Energy & Buildings 174 (2018) 372–379

Contents lists available at ScienceDirect

Energy & Buildings

journal homepage: www.elsevier.com/locate/enbuild

Building occupancy modeling using generative adversarial network

Zhenghua Chen, Chaoyang Jiang*

School of Electrical and Electronics Engineering, Nanyang Technological University, 50 Nanyang Ave, Singapore

A R T I C L E I N F O

Article history:
Received 7 April 2018
Revised 26 May 2018
Accepted 15 June 2018
Available online 4 July 2018

Keywords:
Building occupancy modeling
Generative adversarial network
Neural networks

A B S T R A C T

Due to the energy crisis and the awareness of sustainable development, the research on energy-efficient buildings has increasingly attracted attention. To achieve this objective, one important factor is to capture occupancy properties for building control systems, which refers to occupancy modeling in buildings. Due to the complexity of building occupancy, previous works try to simplify the modeling with some specific assumptions which may not always hold. In this paper, we propose a Generative Adversarial Network (GAN) framework for building occupancy modeling without any prior assumptions. The GAN approach contains two key components, i.e. a generative network and a discriminative network, which are designed as two powerful neural networks. Owing to the strong generalization capacity of neural networks and the adversarial mechanism in the GAN approach, it is able to accurately model building occupancy. We perform real experiments to verify the effectiveness of the proposed GAN approach and compare it with two state-of-the-art approaches for building occupancy modeling. To quantify the performance of all the models, we define five variables with two evaluation criteria. Results show that our proposed GAN approach can achieve a superior performance.

1. Introduction

Energy consumed in buildings accounts for around 40% of total energy usage in the world. Energy saving in buildings is an urgent task. Since occupants determine the cooling/heating load and ventilation requirement of an indoor environment, the number of occupants known as occupancy is vital for energy efficiency in buildings [1]. For example, the authors in [2] proposed an occupancy-driven control system which is able to save 15% of the total energy. Occupancy information also indicates the lighting requirement. If no occupants are present, all the lights can be automatically switched off to save energy. An adaptive lighting system was presented in [3] which demonstrates an energy saving of 35 – 75% for lighting systems in buildings. Moreover, the occupancy information can also be used in emergency egress for efficient evacuation and rescue [4]. Occupancy information is of great importance and with high complexity. In order to understand the properties of occupancy, in this paper, we mainly focus on occupancy modeling. The objective is to capture regular occupancy dynamics which can be leveraged to improve the performance of real-time occupancy estimation [5,13]. Moreover, occupancy models are able to generate occupancy time series for energy simulation tools, such as EnergyPlus [6] and DeST [7], which can simulate the energy performance of buildings before construction so that the facilities of buildings can be sized based on the simulation results. Since occupants are directly related to the energy consumption in buildings, occupancy dynamics is also widely used in enhancing the performance of energy forecasting [8].

Occupancy modeling is to model occupancy dynamics, meaning to understand the distribution of occupancy data. With this model, we can predict the number of occupants at any time instance without sensor observations. Many advanced occupancy models have been presented in prior works. Wang et al. proposed a probabilistic model for occupancy modeling in a single person office [9]. Two exponential distributions were leveraged to model occupied and vacant intervals. The simulation results showed that the vacant interval follows the exponential distribution, but the occupied interval violates it. Page et al. presented an inhomogeneous Markov chain model with two states, i.e. presence and absence, for occupancy modeling in single person offices [10]. They also defined a parameter of mobility to calculate the transition probability matrix of the inhomogeneous Markov chain model. In experiments, some important variables, such as the first arrival and last departure times of an occupant, were defined to evaluate the performance of their proposed model.

The occupancy in single person offices is relatively easy to be modelled. But the modeling of occupancy in multi-occupant rooms is more difficult. Richardson et al. presented a first-order

* Corresponding author.
E-mail addresses: chen0832@e.ntu.edu.sg (Z. Chen), chaoyangjiang@hotmail.com (C. Jiang).

https://doi.org/10.1016/j.enbuild.2018.06.029
0378-7788/

inhomogeneous Markov chain model for occupancy modeling in a residential building [11]. They defined the state of the inhomogeneous Markov chain as the number of active occupants. This inhomogeneous Markov chain model is an extension of the model in [10] where the authors defined the state as the presence and absence of a single person office. Since the number of active occupants is small for residential buildings, the Markov transition probability matrix will not be very large, which makes their proposed approach applicable. But this case does not hold for commercial buildings where a relatively large number of occupants will share one room. For example, a room with 20 occupants will yield a transition probability matrix with a dimension of 21×21 (states from 0 to 20) which is difficult to solve. Hence, in [12], Chen et al. proposed a novel inhomogeneous Markov chain model for occupancy modeling in commercial buildings. In a multi-occupant room, they defined the state of the inhomogeneous Markov chain model as the increment of occupancy with the assumption that only one occupant moves into or out of a room in a short time interval. In this way, the transition probability matrix will be with a dimension of 3×3 regardless of the total number of occupants in the room.

Gunathila et al. proposed a generalized event-driven approach for building occupancy modeling [14]. They assigned each occupant into groups based on some similar properties. Then, the occupant behaviour is fully driven by group and personal events. In [15], Wang et al. also presented an event-driven approach for the modeling of building occupancy [15]. They divided one day into some events, i.e. walking around, going to the office, getting off work and lunch break. Within each event, a Markov chain model was utilized for simulating occupancy patterns. There are two limitations for these two event-driven approaches. The first one is that the definition of events is ambiguous, and some personal events, such as sick and emotion changes, are difficult to define. The second one is that these two works only apply simulation data for model verification instead of real occupancy data.

Another relevant work can be found in [16]. The authors presented an agent-based model with some intuitively defined modules for the modeling of building occupancy. For instance, they defined a damping process for occupants who are in their working places and an acceleration process for occupants who are in hallways or restrooms. These intuitively defined modules lack a theoretical support. Thus, their proposed approach has limited performance. Jia et al. proposed a queueing approach to model building occupancy [17]. The designed queues have an infinite number of servers with time-varying parameters. To deal with the problem of abrupt changes of occupancy, they developed a piecewise homogeneous queue with the adaptation of the length of each homogeneous piece based on occupancy variations. However, their proposed method is built upon the assumption that the occupied time follows an exponential distribution which violates the conclusion in [9]. A comprehensive review of the approaches for building occupancy modeling can be found in [18].

Due to the complex behaviour of occupants, the occupancy data is very difficult to model. Existing works for occupancy modeling based on Markov chains or agent-based schemes all require the data to follow some assumptions such as first-order Markov property. Recently, a new powerful generative model named Generative Adversarial Network (GAN) has been developed [19]. It has been successfully applied in many challenging research areas, such as image processing [20] and natural language processing [21]. Owing to the powerful modeling ability, the GAN is able to model complex data even with some implicit distributions, and it does not require the data to follow any specific assumptions. Therefore, it is naturally suitable for the task of occupancy modeling. In this paper, we propose a GAN framework for building occupancy modeling. Owing to the universal approximation ability of neural networks, the GAN approach is able to learn the implicit distribution of real occupancy data for occupancy modeling. The GAN approach consists of two components: a generative network and a discriminative network. Firstly, the generative network is able to produce an occupancy time series with some random inputs. Then, the discriminative network attempts to distinguish these generated occupancy time series from the real ones. The final objective is to learn these two networks so that the generative network can produce the occupancy time series with the same distribution of the real occupancy time series and the discriminative network is unable to separate the occupancy time series generated by the generative network and the real occupancy time series. We perform real experiments to evaluate the performance of the proposed approach for occupancy modeling. Moreover, a comparison with some state-of-the-art approaches has also been made using real experimental data.

The main contributions of this work are summarized as follows:

- We propose a GAN framework with two powerful neural networks for the complex task of occupancy modeling without any prior assumptions.
- We apply real experimental data to verify the effectiveness of the proposed approach.
- The proposed GAN framework outperforms the state-of-the-art techniques for occupancy modeling.
- We evaluate the impact of one key parameter of neural networks in the proposed GAN framework, i.e., the number of hidden nodes, on the performance of occupancy modeling.

The remaining of the paper is organized as follows: In Section 2, we introduce the basic theory of GAN, followed by the proposed GAN framework for building occupancy modeling. In Section 3, firstly, we present the data collection process and define some variables and criteria to quantify the performance of building occupancy modeling. Then, the experimental setup is presented. After that, we demonstrate the experimental results with some discussions. Finally, we investigate the impact of one key hyperparameter of the proposed GAN approach on modeling performance. In Section 4, we conclude this work and present some potential future works.

2. Methodology

In this section, we show how the GAN is used for occupancy modeling. Firstly, we briefly introduce the basic theory of GAN. Then we propose a GAN framework for building occupancy modeling.

2.1. Generative adversarial network

The generative adversarial network was first proposed by Goodfellow et al. in [19]. It intends to learn the distribution of the given data and then generates the data under the same distribution. More specifically, it contains two parts, i.e. a generative (G) network and a discriminative (D) network. For the G network, the input can be a noise vector $\mathbf{z}$ if no prior knowledge is available. The G network attempts to map the vector $\mathbf{z}$ to data space as $G(\mathbf{z}, \theta_G)$ where θ_G consists of the parameters for the G network, and $G(\cdot)$ is a type of neural network, such as a multilayer perceptron. For the D network, it can be treated as a classifier for a binary classification. Specifically, it tries to distinguish the data generated by the G network $G(\mathbf{z}, \theta_G)$ from the real data $\mathbf{r}$. We define $D(\mathbf{x}, \theta_D)$, where θ_D consists of the parameters for the D network, as the probability that the data $\mathbf{x}$ is from the real data instead of the one generated by the G network. Here, the function $D(\cdot)$ is a classifier. The main objectives of the GAN are: 1) to maximize the classification accuracy for the D network which means to correctly identify the

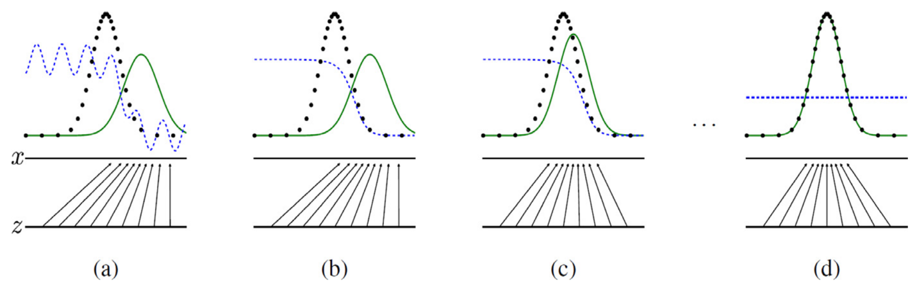


Fig. 1. An example for the GAN [19].

generated data from the real one; and 2) to simultaneously maximize the probability that the D network treats the data generated by the G network as the real data, which is equivalent to minimize $\log(1-D(G(\mathbf{z})))$. In another perspective, we can treat the GAN as a min-max game with the following optimization function:

$$\min_G \max_D O(G,D) = \mathbb{E}_{\mathbf{r}}[\log(D(\mathbf{r}))] + \mathbb{E}_{\mathbf{z}}[\log(1-D(G(\mathbf{z})))]. \tag{1}$$

To better illustrate the operation of the GAN, we apply a simple example shown in Fig. 1. The black dot and the green lines are the real data and the data generated by the G network, respectively. The blue dot represents the probability that the data comes from the real data. Here, $\mathbf{z}$ is the uniform inputs for the G network. The arrow indicates the mapping operation of the G network, which means $\mathbf{x} = G(\mathbf{z})$. At the first stage (stage (a) in Fig. 1), the D network has a limited performance in separating the data generated by the G network and the real data before training. In stage (b), the D network is trained to accurately classify the data. Then, in the next stage, the G network is adjusted based on the classification results of the D network to move towards the region where the data generated will be more likely to be identified as the real data. In the final stage, both G and D networks are well trained so that the distribution of the data generated by the G network is the same as the distribution of the real data. At this time, the D network is unable to separate the data from these two distributions, which means $D(\mathbf{x}) = 0.5$ for all the data.

The GAN has been successfully applied in many applications, for example generating images with a specific style [22]. Due to the complex behavior of occupants, the occupancy data is very difficult to model. The GAN is built upon neural networks which have strong modeling ability, it is able to model complex data even with some implicit distributions. Existing occupancy modeling approaches all require to assume the data follows some specific assumptions, such as first-order Markov property. The complex occupancy data may not follow these assumptions, and it may have some other implicit properties which have not been considered during the modeling, which seriously degrades the performance of the existing modeling approaches. The GAN does not require any specific assumptions about the data. Therefore, the GAN is naturally suitable for the task of occupancy modeling. Next, we present the proposed GAN framework for building occupancy modeling.

2.2. The proposed GAN framework for building occupancy modeling

We intend to explore the GAN for building occupancy modeling. The proposed GAN framework for building occupancy modeling is shown in Fig. 2. Firstly, the samples from the real data and the data generated by the G network are utilized as training samples for the D network. Here, the sample refers to the occupancy data. Then, the D network predicts the label (real or fake) for each sample. Finally, we can compute the loss for adjusting the parameters of the D and G networks. Due to the different functionality of the G and D networks, we design two different neural networks for them. A general framework for neural networks is shown in Fig. 3. Normally, a neural network contains three layers, i.e. an input layer, a hidden layer and an output layer. Every node is a processing unit with different activation functions. And each line has a weight and a bias indicating the importance of the connection. For the G network, the designed neural network can be formulated as follows:

$$h_k = \sum_{i=1}^{L} \Phi\left(\sum_{j=1}^{n} \alpha_{ij}^G z_i + \beta_i^G\right)\omega_{ki}^G + \gamma_k^G, \tag{2}$$

where h_k is the kth component in the output sample, L is the number of hidden nodes, n is the number of components in the input vector, $\{\alpha_{ij}^G, \beta_i^G, \omega_{ki}^G, \gamma_k^G\}$ are the parameters of the G network, and $\Phi(\cdot)$ is the activation function. Here, we employ the efficient rectified linear unit (ReLU) function [23] as the activation function. For the D network, it is a binary classifier. Thus, it only has one output, i.e. 1 meaning that the input sample comes from the real data and 0 meaning that the input sample is generated by the G network, for the neural network in Fig. 3. The mathematical expression of the D network is as follows:

$$d = \Gamma\left(\sum_{i=1}^{L} \Phi\left(\sum_{j=1}^{n} \alpha_{ij}^D z_i + \beta_i^D\right)\omega_i^D + \gamma^D\right), \tag{3}$$

where d indicates the probability that the sample $\mathbf{z}$ is from the real data, $\Gamma(\cdot)$ is the Sigmoid activation function which can provide the probability for classification, and $\{\alpha_{ij}^D, \beta_i^D, \omega_{ki}^D, \gamma_k^D\}$ are the parameters for the D network.

The learning of the D and G networks is of great importance. Based on the training loss of the GAN, gradient-based methods can be implemented for the optimization of model parameters. For the D network, the error gradient g can be expressed as

$$g_D = \nabla_{\theta_D} \frac{1}{m} \sum_{i=1}^{m} [\log(D(\mathbf{r})) + (1 - \log(D(G(\mathbf{z}))))], \tag{4}$$

where m is the batch size for learning. The gradient for the G network can be expressed as

$$g_G = \nabla_{\theta_G} \frac{1}{m} \sum_{i=1}^{m} (1 - \log(D(G(\mathbf{z})))). \tag{5}$$

Here, we use the optimization method of ADAM [24] which computes adaptive learning rates for each parameter for the optimization of all the model parameters. Specifically, assume that θ_t is the parameter to be optimized at time step t, and g_t is the corresponding gradient, the updating of θ_{t+1} using ADAM is given as

$$\begin{aligned}
\alpha_t &= r_1\alpha_{t-1} + (1-r_1)g_t \\
\beta_t &= r_2\beta_{t-1} + (1-r_2)g_t^2 \\
\alpha_t &= \alpha_t/(1-r_1) \\
\beta_t &= \beta_t/(1-r_2) \\
\theta_{t+1} &= \theta_t + \frac{\eta}{\sqrt{\beta_t}+\epsilon}\alpha_t
\end{aligned} \tag{6}$$

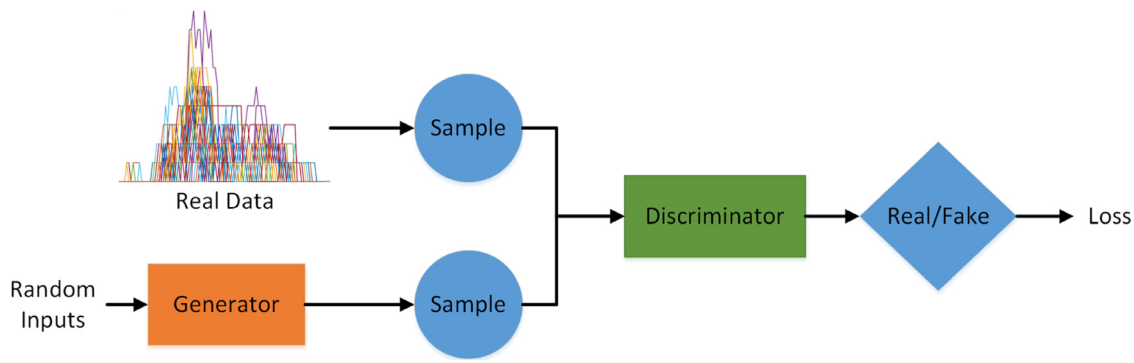


Fig. 2. The proposed GAN framework for building occupancy modeling.

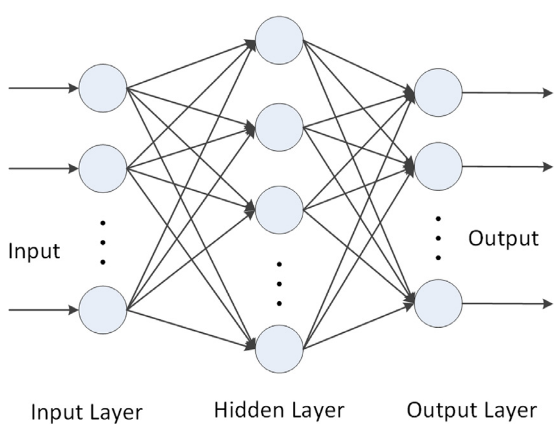


Fig. 3. The general framework for neural networks.

where α_t and β_t are the first and second moments of the gradient respectively, η is the learning rate, and the parameters r_1, r_2 and ϵ are set to be 0.9, 0.999 and 1×10^{-8} respectively. The learning of the D and G networks will be terminated when the output of the D network is around 0.5 for both the occupancy generated by the G network and the ground truth occupancy.

In summary, with the proposed GAN framework, we can learn the D and G networks with a known database (training data). Then, the learned G network is the required occupancy model which can be used to generate occupancy data with random inputs.

3. Experiment

In this section, we first introduce the data for experiments and define some variables and evaluation criteria to quantify the performance of building occupancy modeling. Then, the experimental setup is presented. After that, the experimental results are demonstrated and discussed. Finally, we investigate the impact of one key hyperparameter of the proposed GAN framework on modeling performance.

3.1. Data collection

The data for experiments was collected from a room of MAE-B building in the University of Florida campus by the authors in [16]. The room serves five graduate students who perform research works there regularly and three undergraduate research assistants who come occasionally. Moreover, some undergraduate students may visit the room to meet some graduate students who are teaching assistants. A camera was installed at the entrance of the room. It is able to capture the number of occupants moving into or out of the room. Note that, to guarantee a high accuracy of room occupancy, manual counting is performed for the camera data to determine the number of occupants. Due to the high frame-rate of the camera, the obtained occupancy has a high resolution of seconds. Since previous works suggest a resolution of 15 min [12,16], we convert the raw occupancy data into a resolution of 15 min through down-sampling for easy comparison. Finally, four weeks of data were collected from January 2010 to April 2010. Due to some technique issues, only 70 days of data can be used, which contains 12 days of data for Monday, 10 for Tuesday, 7 for Wednesday, 11 for Friday, 11 for Saturday and 9 for Sunday. According to all the 70 days of data, the largest ground truth occupant number of the room is 9. More detailed description of the data can be found in [16].

3.2. Variables and criteria

Since the occupancy model is to generate the realizations of a random process, the comparison of individual traces generated by the occupancy model with ground truth occupancy traces is not appropriate. A satisfactory occupancy model should be able to produce occupancy data which has similar statistical properties with ground truth occupancy data. Therefore, we define the variable of mean occupancy profile to see whether the generated occupancy data follows the same trend as the ground truth occupancy data. In addition, we also define four random variables, i.e., time of first arrival, time of last departure, cumulative occupied duration, and number of occupied/unoccupied transitions, which are of great importance for building control systems, to compare some properties of the generated occupancy data with the ground truth occupancy data. These variable have also been used in [12,16] for model evaluation. The definitions of these variables are the same as those defined in [12,16]. Assume that $Q_{occupied}$ and $Q_{unoccupied}$ are the two thresholds for occupied and unoccupied status respectively, and T is the time interval, the five variables can be defined as:

- *Mean occupancy*: The mean occupancy of the room at a time step t is defined as $E(x_t)$, where $E(\cdot)$ is the expectation operation.
- *Time of first arrival*: The time of first arrival is the first time when the room is occupied. Specifically, if $x_t > Q_{unoccupied}$ and $x_{0:t-1} \leqslant Q_{unoccupied}$, then t is the time of first arrival.
- *Time of last departure*: The time of last departure is the time when the room is unoccupied. Specifically, if $x_t > Q_{unoccupied}$ and $x_{t:\,24 \times 60/T} \leq Q_{unoccupied}$, then, $t+1$ is the time of last departure.
- *Cumulative occupied duration*: The cumulative occupied duration is the total length of time when the room becomes occupied. Specifically, it is the number of elements of the set $\{t | x_t \geq Q_{occupied},\ 1 \leq t \leq 24 \times 60/T\}$.
- *Number of occupied/unoccupied transitions*: The number of occupied/unoccupied transitions is the total number of tran-

sitions between "occupied" and "unoccupied" status of the room. Specifically, it is the number of elements of the set $\{t|x_t > Q_{unoccupied}, x^i_{t+1} \leqslant Q_{unoccupied}, 1 \leqslant t \leqslant 24 \times 60/T\} \bigcup \{t|x_t \leqslant Q_{unoccupied}, x^i_{t+1} > Q_{unoccupied}, 1 \leqslant t \leqslant 24 \times 60/T\}$.

To quantify the performance of the proposed approach for building occupancy modeling, we define two criteria, i.e. normalized root mean squared error (NRMSE) and total variation distance (TVD). The NRMSE is used to compare the mean occupancy profile of the real data with that of the data generated by an occupancy model. Given two mean occupancy profiles, $a(t)$ and $b(t)$, $t = 1, 2, \ldots, K$, the NRMSD between $\mathbf{a}$ and $\mathbf{b}$ is defined as

$$NRMSD(\mathbf{a}, \mathbf{b}) = \frac{\|\mathbf{a} - \mathbf{b}\|/\sqrt{K}}{max(\mathbf{c}) - min(\mathbf{c})}, \tag{7}$$

where $\mathbf{a} = [a(1), a(2), \ldots, a(K)]^\intercal$, $\mathbf{b} = [b(1), b(2), \ldots, b(K)]^\intercal$, $\mathbf{c} = [\mathbf{a}^\intercal, \mathbf{b}^\intercal]^\intercal$, $_\intercal$ is the transpose operation, $\|\cdot\|$ is the Euclidean norm and K is the length of the mean occupancy profile.

In order to quantitatively evaluate the modeling performance for the four important random variables, i.e. time of first arrival, time of last departure, cumulative occupied duration and number of occupied/unoccupied transitions, the authors in [12,16] applied the criterion of Kullback–Leibler divergence (KLD) [25], which is defined as

$$KLD(\mathbf{p} \parallel \mathbf{q}) = \sum_t p_t \log\left(\frac{p_t}{q_t}\right), \tag{8}$$

where $\mathbf{p}$ and $\mathbf{q}$ are two probability density functions (pdfs). Since the variables have zero components and $p_k \log(\frac{p_k}{0}) = \infty$, the K-L divergence will fail to give a real value for comparison. Therefore, the authors assume $p_k \log(\frac{p_k}{0}) = 0$ which is not accurate for evaluation. Because of this issue, the authors in [17] suggest using TVD [26] which is adopted in this work. The TVD can be used to compare the distance between two random variables, and it can be formulated as

$$TVD(\mathbf{p} \parallel \mathbf{q}) = \frac{1}{2} \sum_t |p_t - q_t|, \tag{9}$$

where $|\cdot|$ is the absolute operation.

3.3. Experimental setup

During experiments, we have compared with two state-of-the-art approaches, i.e. agent-based model (ABM) [16] and inhomogeneous Markov chain (IMC) [12], which are two novel approaches for building occupancy modeling. The two models also used the same data for model construction. One hundred Monte Carlo simulations with a duration of one week for each are performed for all the models. Note that the resolution is 15 min resulting 96 points for one day. The default parameters are applied for the ABM and IMC models. For the proposed GAN approach, the number of hidden nodes is 240 and the learning rate is 8.0×10^{-5} for both the G and D networks. The selection of these parameters is based on the technique of grid search.

3.4. Experimental results and discussion

The mean occupancy profiles of the real data and the data generated by the ABM, the IMC, and the proposed GAN are shown in Fig. 4. It can be found that the ABM model performs the worst. This may be caused by the intuitively defined modules which lack the mathematical representation in the ABM model. The IMC model performs much better than the ABM model. However, since the IMC model applies the incremental information as the state, the modeling error may accumulate when transferring the incremental information into final occupancy. Note that the mean occupancy profile is for the entire week. We can clearly observe some accumulative errors on Friday afternoon and Saturday for the IMC model. Owing to the powerful modeling ability of the proposed GAN framework, it significantly outperforms the two state-of-the-art approaches for the variable of mean occupancy profile.

The pdfs of the four random variables, i.e. time of first arrival, time of last departure, cumulative occupied duration and number of occupied/unoccupied transitions, estimated from the ground truth data and the data generated by the ABM, the IMC, and the proposed GAN are illustrated in Fig. 5. Note that the authors in [12,16] only consider the working days when calculating the pdfs of the four random variables. However, according to Fig. 4, we can find that some occupants still attempt to go to the room at weekends. Thus, the control system of the room requires to be operated to provide a comfortable indoor environment. Therefore, in this work, we take into consideration all days including weekdays and weekends for the calculation of the pdfs of the four random variables. In an overall perspective, the ABM model has the worst performance, especially for the variables of time of first arrival and number of occupied/unoccupied transitions. The IMC model is able to perform much better than the ABM model. The proposed GAN approach can achieve a superior performance than the IMC model, especially for some detailed parts when following the pdfs of the ground truth data. Another observation is that the pdfs of the ground truth data seem to have larger variations such as the variable of cumulative occupied duration. This may be because the ground truth data only contains 70 days, while the data generated by the models includes 100 weeks, i.e. 700 days. Another interesting phenomenon in Fig. 5 is that the pdf of the number of occupied/unoccupied transitions contains large fluctuations, which can be caused by the including of the weekends. Since few occupants are present and the behaviours of occupants are more random in weekends, the transition between the two states, i.e. occupied and unoccupied, will be more random.

Table 1 quantifies the performance of the ABM, the IMC and the proposed GAN for occupancy modeling by using two criteria of NRMSD for mean occupancy profile and TVDs for the four random variables. The results are consistent with the observations in Fig. 4 and Fig. 5. The IMC model performs better than the ABM model, except for the variable of cumulative occupied duration. The proposed GAN outperforms the two state-of-the-art approaches. For example, the proposed GAN achieves improvements of 58.94% and 36.94% over the ABM and IMC on the variable of mean occupancy profile. We can conclude that the proposed GAN is efficient for building occupancy modeling.

3.5. Impact of the number of hidden nodes

The number of hidden nodes is one of the most important parameters for neural networks. Here, we attempt to investigate the impact of the number of hidden nodes on the performance of building occupancy modeling. For the sake of convenience, we set the number of hidden nodes to be the same for both the D and G networks. We investigate the number of hidden nodes changing from 80 to 400 with a step-size of 80. The NRMSE of mean occupancy profile and the TVDs of the four variables for the proposed GAN approach with a different number of hidden nodes are shown in Figs. 6 and 7, respectively. It can be found that the performance of the GAN approach is quite poor with a fewer number of hidden nodes, i.e. 80. This can be caused by the underfitting of the neural networks in the GAN with a limited number of hidden nodes. When the number of hidden nodes is large enough, i.e. from 160 to 400, the performance of the GAN is relatively stable. Specifically, the NRMSE of mean occupancy profile and the TVDs of time of first arrival, time of last departure and number of occupied/unoccupied transitions change slightly with the hidden nodes changing from 160 to 240. But the TVD of cumulative occupied duration decreases

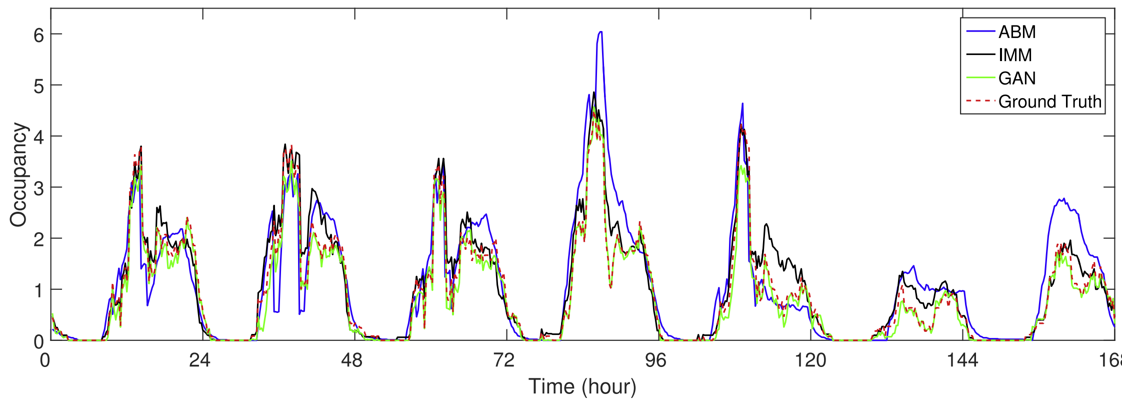


Fig. 4. Mean occupancy profiles estimated from the ground truth data and the data generated by the ABM, the IMC, and the proposed GAN.

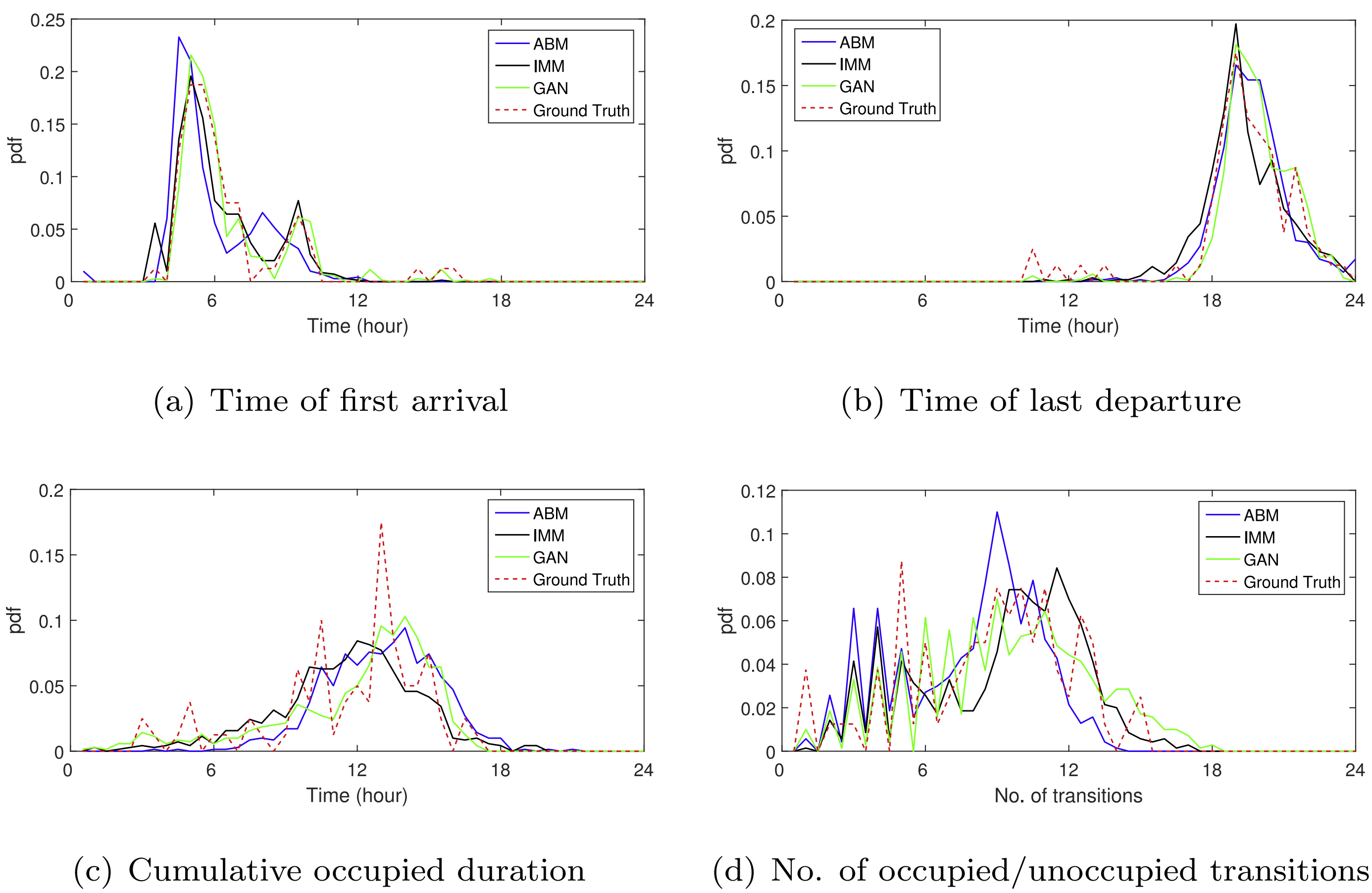


Fig. 5. The pdfs of the four random variables estimated from the ground truth data and the data generated by the ABM, the IMC, and the proposed GAN. The binsize is half an hour.

Table 1
NRMSD of the mean occupancy and TVDs of the four random variables for the ABM, the IMC and the proposed GAN.

Criteria	Variables	ABM	IMC	Proposed GAN
NRMSD	Mean occupancy	0.3468	0.2258	**0.1424**
TVD	Time of first arrival	0.3564	0.1614	**0.1346**
	Time of last departure	0.1746	0.1725	**0.1671**
	Cumulative occupied duration	0.3132	0.3211	**0.2807**
	No. of occupied/unoccupied transitions	0.2793	0.2614	**0.2282**

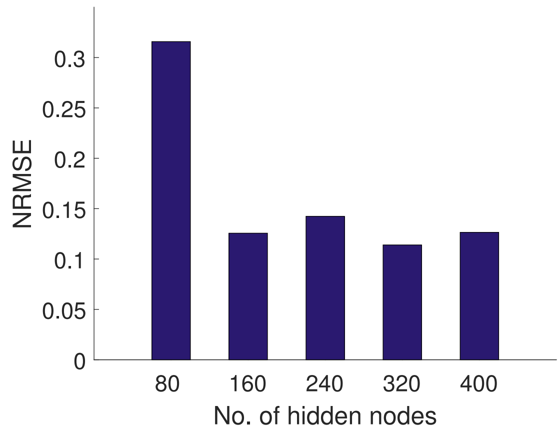
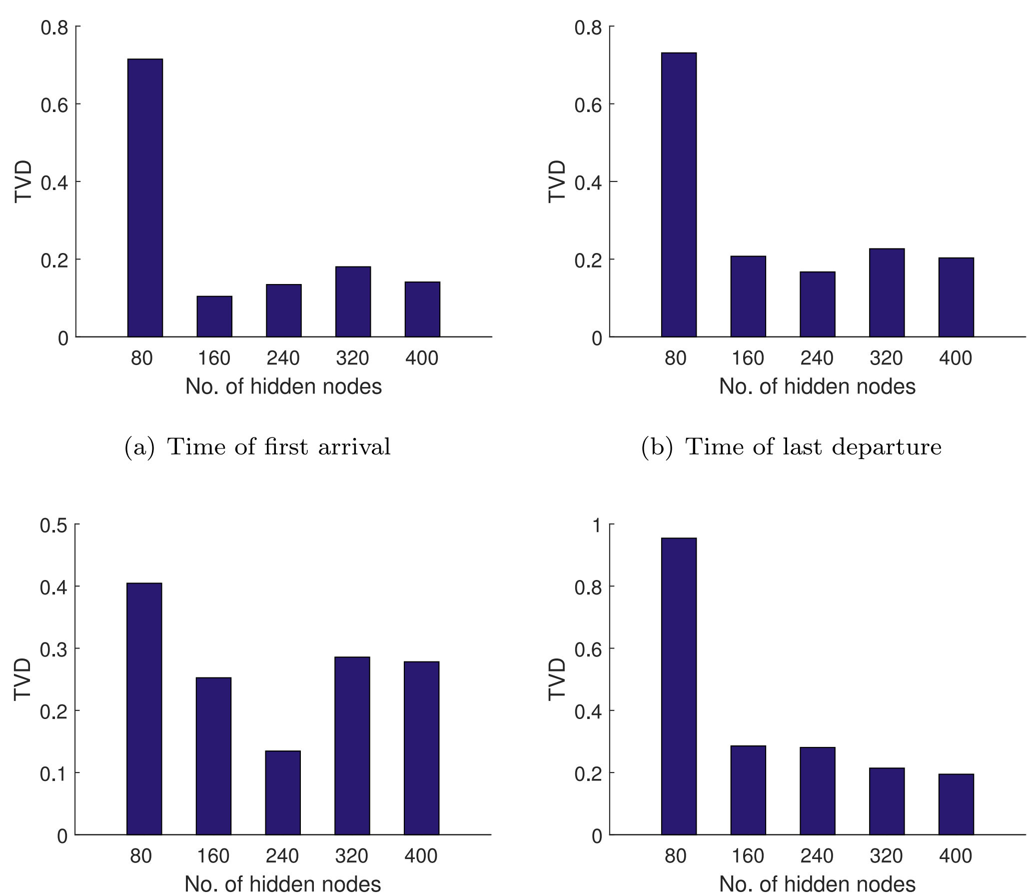


Fig. 6. The NRMSE of Mean occupancy profiles estimated from the data generated by the proposed GAN approach with different numbers of hidden nodes.

first and then increases when the number of hidden nodes is larger than 240. This indicates that the neural networks in GAN slightly overfit for this parameter with a high number of hidden nodes. We can conclude that in this case the neural networks in GAN are not very sensitive to the number of hidden nodes (within a certain range) when they are larger than a certain value. Theoretically, if the number of hidden nodes is too large, the GAN will tend to be overfitting, resulting poor performance. In general cases, we need to carefully tune the number of hidden nodes by using cross-validation technique. Specifically, we test different numbers of hidden nodes. Based on the results, a proper number of hidden nodes can be selected.

4. Conclusion

In this paper, we propose a Generative Adversarial Network (GAN) framework for building occupancy modeling. Since the behaviours of occupants in buildings are quite complex, the previously developed occupancy models attempt to simplify the problem with some specific assumptions, which limits the applicability of these models in real situations. To overcome these limitations, we leverage on a power GAN framework for building occupancy modeling without any specific assumptions. Two neural networks are designed for the generative (G) and discriminative (D) networks in GAN. Owing to the powerful generalization ability of the neural networks and the adversarial mechanism in GAN, a satisfactory occupancy modeling performance can be expected. Real experiments have been performed to verify the performance of the proposed GAN approach. We define five variables, i.e. mean occupancy, time of first arrival, time of last departure, cumulative occupied duration and number of occupied/unoccupied transitions, and two evaluation criteria, i.e. normalized root mean square error (NRMSE) and total variation distance (TVD), to quantify the performance of the proposed GAN framework for building occupancy modeling. Moreover, we compare the proposed GAN approach with two state-of-the-art methods, i.e. agent-based model (ABM) and inhomogeneous Markov chain (IMC), which applied the same data for model construction. Experimental results show that



(a) Time of first arrival
(b) Time of last departure
(c) Cumulative occupied duration
(d) No. of occupied/unoccupied transitions

Fig. 7. The TVDs of the four variables estimated from the data generated by the proposed GAN approach with different number of hidden nodes.

the proposed GAN approach achieves a superior performance over the two state-of-the-art methods on all the five variables under the two criteria. This indicates the effectiveness of the proposed GAN framework for building occupancy modeling. We also investigate the impact of one key hyperparameter, i.e. the number of hidden nodes, of the neural networks in GAN on modeling performance for the five variables. It turns out that the performance of the proposed GAN approach is relatively stable when the number of hidden nodes is larger than a certain value, and a careful selection of the number of hidden nodes will boost the performance of the GAN.

In our future works, we intend to investigate different neural network structures such as the popular deep recurrent neural network of long short-term memory [27] for the GAN approach in building occupancy modeling. Since the proposed GAN approach is able to achieve a good performance for occupancy modeling, a combination of the GAN approach with some occupancy estimation strategies may enhance the performance of building occupancy estimation due to the well consideration of occupancy properties.

References

[1] X. Feng, D. Yan, T. Hong, Simulation of occupancy in buildings, Energy Build. 87 (2015) 348–359.

[2] Agarwal, Y., Balaji, B., Gupta, R., Lyles, J., Wei, M., & Weng, T. (2010). Occupancy-driven energy management for smart building automation. ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building, (pp. 1–6).

[3] T. Leephakpreeda, Adaptive occupancy-based lighting control via grey prediction, Build. Environ. 40 (7) (2005) 881–886.

[4] R. Tomastik, Y. Lin, A. Banaszuk, Video-based Estimation of Building Occupancy During Emergency Egress, in: IEEE American Control Conference, 2008, pp. 894–901.

[5] Z. Chen, M.K. Masood, Y.C. Soh, A fusion framework for occupancy estimation in office buildings based on environmental sensor data, Energy Build. 133 (2016) 790–798.

[6] D.B. Crawley, L.K. Lawrie, F.C. Winkelmann, W.F. Buhl, Y.J. Huang, C.O. Pedersen, R.K. Strand, R.J. Liesen, D.E. Fisher, M.J. Witte, et al., Energyplus: creating a new-generation building energy simulation program, Energy Build. 33 (4) (2001) 319–331.

[7] D. Yan, J. Xia, W. Tang, F. Song, X. Zhang, Y. Jiang, DeST-an integrated building simulation toolkit part i: fundamentals, Build. Simul. 1 (2) (2008) 95–110.

[8] S.S. Kwok, R.K. Yuen, E.W. Lee, An intelligent approach to assessing the effect of building occupancy on building cooling load prediction, Build. Environ. 46 (8) (2011) 1681–1690.

[9] D. Wang, C.C. Federspiel, F. Rubinstein, Modeling occupancy in single person offices, Energy Build. 37 (2) (2005) 121–126.

[10] J. Page, D. Robinson, N. Morel, J.L. Scartezzini, A generalised stochastic model for the simulation of occupant presence, Energy Build. 40 (2) (2008) 83–98.

[11] I. Richardson, M. Thomson, D. Infield, A high-resolution domestic building occupancy model for energy demand simulations, Energy Build. 40 (8) (2008) 1560–1566.

[12] Z. Chen, J. Xu, Y.C. Soh, Modeling regular occupancy in commercial buildings using stochastic models, Energy Build. 103 (2015) 216–223.

[13] Z. Chen, C. Jiang, L. Xie, Building occupancy estimation and detection: a review, Energy Build. 169 (2018) 260–270.

[14] G. Gunathilak, A. Prasannakumar, N. Nazarian, H. Naeimi, A generalized event driven framework for building occupancy (WIP), Symp. Simul. Archit. Urban Des. (2013) 26.

[15] C. Wang, D. Yan, Y. Jiang, A novel approach for building occupancy simulation, Build. Simul. 4 (2) (2011) 149–167.

[16] C. Liao, Y. Lin, P. Barooah, Agent-based and graphical modelling of building occupancy, J. Build. Perform. Simul. 5 (1) (2012) 5–25.

[17] R. Jia, C. Spanos, Occupancy modelling in shared spaces of buildings: a queueing approach, J. Build. Perform. Simul. 10 (4) (2017) 406–421.

[18] J. Yang, M. Santamouris, S.E. Lee, Review of occupancy sensing systems and occupancy modeling methodologies for the application in institutional buildings, Energy Build. 121 (2016) 344–349.

[19] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, Adv. Neural Inf. Process. Syst. (2014) 2672–2680.

[20] N. Wang, W. Zha, J. Li, X. Gao, Back projection: an effective postprocessing method for GAN-based face sketch synthesis, Pattern Recognit. Lett. (2017), doi:10.1016/j.patrec.2017.06.012.

[21] L. Yu, W. Zhang, J. Wang, Y. Yu, Seqgan: Sequence generative adversarial nets with policy gradient, in: AAAI Conference on Artificial Intelligence, 2017, pp. 2852–2858.

[22] Zhang, L., Ji, Y., & Lin, X. (2017). Style transfer for anime sketches with enhanced residual u-net and auxiliary classifier GAN. ArXiv:1706.03319.

[23] V. Nair, G.E. Hinton, Rectified linear units improve restricted boltzmann machines, in: International Conference on Machine Learning, 2010, pp. 807–814.

[24] Kingma, D., & Ba, J. (2014). Adam: A method for stochastic optimization. ArXiv:1412.6980.

[25] T.M. Cover, J.A. Thomas, Elements of Information Theory, John Wiley & Sons, Hoboken, New Jersey, 2012.

[26] E. Wilmer, D.A. Levin, Y. Peres, Markov Chains and Mixing Times, American Mathematical Society, Providence, Rhode Island, 2017.

[27] Z. Meng, S. Watanabe, J.R. Hershey, H. Erdogan, Deep long short-term memory adaptive beamforming networks for multichannel robust speech recognition, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017, pp. 271–275.