$, where z_i denotes zone i. Then, the occupancy detection problem would be searching for a fingerprint vector which best matches the real-time observation. The real-time observation indicates the list of RSS values measured by a device (typically a Bluetooth compatible receiver) when the device is located at the particular zone. Note that our zone is defined according to the social nature of a space rather than a random zone on the corridor defined by [8]. For example, our zone can be a private entertainment space or a mini drinking bar inside a room. Note that by defining the zone according to the social nature is useful for many IoT applications which aim to deliver a smart service at the right time in the right place.

The occupancy detection problem described above is trivial especially when $N = M$, i.e., when each zone is associated with a beacon (Note that this is a common approach adopted by many commercialized applications). Despite such a trivial problem, it is still relatively hard to achieve perfect performance (i.e., with 100% detection accuracy) owing to the unreliable RSS measurements. In other words, the occupancy detection can be very challenging when the problem is non-trivial (i.e., when $N >> M$). In this paper, our research goal is to achieve a sub-room level occupancy detection given the non-trivial problem.

To this end, this paper proposes occupancy detection with 1) an unsupervised denoising-contractive autoencoder (DCAE) to learn a robust fingerprint representation, and 2) a supervised softmax function at the last layer for classification. Note that denoising autoencoder (DAE) is an unsupervised neural network designed to learn a robust feature against noisy input [9]; whereas contractive autoencoder (CAE) is designed to ensure the learned feature is insensitive to the input variations [10]. Consider the abilities of both autoencoders in learning a better fingerprint representation from different aspects, this paper skillfully combines both autoencoders into a single DCAE such that we can deal with 1) the always changing RSS values and 2) the noisy RSS values, at the same time. However, it is not obvious to simply combine both autoencoders since both of them are using different loss functions (cf. [9] [10]). The main contributions of this paper are summarized as follows:

- Rather than simply stacking up both autoencoders to form a deep neural network, we jointly manipulate the 2 loss functions to achieve a single DCAE.
- Instead of using synthetic data, our experiment is based on real RSS measurements, which were collected with different devices at different times, to better reflect the environmental variations.
- Our approach achieves a sub-room level occupancy detection for a non-trivial problem, where $M << N$ and with the noisy and always changing RSS measurements.

The rest of the paper is organized as follows. Section II presents the system model. Section III describes our proposed DA. Section IV illustrates the experimental setup and discusses the results. Section V concludes the paper with future works.

II. BLE-based Occupancy Detection System

For a given zone, the occupancy detection tells the presence or absence of an occupant which usually refers to a person, and is regarded as a binary classification problem with output $\mathcal{Y} \in \{0,1\}$. Without loss of generality, the occupancy detection can be extended to the multinomial classification given N zones. In this section, we first provide an overview regarding the BLE beacon before describing our system model in Section II-B.

A. Bluetooth Low Energy Beacon

BLE beacon has emerged as a promising alternative for IoT development [11]. It is an active broadcaster which broadcasts its advertising packet periodically according to the pre-defined advertising interval T_a [12]. The advertising packet can be received by any Bluetooth compatible device. The parameter of interest is the RSS, which can be measured at the receiving end. RSS is a measurement in dBm scale (i.e., $P_{r,(dBm)} = 10\log(\frac{P_{r,(watt)}}{1mW})$), which typically ranges from $-20dBm$ to $-90dBm$ subjects to the distance between the beacon and the receiver. Many RF fingerprint approaches exploit the RSS measurements to construct the fingerprint. However, RSS is unreliable in consequence to shadowing and multipath fading [13] [14]. In this paper, we also use the RSS for fingerprint construction by introducing DCAE to combat the noisy and always changing RSS measurements.

B. System Model

Fig. 1 illustrates the system model of our BLE-based occupancy detection system. Note that the system can be divided into training and detection phase. In this section, we provide a generic formulation about our system model, whereas more elaborations on how a robust fingerprint and classifier model is learned during the training phase, and how we apply the trained model for occupancy detection are provided in the next section.

Suppose that a large area can be divided into N number of zones, then we have $\mathcal{Z} = \{z_i | 0 < i \leq N\}$, in which each element in $\mathcal{Z}$ can be mapped to either 1 or 0 with 1 indicates the presence of the occupants and 0 otherwise. Mathematically, the occupancy detection in a given zone is just a function that maps $\mathcal{Z}$ to $\mathcal{Y}$, i.e., $g : \mathcal{Z} \rightarrow \mathcal{Y}$. By vectorizing the above equation for all zones, we have an N-dimensional occupancy vector $\mathbf{z}$, i.e.,

$$\begin{aligned} \mathbf{z} &= \begin{pmatrix} g(z_1) & g(z_2) & ... & g(z_i) & ... & g(z_{N-1}) & g(z_N) \end{pmatrix}^T \\ &= \begin{pmatrix} 0 & 0 & ... & 1 & ... & 0 & 0 \end{pmatrix}^T \end{aligned} \tag{1}$$

Note that the vector $\mathbf{z}$ consists of only one 1 and $N-1$ zeros that comes from the truth that the occupant can not show in several different zones simultaneously. Furthermore, at the system backend, we can know the total number of occupants in a given zone by summing up all the $\mathbf{z}$ vectors returned by occupants' devices (i.e., the smartphone or smartwatch).

Given M beacons, the relationship between the occupancy vector $\mathbf{z} \in \{0,1\}^N$ and the observed vector $\mathbf{\Phi}^{(y)} \in \mathbb{R}^M$ can be modeled as follows:

$$\mathbf{\Phi}^{(y)} = \mathbf{\Psi}\mathbf{\Omega}\mathbf{z} + \Delta \tag{2}$$

where Δ is the noise and $\mathbf{\Omega} \in \mathbb{R}^{M \times N}$ is the fingerprint matrix. Each column of $\mathbf{\Omega}$ is fingerprint vector $\mathbf{\Phi}^{(z_i)} \in \mathbb{R}^M$ for zone z_i. $\mathbf{\Psi}$ is the sparsify matrix which is related to the number of active beacons during a detection phase and should be determined in real-time.

C. Data Acquisition

Given $\mathcal{B} = \{b_j | 0 < j \leq M\}$, the number of RSS values a smartphone can measure during data acquisition is subject to the advertising interval T_a by each beacon and the length of the scanning duration T_s at the smartphone. In general, we will use longer T_s during the training phase such that we can obtain enough measurements for constructing the fingerprint, and keep the T_s as short as possible during the detection phase to ensure a real-time detection. Let $\mathcal{P} = \{P_r^{(b_j)}(t) | 0 < t \leq T_s, b_j \in \mathcal{B}\}$ be the instantaneous RSS measurements collected during T_s, each element in $\mathcal{P}$ is first grouped into a subset $\mathcal{P}^{(b_j)} \subset \mathcal{P}$ according to the source of the RSS (i.e., the beacon). The fingerprint vector is constructed by averaging the RSS measurements in each subset, i.e., $\phi_{b_j}^{(z_i)} = \frac{1}{|\mathcal{P}^{(b_j)}|} \sum P_r^{(b_j)}(t)$. After that, we rearrange all $\phi_{b_j}^{(z_i)}$ according to the sequence defined by set $\mathcal{B}$ to obtain

$$\mathbf{\Phi}^{(z_i)} = \begin{pmatrix} \phi_{b_1}^{(z_i)} & \phi_{b_2}^{(z_i)} & \ldots & \phi_{b_j}^{(z_i)} & \ldots & \phi_{b_M}^{(z_i)} \end{pmatrix}^T \tag{3}$$

By repeating the same process at each zone, a $M \times N$ dimensional fingerprint matrix can be constructed, i.e., $\mathbf{\Omega} = \begin{pmatrix} \mathbf{\Phi}^{(z_1)} & \mathbf{\Phi}^{(z_2)} & \ldots & \mathbf{\Phi}^{(z_i)} & \ldots & \mathbf{\Phi}^{(z_N)} \end{pmatrix}$.

During the detection phase, we might have the case where $\phi_{b_j}^{(z_i)} = 0$ when the smartphone might miss to capture the data from beacon b_j, and thus $\mathcal{P}^{(b_j)} = \emptyset$. These empty set will simply be discarded from being included into the observed vector $\mathbf{\Phi}^{(y)}$. Given there are only an m number of subsets $\mathcal{P}^{(b_j)} \neq \emptyset$ from set $\mathcal{P}$, the observed vector $\mathbf{\Phi}^{(y)}$ would be an m dimensional vector defined as follows:

$$\mathbf{\Phi}^{(y)} = (\phi_{b_j}^{(y)} | \mathcal{P}^{(b_j)} \neq \emptyset)^T \tag{4}$$

Since the dimension of $\mathbf{\Phi}^{(y)}$ is comparatively smaller than the dimension of the registered fingerprint, we can derive a sparsify matrix $\mathbf{\Psi}$ during data processing such that $\mathbf{\Psi}\mathbf{\Omega}$ has the same dimension as $\mathbf{\Phi}^{(y)}$.

D. Data Processing

Let $\mathcal{A} = \{a_1, a_2, ..., a_j, ..., a_m\}$ be a subset of beacons observed by the smartphone during the detection phase, the

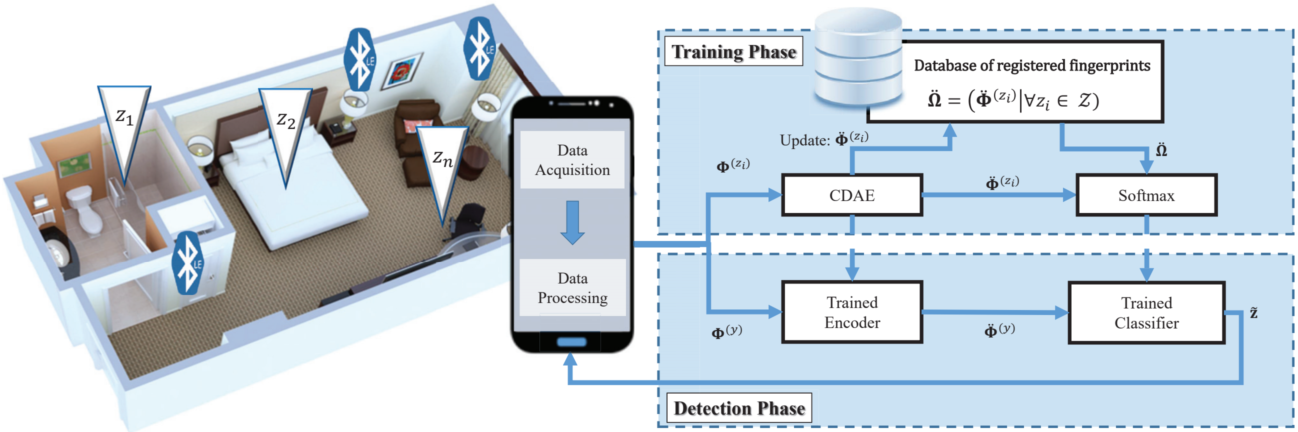

Fig. 1. The BLE-based occupancy detection consists of the following two major phases: a training phase to learn the robust fingerprint, and a detection phase to deliver real-time occupancy detection.

sparsify matrix can be derived by comparing the elements in set $\mathcal{A}$ with set $\mathcal{B}$

$$\Psi = \begin{pmatrix} f(a_1, b_1) & \dots & f(a_1, b_i) & \dots & f(a_1, b_M) \\ \vdots & \dots & \vdots & \dots & \vdots \\ f(a_i, b_1) & \dots & f(a_i, b_i) & \dots & f(a_i, b_M) \\ \vdots & \dots & \vdots & \dots & \vdots \\ f(a_m, b_1) & \dots & f(a_m, b_i) & \dots & f(a_m, b_M) \end{pmatrix} \quad (5)$$

where $f(\cdot)$ is the function which returns 1 when $a_i = b_i$ and 0 otherwise. Upon obtaining Ψ, the fingerprint matrix Ω can be refined by multiplying both matrices, i.e., $\Psi\Omega$. According to Eq. (3) and (4), each element ϕ in $\Phi^{(z_i)}$ and $\Phi^{(z_i)}$ is based on the raw RSS data, which is susceptible to noise and environmental variations. In the next section, we present our DCAE to learn a robust fingerprint. Furthermore, a softmax function is added at the last layer to train a classifier for occupancy detection.

III. Fingerprint Learning with Autoencoder

Recently, many works have started to exploit autoencoder to boost the system performance, for example, . [15] leverages DAE to learn a robust fingerprint for 3D localization, whereas [16] exploits deep autoencoder (by stacking up a few autoencoders) to learn a robust fingerprint with WiFi technology. However, the fine granularity of 3D positioning complicates the occupants counting, and the stacking of autoencoder imposes expensive computation during the training phase. In this paper, we propose a single DCAE to learn a robust fingerprint rather than stacking up a few autoencoders.

This section first presents our proposed DCAE, and then describes the softmax classifier in Section III-B.

A. Training Phase

Autoencoder is an unsupervised neural network which learns to reconstruct the output given the input data. For example, suppose that we feed the unlabeled fingerprint sample $\Phi \in \mathbb{R}^M$ to the autoencoder, the autoencoder will learn to reconstruct $\ddot{\Phi} \in \mathbb{R}^M$ through backpropagation. Note that the reconstruction problem is non-trivial since the autoencoder impose a bottleneck (i.e., ensure the number of hidden units H is smaller than the size of the input data) to its hidden layer.

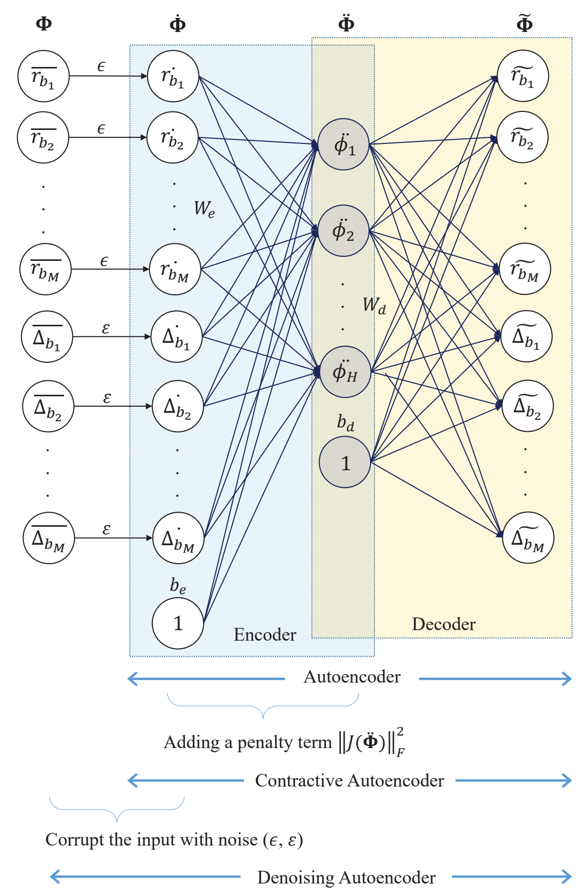

Fig. 2. Our proposed DCAE trains a model to learn a robust fingerprint representation by adding some noise Δ to the training data $\tilde{\Phi}^{(t)}$.

In general, the autoencoder consists of 2 parts: encoder and decoder. The encoder learns a hidden representation given the input data; whereas the decoder reconstruct the output data from the hidden representation. Given the fingerprint vector

$\mathbf{\Phi} \in \mathbb{R}^M$, the encoder and decoder function can be described as follows:

$$\begin{aligned} \dot{\mathbf{\Phi}} &= \sigma(W_e \mathbf{\Phi} + b_e) \\ \ddot{\mathbf{\Phi}} &= \sigma(W_d \dot{\mathbf{\Phi}} + b_d) \end{aligned} \tag{6}$$

where $W_e \in \mathbb{R}^{H \times M}$ and $b_e \in \mathbb{R}^H$ are the weight and bias learned by the encoder and $W_d \in \mathbb{R}^{M \times H}$ and $b_d \in \mathbb{R}^M$ are the weight and bias learned by the decoder. $\sigma(\cdot)$ is the neuron activation function, which can be either a sigmoid or tanh function. Hence, the dimension of encoded feature (i.e., the hidden representation) $\dot{\mathbf{\Phi}} \in \mathbb{R}^H$ is always smaller than the dimension of the input fingerprint $\mathbf{\Phi} \in \mathbb{R}^M$. The optimal weight and bias can be learned via the following loss function:

$$\mathcal{L}(\mathbf{\Phi}, \ddot{\mathbf{\Phi}}) = \|\ddot{\mathbf{\Phi}} - \mathbf{\Phi}\|_2^2 \tag{7}$$

Given the above loss function, we can train our autoencodeer to minimize $\mathcal{L}(\cdot)$ through backpropagation.

Fig. 2 depicts the neural network architecture of our proposed DCAE. In general, DCAE uses the same encoder and decoder function as the conventional autoencoder described above. However, we impose some noise $\Delta \sim \mathcal{N}(0, 1)$ to the training data $\tilde{\mathbf{\Phi}}^{(t)}$ with respect to the expected fingerprint vector obtained by Eq. (3). Since there might some variations between the fingerprint samples collected from the same zone, we refine the loss function $\mathcal{L}(\cdot)$ by adopting the penalty term introduced by the CAE. Hence, DCAE redefine the loss function by Eq. (7) to the following:

$$\begin{aligned} \mathcal{L}(\mathbf{\Phi}, \ddot{\mathbf{\Phi}}) &= \|\ddot{\mathbf{\Phi}} - \mathbf{\Phi}\|_2^2 + \lambda \|J(\tilde{\mathbf{\Phi}}^{(t)})\|_F^2 \\ &= \|\sigma(W_d \dot{\mathbf{\Phi}} + b_e) - \mathbf{\Phi}\|_2^2 + \lambda \|J(\tilde{\mathbf{\Phi}}^{(t)})\|_F^2 \\ &= \|\sigma(W_d \sigma(W_e \tilde{\mathbf{\Phi}}^{(t)} + b_e) + b_d) - \mathbf{\Phi}\|_2^2 \\ &\qquad + \lambda \|J(\tilde{\mathbf{\Phi}}^{(t)})\|_F^2 \end{aligned} \tag{8}$$

where the decoder output $\ddot{\mathbf{\Phi}}$ is reconstructed with respect to the noisy sample $\tilde{\mathbf{\Phi}}^{(t)}$ instead of $\mathbf{\Phi}$. The second term is the penalty term $\|J(\tilde{\mathbf{\Phi}}^{(t)})\|_F^2$ with regulation parameter λ. $\|J(\tilde{\mathbf{\Phi}}^{(t)})\|_F^2$ is known as the Frobenius norm of the jacobian matrix, which can be calculated as follows:

$$\|J(\tilde{\mathbf{\Phi}}^{(t)})\|_F^2 = \sum_{mh} \left(\frac{\partial \dot{\phi}_h}{\partial \tilde{\phi}_m^{(t)}} \right)^2 \tag{9}$$

The goal of DCAE is to search for the optimal W_e, b_e, W_d, b_d that minimize the loss function described in Eq. (8). This can be achieved via stochastic gradient descent or more advanced optimization method such as limited-memory BFGS.

B. Softmax Classification

Given the learned fingerprint representation, we added a softmax function to the last layer to train a multinomial classifier. Refer back to Fig. 1, the softmax layer can obtain the label (i.e., the zone) of the learned fingerprint from the fingerprints database. Hence, we can train a supervised classifier with the softmax function. Mathematically, softmax function can be described as follows:

$$P(\tilde{z} = z_i | \ddot{\mathbf{\Phi}}) = \frac{exp(W_s \ddot{\mathbf{\Phi}} + b_s)}{\sum_{k=1}^{N} exp(W_s \ddot{\mathbf{\Phi}}^{(z_k)} + b_s)}, \quad \forall z_i \in \mathcal{Z} \tag{10}$$

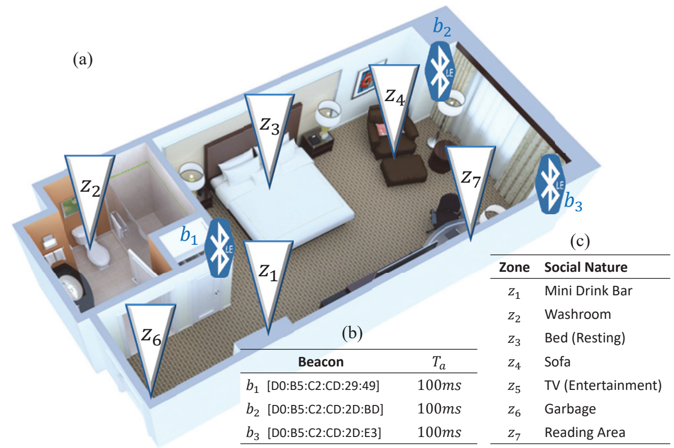

Fig. 3. The experiment was conducted inside a cozy room. (a) The floor plan of the room and the location of the deployed beacons, (b) the configuration of each beacon, and (c) the divided zone according its social nature.

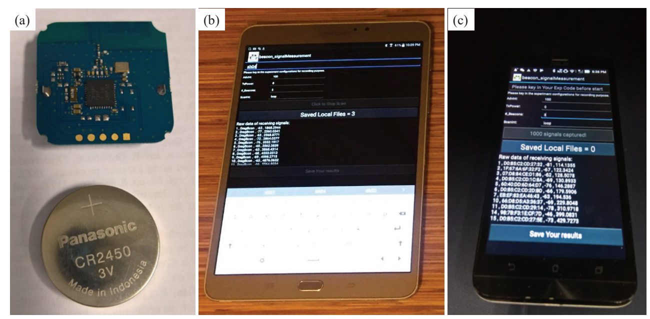

Fig. 4. (a) The deployed beacon is based on CC2541 chipset from Texas Instrument. Two devices were used to collect the RSS measurements: (b) Samsung Tablet, and (c) Asus Smartphone.

where $W_s \in \mathbb{R}^{N \times M}$ and $b_s \in \mathbb{R}^N$ are the weight and bias learned by the softmax function. Then, the softmax classifier can be trained by minimizing the following cross-entropy loss,

$$\min_{W_s, b_s} (\mathcal{L}(z, p)) = -\sum\nolimits_i z_i log(p_i)) \tag{11}$$

where $p_i = P(\tilde{z} = z_i | \ddot{\mathbf{\Phi}})$.

IV. Experiment and Results

This section describes our experimental setup for data collection. After that we evaluate the performance of our proposed approach in Section IV-C.

A. Experimental Testbed

We have our experimental testbed set up inside a cozy room offered by a 5-star hotel, Rio Hotel in Macau [1]. Three beacons were deployed, as marked by the Bluetooth symbols in Fig. 3(a). The deployed beacons are based on the CC2541 chipset from Texas Instrument, as depicted in Fig. 4(a), each beacon was configured to have the same T_a. We divide the room into seven zones, as indicated in Fig. 3(c). Fig. 4(b)

[1]"Rio Hotel", https://www.riomacau.com/

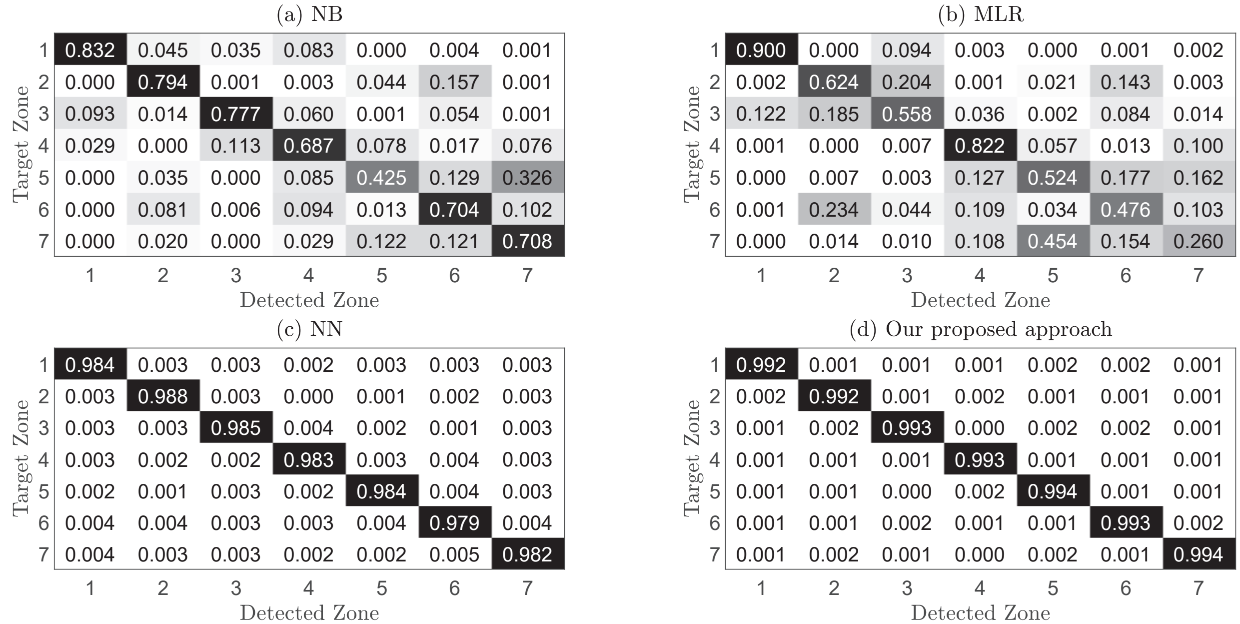

Fig. 5. The confusion matrix indicates the classification accuracy achieved by (a) NB, (b) MLR, (c)NN, and (d) our proposed approach combining DCAE and softmax classifier.

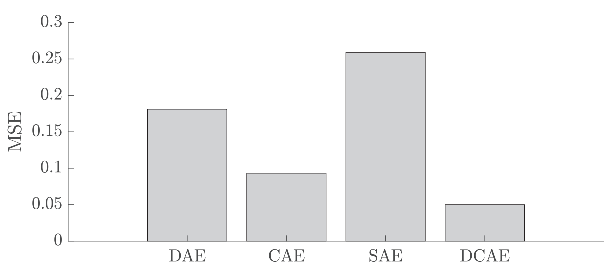

Fig. 6. The reconstruction error suffers by our approach (DCAE) in comparison to DAE, CAE and SAE.

and (c) show the two different devices (i.e., Samsung tablet and Asus Smartphone) which were used to collect the RSS measurements. Total 1000 measurements were collected at each zone by each device resulting in a total of 14000 measurements from all zones. The collected data was uploaded as .csv file to the two servers located at two different locations: Hong Kong and Korea.

B. Dataset

The measurements data consists of the following information: beacon MAC address, RSS value, timestamp and data payload. After consolidating the data from both servers, we reorganize the data and extract the RSS measured from each beacon according to the logged timestamp. The reorganize data consists of $98k$ fingerprint samples. We divided the data into training and testing sample, i.e., $70k$ as training and $28k$ as testing. The challenge with our current dataset is that the dimension of the fingerprint vector is very small. Such a small dimension is unfavorable to autoencoder because autoencoder is generally used to compress the high dimension input data to low dimension hidden representation. Both original and reorganized dataset are available in Github[2] for download.

C. Performance Evaluation

We first evaluate the reconstruction error by comparing our proposed DCAE with another 3 variants of autoencoders, i.e., stacked autoencoder (SAE), DAE and CAE. Note that DAE has been used by the work [15] for 3D localization, whereas SAE has been used by [16]. The reconstruction error is computed by taking the mean squared error (MSE) between the reconstructed data with the input data. According to the result shown in Fig. 6, our proposed DCAE outperforms the rest with the minimum MSE (i.e., around 0.047). Since SAE did not apply any denoising function, its performance is the worst. One of the reasons that CAE achieves better performance than DAE is due to the RSS variations of the testing sample has a greater effect on the reconstruction process than the ambient noise.

Next, we evaluate the classification accuracy of our proposed approach in comparison to the conventional classification techniques such as naive Bayes (NB), multinomial logistic regression (MLR) and neural network (NN) classifier. Our approach enables classification by adding a softmax function at the end of the reconstruction output, as described in Section III-B. Note that NN classifier has been used by [17] for occupancy detection and by [8] for zone-based localization, whereas [18] use MLR for indoor localization.

Confusion matrices are used to visualize the classification accuracy achieved by all the classifiers described above. The results are illustrated in Fig. 5. Our proposed approach outperforms the rest with 99.31% accuracy (the accuracy can

[2]"https://github.com/pcngnotgood/Data-from-Rio-Hotel"

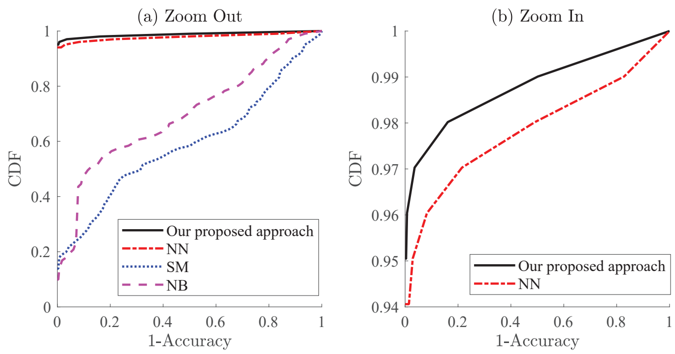

Fig. 7. The CDF achieved by all the approaches: (a) Zoom out view, (b) Zoom in view.

be computed by averaging all the diagonal elements). The accuracy achieved by NN is 98.36%, NB 70.32%, and MLR 59.44%. Even though our proposed approach only improves 0.95% from NN classifier, the training time used by our proposed approach is relatively shorter than the NN training time. In particular, our proposed approach only use less than $3min$ to train the classifier with $70k$ training samples, whereas NN use more than $30min$ to finish the training. The short training time and high accuracy indicate the superiority of our proposed approach to NN.

We repeated the same experiment for 270 times by randomly sampling 1000 samples from the testing samples. For each 1000 sub-testing samples, we further inject some random noise to the data. The accuracy for all the 270 times were logged. After that, we computed the cumulative distribution function (CDF) by using the Matlab function - "ecdf". According to the result shown in Fig. 7, it is clear that our proposed approach is able to achieve 99.3% at least 96% of the time. Such a result indicates that our proposed approach can still maintain its superior performance under a noisy environment by learning a robust fingerprint representation with our DCAE.

V. Conclusions

This paper achieves a sub-room level occupancy detection with robust RF fingerprint learned by our proposed DCAE. Specifically, our proposed DCAE is robust towards the RSS variations and environmental noise. The experimental results indicate that our proposed approach achieves a substantial performance gain in comparison to the conventional autoencoders and machine learning techniques. In this work, we use a fixed dictionary to expand the dimensionality of the fingerprint vector such that we can still apply the autoencoder to learn a robust fingerprint. While most of the autoencoders require the size of the hidden units to be smaller than the size of the input, sparse autoencoder, on the other hand, is able to encode meaningful hidden representation even though the size of the hidden units is much larger than the size of the input data. Hence, in future work, we can add a sparse autoencoder to enlarge the dimensionality of the encoded feature given the low dimensional fingerprint vector.

Acknowledgment

This work was supported by the frame of international cooperation program managed by the National Research Foundation of Korea under Grant 2017K2A9A2A06016102. This work was also supported by HKUST-NIE Social Media Laboratory.

References

[1] F. Yin, Y. Zhao, F. Gunnarsson, and F. Gustafsson, "Received-signal-strength threshold optimization using gaussian processes," *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 2164–2177, April 2017.

[2] P. C. Ng, J. She, and S. Park, "High resolution beacon-based proximity detection for dense deployment," *IEEE Transactions on Mobile Computing*, vol. 17, no. 6, pp. 1369–1382, June 2018.

[3] R. Faragher and R. Harle, "Location fingerprinting with bluetooth low energy beacons," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 11, pp. 2418–2428, Nov 2015.

[4] Y. H. Ho and H. C. B. Chan, "Blueprint: Ble positioning algorithm based on nufo detection," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, Dec 2017, pp. 1–6.

[5] C. H. Lam, P. C. Ng, and J. She, "Improved distance estimation with ble beacon using kalman filter and svm," in *2018 IEEE International Conference on Communications (ICC)*, May 2018, pp. 1–6.

[6] P. Barsocchi, A. Crivello, M. Girolami, F. Mavilia, and F. Palumbo, "Occupancy detection by multi-power bluetooth low energy beaconing," in *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Sept 2017, pp. 1–6.

[7] X. Tian, R. Shen, D. Liu, Y. Wen, and X. Wang, "Performance analysis of rss fingerprinting based indoor localization," *IEEE Transactions on Mobile Computing*, 2016.

[8] N. Anzum, S. F. Afroze, and A. Rahman, "Zone-based indoor localization using neural networks: A view from a real testbed," in *2018 IEEE International Conference on Communications (ICC)*, May 2018, pp. 1–7.

[9] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning*, ser. ICML '08, 2008, pp. 1096–1103.

[10] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: Explicit invariance during feature extraction," in *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ser. ICML'11, 2011, pp. 833–840.

[11] M. Collotta, G. Pau, T. Talty, and O. K. Tonguz, "Bluetooth 5: A concrete step forward toward the iot," *IEEE Communications Magazine*, vol. 56, no. 7, pp. 125–131, July 2018.

[12] P. C. Ng, L. Zhu, J. She, R. Ran, and S. Park, "Beacon-based proximity detection using compressive sensing for sparse deployment," in *2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, June 2017, pp. 1–6.

[13] M. Ayadi and A. B. Zineb, "Body shadowing and furniture effects for accuracy improvement of indoor wave propagation models," *IEEE Transactions on Wireless Communications*, vol. 13, no. 11, pp. 5999–6006, Nov 2014.

[14] J. Yang, X. Wang, S. I. Park, and H. M. Kim, "Optimal direct path detection for positioning with communication signals in indoor environments," in *2012 IEEE International Conference on Communications (ICC)*, June 2012, pp. 4798–4802.

[15] C. Xiao, D. Yang, Z. Chen, and G. Tan, "3-d ble indoor localization based on denoising autoencoder," *IEEE Access*, vol. 5, pp. 12 751–12 760, 2017.

[16] W. Zhang, K. Liu, W. Zhang, Y. Zhang, and J. Gu, "Deep neural networks for wireless localization in indoor and outdoor environments," *Neurocomput.*, vol. 194, no. C, pp. 279–287, Jun. 2016.

[17] T. Ekwevugbe, N. Brown, V. Pakka, and D. Fan, "Real-time building occupancy sensing using neural-network based sensor network," in *2013 7th IEEE International Conference on Digital Ecosystems and Technologies (DEST)*, July 2013, pp. 114–119.

[18] Z. Peng, Y. Xie, D. Wang, and Z. Dong, "One-to-all regularized logistic regression-based classification for wifi indoor localization," in *2016 IEEE 37th Sarnoff Symposium*, Sept 2016, pp. 154–159.