The 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications
22-25 September, 2021, Cracow, Poland

Building Occupancy Classification from Indirect Sensing with Heterogeneous Datasets

Georgiana Creţu, Iulia Stamatescu, Grigore Stamatescu
Department of Automation and Industrial Informatics, University Politehnica of Bucharest
313 Splaiul Independentei, 060042 Bucharest, Romania
georgiana.cretu@stud.acs.pub.ro, {iulia.stamatescu, grigore.stamatescu}@upb.ro

***Abstract* — Accurate estimation of occupancy levels in residential and commercial buildings has become a key feature of advanced building automation systems. This allows the control system to adjust its setpoints to account for current and predicted occupancy in optimizing energy use while avoiding under-conditioning and over-conditioning of indoor spaces. We present the implementation and evaluation of a building occupancy classification system that can potentially improve energy management strategies in smart buildings through occupant-adaptive control. The system uses indirect sensing of ambient conditions such as temperature and humidity variations and carbon dioxide levels to provide a relative estimate of the occupancy ratio in the form of low, medium, high and zero occupancy. This serves a mean to preserve occupant privacy i.e. by not using cameras and image processing, and avoiding large hardware and installation costs through direct measurements with specialised occupancy sensors or people counters. Our system is tested on combinations of four different publicly available datasets with accuracy metrics ranging from 87% up to 100% in the most favourable cases.**

Keywords* — *occupancy classification, energy efficiency, smart buildings, convolutional neural networks, heterogeneous datasets

I. Introduction

In the developed world buildings consume almost 40% of the primary energy resources and the tendency is increasing given continuing urbanization. Buildings, building clusters and neighborhoods are also becoming key players in consumer-side energy management schemes for future smart grids. As the largest energy consumption factor, the Heating, Ventilation and Air Conditioning (HVAC) subsystem needs to balance personalised thermal comfort of the occupants as end-users with the energy consumption and costs on behalf of the building owner or operator. This balance can be achieved more effectively by knowing in real time and having the ability to predict future occupancy levels. Importance of building occupancy detection and accurate predictions is thus increasing in localized energy control of HVAC as well as for intelligent lighting as secondary energy saving source.

This work has been partially supported by the University "Politehnica" of Bucharest through the project „Engineer in Europe", ME no. 140/GP/19.04.2021.

Indoor occupancy [1] can be measured directly through cameras, people counters, access logs and user input, or estimated indirectly through existing ambient sensors for indoor conditions, from passive presence detectors as well as from activity patterns derived from appliance usage and electrical energy consumption. Indirect methods have the advantage that they require no dedicated system for detecting occupancy which saves hardware, installation and maintenance costs, while demanding large quality labeled datasets and computing resources for training robust models.

Given the increased data collection, storage and processing resources available at the building automation system (BAS) level, computational intelligence algorithms can be apply to the large amounts of collected data [2]. This results in data-driven occupancy models as well as in reliable forecasts that can incorporate contextual information such as seasonality factors, work schedules, dominant space usage. Conventional models based on machine learning techniques as well as new deep learning architectures are currently being deployed and compared.

Main contributions of the paper are argued to be as follows:

- Description of a methodology based on computational intelligence for building occupancy estimation using indirect and ambient sensor readings, with application to building energy management;
- Implementation and evaluation of a convolutional neural network classifier using data from five publicly available heterogeneous datasets.

The rest of the paper is structured as follows. Section II presents a detailed review of current approaches in the field of indirect occupancy sensing and estimation using intelligent techniques. Section III describes in depth our methodology including the description of the datasets used for training and data preprocessing. The results in Section IV highlight implementation aspects and the main accuracy metrics for testing our approach. Section V concludes the paper with the main lessons learned and potential for deployment on a dedicated embedded system for online inference.

II. Related Work

Multiple applications for indoor occupancy detection rely on Passive Infrared (PIR) detectors which are mainly used in security and automated lighting control systems [3]. One important drawback is that such sensors cannot detect still persons. Also detection can be unreliable over short time spans which introduces delays in the processing and update of the occupancy, especially in the case of large occupancy variations over short periods [4]. An alternative system is described by [5] where an 8x8 thermopile sensor matrix is used to detect heat fingerprints of building users, this offers better accuracy and robustness to noise while preserving privacy. However the requirement for additional hardware and calibration of the system for particular rooms or buildings remains [6].

An improved alternative consists of data collection from ambient sensors such as: temperature, humidity, air quality/gas, carbon dioxide, light, barometric pressure, to infer occupancy. For this type of data many studies have been carried out which consist of the application of computational intelligence and machine learning algorithms [7] such as: Hidden Markov Models (HMM), Support Vector Machines (SVM), Extreme Learning Machines (ELM), Linear Discriminant Analysis (LDA), Classification and Regression Trees (CART), Random Forests (RF) and Artificial Neural Networks (ANN) [8]. In one application of HMM [3], the reported average model accuracy is around 73%, obtained after an extensive testing on a real building with a multi-node sensor network and by using cameras to collect the ground truth information needed to label the training examples. Superior accuracy values, between $95\% - 99\%$ are achieved using LDA, CART and RF in [7] based on sensor readings from light, humidity and carbon dioxide sensors.

An alternative approach is presented by [9] where the network infrastructure, MAC and IP addresses of the PC users are monitored alongside the information regarding keyboard and mouse usage. The reported accuracy in estimating occupancy from this data is 80% at the whole building level, with the main limitation being that the applicability is limited only to occupants that use a PC to carry out their work related tasks.

All the studies listed so far leverage algorithms that handle individual training examples for learning data-driven models of occupancy such as Bayesian networks in [10] and [11]. Newer neural network structures such as recurrent neural networks (RNN) and long short-term memory networks (LSTM) have the ability to operate on input sequences of varying sizes e.g. the carbon dioxide variation over the last five minutes, thereby learning fine grained dependencies in the input sensor data. This compares favourably to averaged values used for discrete valued datasets. Another feature that we focus on exploiting is incorporating data from different heterogenous data sources in a single approach which should lead to improved robustness of the classifier.

A summary of related work with the main identified approaches is presented in Table I.

III. Methodology

The main goal of our application is to output the occupancy level in a building as a discrete class label based on various indirect measurement from ambient sensors and contextual information. The developed algorithm provides four probability scores, associated to four output classes: empty (E), low (L), medium (M), and high (H). Main stages in the development include the following:

- Collecting raw sensor reading from sensors, such as: ambient temperature, light, carbon dioxide detectors. In the current study we rely on previously documented and provided heterogeneous datasets from [15] and [7];
- Data preprocessing and reshaping into adequate input sequences for the neural network input layer compatibility;
- Dividing the processed dataset into training, validation and testing segments;
- Neural network training and output of the probability values associated to each occupancy class/category.

A. Datasets Descriptions

Raw data is provided as a list of vectors, with each vector containing multiple sensor readings and the occupancy level at the end as training and testing label as: [sensor1, sensor2, sensor3, occupancy] or [sensor1, sensor2, occupancy], depending on the number of sensors used. The datasets that we use are described next:

- Dataset collected in the living room of a house (*home*), which includes readings from three ambient sensors: temperature, humidity, pressure [15];
- Dataset collected in a fitness hall (*gym*) for seven non-consecutive days. It provides readings from temperature, humidity and pressure sensors [15];
- Dataset collected in an office (*datatest*) for three consecutive days. It provides readings from temperature, humidity, light and carbon dioxide sensors [7];
- Dataset collected in an office (*datatest2*) for eight consecutive days. It provides readings from temperature, humidity, light and carbon dioxide sensors [7].

Data is reported every second for *home* and *gym* datasets and every minute for *datatest* and *datatest2*. For the latter two datasets, combinations of two and three sensors were used from the available readings. One limitation is that these datasets only report binary occupancy values, that is if the room is occupied or not, which in our case and in order to bridge this difference is rescaled as empty and medium occupancy. Finally, we additionally define a fifth dataset through the combination of the home and gym datasets as global model and run our sequence model on this dataset as well.

Table I. SUMMARY OF REFERENCE APPROACHES FOR OCCUPANCY ESTIMATION

Algorithm Type	Sensors Used	Accuracy	Reference
RBF Neural Network	Light, sound, Reed switches, CO_2, Temperature, PIR	$63.23\% - 66.43\%$	[12]
Support Vector Machines, K-Nearest Neighbors (KNN), Thresholding	Electrical energy usage	$59\% - 90\%$	[13]
Artificial Neural Networks	Temperature, PIR, CO_2, sound, computer temperature, relative humidity (RH)	$70.4\% - 72.3\%$	[14]
RF, CART, LDA	Temperature, humidity, light, CO_2, relative humidity	$95.5\% - 98.7\%$	[7]

B. Data Processing

The first step in processing the data is assembling the measurement sequences that are used by the neural network input layer. For this, 10 second steps have been chosen for the *home* and *gym* datasets and 60 seconds steps have been chosen for the *datatest* and *datatest2* sets. For the sequence size a value of 50 has been experimentally chosen as an adequate parameter, resulting in overlapping input examples for each step. In the case of shorter sequences, the padding technique is used to complete sequences which are partial given larger distances between consecutive measurements. For the minimum sequence size, this is computed as 0.75 of the maximum sequence size. When assigning the labels of the individual examples to the newly formed measurement sequences we consider either the dominant occupancy value for all examples in the sequence through majority voting or a suitable value from the end of the sequence.

A second step in the data processing for the training procedure consists of balancing the datasets in order to mitigate the bias effect of the classifier when being exposed to dominant class examples in the training data. A balancing function has been implemented which computes the minimum value among the number of examples in each class label and reduces the dominant class size in a ration of $4 - 8$ times the size of the smallest class.

A summary of the properties of the datasets after processing is listed in Table 2.

C. Network Architecture

We use a convolutional neural network (CNN) architecture to estimate the class probabilities based on the labeled dataset presented in the previous subsection. CNN has been originally developed for 2D image data but has been shown recently to work well on 1D time series data such as scalar sensor readings or energy measurements [16] with very good performance. The architecture includes the following layers: batch normalization layer, two Conv2D convolutional layers, two MaxPool2D maximum pooling layers with batch normalization after each convolutional layer, two fully connected layers with 32 and 4 neurons, where the last output layer computes the class probabilities associated to each occupancy category. The network architecture is illustrated in Figure 1.

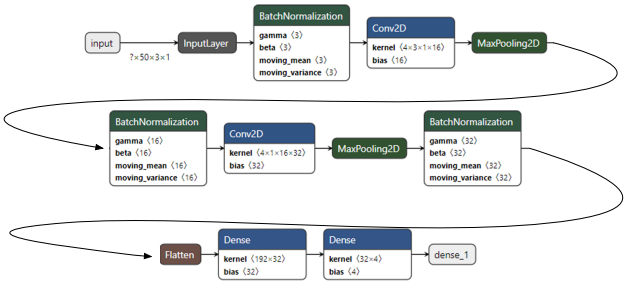

Figure 1. CNN Architecture for Occupancy Classification

The role of batch normalization layer is to translate the individual sensor readings on a comparable scale as to avoid the algorithm to overweight higher absolute values e.g. pressure is reported into thousands of milibars while the temperature value is reported as tens of degrees Celsius. The convolution product on a two dimensional table of values is computed as:

$$G(x,y) = \omega * F(x,y) = \sum_{\delta x=-k_i}^{k_i} \sum_{\delta y=-k_j}^{k_j} \omega(\delta x, \delta y) \cdot F(x+\delta x, y+\delta y) \quad (1)$$

where ω is the convolution kernel. The max pooling layer computes the maximum value within each resulting patch of the feature map.

To speed up the network training the *batch_size* parameter is set at 64. Adam optimization is used for computationally efficient minimization of the loss function during the training procedure. The learning rate is set progressively lower with the values $[0.01, 0.001, 0.0001]$ to improve convergence with various epoch numbers between $[30, 20, 10]$. To compare the network output over various datasets and parameter sets a dedicated function has been created that creates a tabular file including the real occupancy, predicted occupancy and the timestamp of each sequence in the dataset alongside the overall accuracy of the trained model.

Table II. DATASET PROPRETIES

Dataset	No. Examples	Timestep [s]	Sequences Pre-Balancing	Sequences Post-Balancing	Sequences per Class
home	295823	10	29015	10843	E:3091, L:3782, M:3340, H:630
gym	10129	10	586	586	E:0, L:150, M:294, H:142
datatest (T,H,CO_2)	2665	60	2614	2614	E:1678, L:0, M:936, H:0
datatest2 (T,H,lux)	9752	60	9701	9701	E:7701, L:0, M:2000, H:0
datatest2 (T,H,CO_2)	9752	60	9701	9701	E:7701, L:0, M:2000, H:0
home + gym	305952	10	29605	12855	E:3889, L:4471, M:3723, H:772

IV. RESULTS

For the practical evaluation, the algorithms are implemented using the Python programming language with the *Tensorflow* - as higher level framework for specifying neural network architectures and training, *os* - useful functions for operating system interactions, *json* and *csv* - for structured data parsing and reformatting, libraries. Visualization of network architecture is performed using *Netron* and *Tensorboard* tools. Network training is carried out by randomly splitting the input datasets into training (60%), validation (20%) and testing (20%) subsets. A random seed parameter is defined in order to control for the train-validation-test split for all the experiments. For the loss function the categorical crossentropy is typically used in such problems and is expressed as:

$$Loss = -\sum_{i=1}^{outputs} y_i \cdot log(\hat{y}_i) \tag{2}$$

where $\hat{y}_i$ is the i-th value of the array of potential classes and y_i is the respective target value. Multi-class accuracy is computed as the number of the total correct classifications in each of the bins, over the total number of classifications, correct and incorrect. Figures 2-4 present the evolution of the accuracy over the multiple training and validation epochs for the *home*, *datatest2* and *home + gym* datasets.

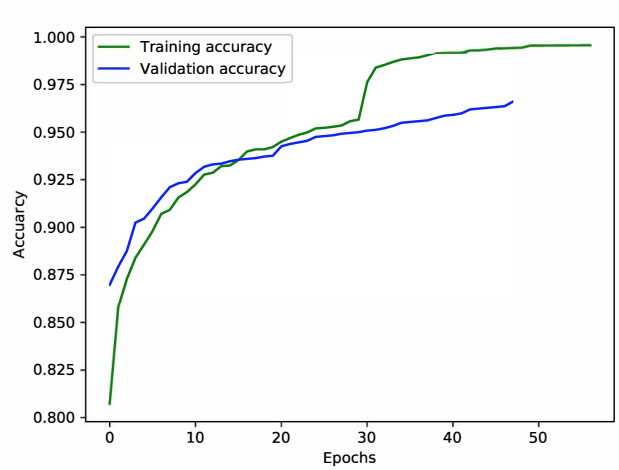

Figure 2. Training and Validation Accuracy: home Dataset

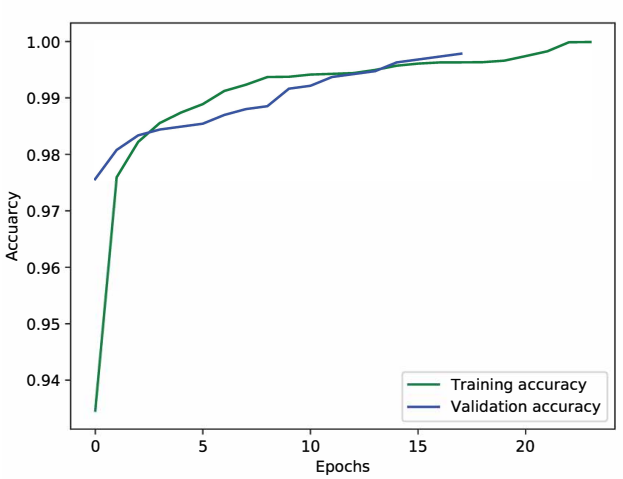

Figure 3. Training and Validation Accuracy: datatest2 Dataset

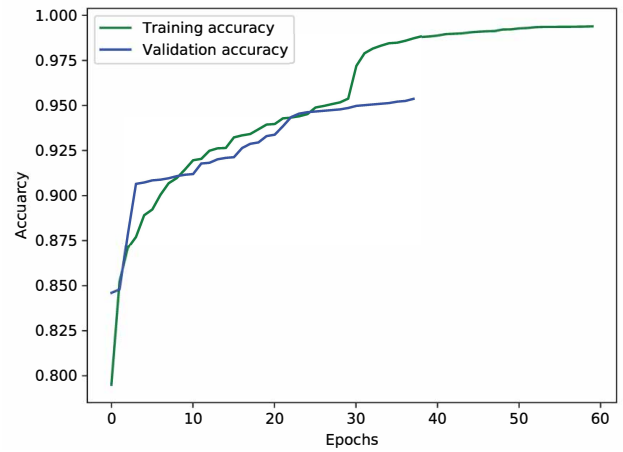

Figure 4. Training and Validation Accuracy: home + gym Dataset

We report next the accuracy results when the model is used for classification on the testing subsets of each dataset. One observation is that the decrease in the reported testing accuracy is limited when using two instead of three sensors inputs in the model. Testing on both individual and combined datasets yields a more robust classification model as is the case with the *home*, *gym* and *home + gym* datasets. The loss function implemented is

the sparse variant for computational efficiency.

Table III. CLASSIFICATION TESTING RESULTS

Dataset	No. Sensors	Accuracy [%]	Loss
home	3	0.9	0.5
gym	3	1	0
home + gym	3	0.93	0.5
datatest	3	0.99	0.01
datatest2	3	0.99	0.02
home	2	0.87	1.16
home + gym	2	0.87	1.33
Datatest	2	0.99	0.01
Datatest2	2	0.99	0.05

An example for the obtained multi-class classification confusion matrix, for the *home* dataset with three inputs/sensors, is listed in Figure 5. The class ids 1-4 are associated to the empty, low, medium and high categories respectively. The associated accuracy metrics are computed row- and column- wise as well as globally.

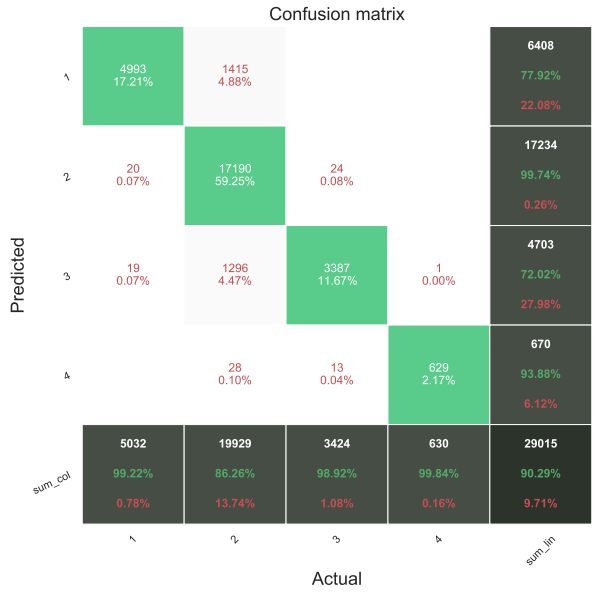

Figure 5. Confusion Matrix for *home* Dataset with Three Inputs: Accuracy = 0.9

The disk size of the trained model is 39.6 kB for the model with real-typed float weights and 15.4 kB for a quantized model with integer weights for future portability on efficient low-power embedded platforms.

V. CONCLUSION

We have presented an approach to classify building occupancy based on indirect readings from ambient sensors. A Convolutional Neural Network (CNN) has been designed and evaluated on publicly available heterogeneous datasets yielding an approach with increased robustness to variations in the input data. The sequence modelling technique offers the ability to quantify the dependency between the variations of the ambient parameters such as temperature, humidity, light and carbon dioxide on the reported occupancy levels. For future development, we aim to validate the results through a small scale laboratory data collection which allows more control over the generative data process in real conditions. Porting a compressed version of the algorithm to embedded development boards, such as Raspberry Pi class devices, for online inference is also foreseen.

REFERENCES

[1] D. Trivedi and V. Badarla, "Occupancy detection systems for indoor environments: A survey of approaches and methods," *Indoor and Built Environment*, vol. 29, no. 8, pp. 1053–1069, 2020.

[2] G. Stamatescu, I. Stamatescu, N. Arghira, and I. Fagarasan, "Data-driven modelling of smart building ventilation subsystem," *Journal of Sensors*, vol. 2019, 2019.

[3] 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)*, 2013, pp. 114–119.

[4] H. Elkhoukhi, Y. NaitMalek, A. Berouine, M. Bakhouya, D. Elouadghiri, and M. Essaaidi, "Towards a real-time occupancy detection approach for smart buildings," *Procedia computer science*, vol. 134, pp. 114–120, 2018.

[5] C. Chiţu, G. Stamatescu, I. Stamatescu, and V. Sgârciu, "Wireless system for occupancy modelling and prediction in smart buildings," in *2017 25th Mediterranean Conference on Control and Automation (MED)*, 2017, pp. 1094–1099.

[6] C. Wang, J. Jiang, T. Roth, C. Nguyen, Y. Liu, and H. Lee, "Integrated sensor data processing for occupancy detection in residential buildings," *Energy and Buildings*, vol. 237, 2021.

[7] L. M. Candanedo and V. Feldheim, "Accurate occupancy detection of an office room from light, temperature, humidity and co2 measurements using statistical learning models," *Energy and Buildings*, vol. 112, pp. 28–39, 2016.

[8] Y. Peng, A. Rysanek, Z. Nagy, and A. Schlüter, "Using machine learning techniques for occupancy-prediction-based cooling control in office buildings," *Applied energy*, vol. 211, 2018.

[9] R. Melfi, B. Rosenblum, B. Nordman, and K. Christensen, "Measuring building occupancy using existing network infrastructure," in *2011 International Green Computing Conference and Workshops*, 2011, pp. 1–8.

[10] M. Amayri, S. Ploix, H. Kazmi, Q.-D. Ngo, and E. Safadi, "Estimating occupancy from measurements and knowledge using the bayesian network for energy management," *Journal of Sensors*, vol. 2019, 2019.

[11] M. Amayri, Q.-D. Ngo, E. A. E. Safadi, and S. Ploix, "Bayesian network and hidden markov model for estimating occupancy from measurements and knowledge," in *9th IEEE Intl Conf Intelligent Data Acquisition and Advanced Computing Systems*, '17.

[12] Z. Yang, N. Li, B. Becerik-Gerber, and M. Orosz, "A multi-sensor based occupancy estimation model for supporting demand driven hvac operations," in *Proc. of the Symposium on Simulation for Architecture and Urban Design*, San Diego, 2012.

[13] W. Kleiminger, C. Beckel, T. Staake, and S. Santini, "Occupancy detection from electricity consumption data," in *Proc. of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, ser. BuildSys'13, New York, NY, USA, 2013, p. 1–8.

[14] T. Ekwevugbe, N. Brown, and V. Pakka, "Real-time building occupancy sensing for supporting demand driven hvac operations," *Proc. of 13th Intl Conf for Enhanced Building Operations*, 2013.

[15] A. Vela, J. Alvarado-Uribe, M. Davila, N. Hernandez-Gress, and H. G. Ceballos, "Estimating occupancy levels in enclosed spaces using environmental variables: A fitness gym and living room as evaluation scenarios," *Sensors*, vol. 20, no. 22, p. 6579, 2020.

[16] A. Tudose, D. Sidea, I. Picioroaga, V. Boicea, and C. Bulac, "A cnn based model for short-term load forecasting: A real case study on the romanian power system," in *55th International Universities Power Engineering Conference (UPEC)*, 2020.