Indoor occupancy estimation using environmental parameters

Masood, Khalid Mustafa

2017

Masood, K. M. (2017). Indoor occupancy estimation using environmental parameters.
Doctoral thesis, Nanyang Technological University, Singapore.

<http://hdl.handle.net/10356/72520>

<https://doi.org/10.32657/10356/72520>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0
International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 11:52:50 SGT

Indoor Occupancy Estimation Using Environmental Parameters

MUSTAFA KHALID MASOOD

School of Electrical and Electronic
Engineering

A thesis submitted to the Nanyang
Technological University, Singapore in partial
fulfilment of the requirement for the degree of
Doctor of Philosophy

2017

Abstract

In this work, we address the problem of obtaining reliable estimates of the real-time occupancy for Air-Conditioning and Mechanical Ventilation (ACMV) systems. We take a non-intrusive approach that is gaining popularity, which is the use of environmental parameters such as CO_2 , temperature, humidity and pressure. The approach requires the extraction and selection of useful features from the sensor data, which can be used with machine learning techniques to obtain occupancy estimates. To date, occupancy estimation has primarily focused on *static* features, which we define to be features sampled instantaneously, as opposed to *dynamic* features that are sampled over a moving window. In this work, we propose novel feature selection techniques that employ both static and dynamic features. We study the performance of these algorithms with experimental data collected through a multi-sensory network in an office space.

First, we propose novel feature selection algorithms within the static feature paradigm. In past works, feature selection has generally been implemented using filter-based approaches. In this work, we introduce the use of wrapper and hybrid feature selection for occupancy estimation. Compared to filter methods, our approach can achieve a better occupancy estimation accuracy. Additionally, we use a ranking-based combinatorial search in our algorithms, which is more efficient than the exhaustive search used in past works. For wrapper feature selection, we propose the WRANK-ELM, which searches an ordered list of features using the Extreme Learning Machine (ELM) classifier. For hybrid feature selection, we propose the RIG-ELM, which is a filter-wrapper hybrid that uses the Relative Information Gain (RIG) criterion for feature ranking and the ELM for a combinatorial search. Experimental results for different sensor positions and different time resolutions show that the proposed algorithms outperform past work in terms of accuracy and computation time. However, the performance is not so smooth for the complex occupancy profile at a 1-minute time resolution.

Dynamic features are more informative than static features, but their use in occupancy estimation has to date been very limited. In this work, we introduce the Feature-Scaled ELM (FS-ELM), which is an ELM-based estimator that uses dynamic features. The FS-ELM is in fact a novel architecture of the ELM, in which a feature layer is added to the standard ELM. The feature layer extracts dynamic features from the raw data. Additionally, we present an effective smoothing strategy to address the problem of noise in environmental parameter data. We demonstrate through experimental results that the FS-ELM yields excellent occupancy estimation accuracy, while retaining the computational efficiency of the standard ELM.

Also in this work, we propose a generalized feature selection framework for constructing an occupancy estimator for dynamic features. The framework is a kind of filter-wrapper hybrid feature selection method, which is novel in that it uses a combination of static and dynamic features. In the framework, the filter component works with static features, while the wrapper component works with dynamic features. We use the static features for purposes of speed, since filter methods of feature selection (which work with static features) are quite fast. Thus, the overall computation time of the framework is kept low, while ensuring good accuracy of estimation due to the use of dynamic features. The framework thus offers a reliable method of evaluating a large set of features. To perform occupancy estimation within the proposed framework, we present a novel technique called the Hybrid Feature-Scaled Extreme Learning Machine (HFS-ELM). The HFS-ELM is a dynamic model of the occupancy level in which the present occupancy depends on the measurements of multiple environmental parameters and the estimated occupancy level in a past time horizon. It is thus a generalized version of the FS-ELM, which works with only one type of environmental parameter. Also, while the FS-ELM is limited to two types of arbitrarily selected features, the HFS-ELM accommodates multiple feature types. Through experimental results, we show the excellent performance of the HFS-ELM even for complex occupancy profiles at a 1-minute time resolution.

Acknowledgments

First and foremost, I thank the Almighty for all the blessings He has conferred on me. I attribute this work entirely to His Grace and Mercy.

I owe a deep debt of gratitude to my supervisor, Prof. Yeng Chai Soh, for the patience, foresight and wisdom with which he guided me throughout this work. Aside from our insightful discussions on my work, just observing his thought process has been an invaluable source of learning for me as a researcher. His kind support and advice, in both achievements and setbacks, will always remain an inspiration for me.

I gratefully acknowledge the funding provided by Singapore's National Research Foundation under the CRP scheme.

I am very thankful to my dear colleagues, Chaoyang Jiang and Chen Zhenghua, whose cheerful company, brilliant ideas and constant encouragement have greatly enriched my PhD experience. I am also thankful to my dear friends in the IoT lab, Xu Jinming, Zhou Hongming and Zhu Qingchang, whose well wishing and support means a lot to me.

I am indebted to Prof Xie Lihua for his kind and valuable suggestions on my work. I am also very grateful to Dr Victor Chang, collaborating with whom was a joy and a great learning experience.

I especially thank my parents and siblings, who have been loving, patient, compassionate and supportive beyond words. Finally, I am an incredibly fortunate man to have my wife, whose love and support is at the heart of this work.

Table of Contents

Table of Contents	i
List of Figures	viii
List of Tables	x
Abbreviations	xi
1 Introduction	1
1.1 Scope and Overview	1
1.2 Major Contributions	3
1.3 Outline of the thesis	4
2 Literature Review	6
2.1 Introduction	6
2.2 Sensors for occupancy estimation	6
2.2.1 Binary sensors	6
2.2.2 Wearable sensors	8
2.2.3 Device and appliance monitoring	9
2.2.4 Camera-based sensors	9
2.2.5 Environmental sensors	11
2.2.6 Applications of sensor-based occupancy detection . .	15
2.3 Feature selection methods	15
2.3.1 Filter methods	15
2.3.2 Wrapper methods	16

2.3.3	Hybrid methods	17
2.4	Occupancy estimation using ELM	18
2.5	Occupancy estimation using dynamic models	19
I	Occupancy estimation using static features	21
3	Occupancy estimation using wrapper and hybrid feature selection	23
3.1	Introduction	23
3.2	Background	23
3.2.1	Relative Information Gain (RIG)	23
3.2.2	Extreme Learning Machine (ELM)	24
3.3	Data collection	27
3.3.1	Environmental sensors	27
3.3.2	Ground truth data	28
3.4	Feature selection	31
3.4.1	Feature extraction	31
3.4.2	Wrapper-based ranking using ELM (WRANK-ELM)	32
3.4.3	Filter-wrapper hybrid algorithm (RIG-ELM)	33
3.5	Analysis of selected features	34
3.6	Summary	35
4	Results and Discussion: Part I	38
4.1	Introduction	38
4.2	Occupancy level estimation	38
4.2.1	Individual sensor locations	38
4.2.2	Sensor combinations	41
4.3	Occupancy number estimation	42
4.3.1	15-minute resolution	42
4.3.2	1-minute resolution	44
4.4	Summary	45

II	Occupancy estimation using dynamic features	61
5	Occupancy estimation using CO_2-based dynamic features: FS- ELM	63
5.1	Introduction	63
5.2	Indoor Occupancy estimation from CO_2 data	63
5.3	FS-ELM for occupancy estimator identification	64
5.3.1	The standard ELM for occupancy estimation	64
5.3.2	The feature layer of the FS-ELM	66
5.3.3	The hidden layer of the FS-ELM	68
5.3.4	FS-ELM training	68
5.4	Smoothed CO_2 data for occupancy estimation	71
5.4.1	Globally smoothed CO_2 data	71
5.4.2	Locally smoothed CO_2 data	72
5.5	Summary	75
6	Feature selection with the Hybrid Feature-Scaled ELM (HFS-ELM)	76
6.1	Introduction	76
6.2	Hybrid Feature Selection Framework	76
6.2.1	The filter component of the framework	77
6.2.1.1	Extraction of static features	77
6.2.1.2	RIG-based feature selection	77
6.2.2	The wrapper component of the framework: Hybrid Feature Scaled Extreme Learning Machine (HFS-ELM)	79
6.3	Analysis of selected features	84
6.4	Summary	85
7	Results and Discussion: Part II	86
7.1	Introduction	86
7.2	Parameter settings	86

7.3	Occupancy level estimation	87
7.3.1	Individual sensor positions	87
7.3.2	Sensor combinations	89
7.4	Occupancy number estimation	91
7.4.1	15-minute resolution	91
7.4.2	1-minute resolution	92
7.5	Summary	94
8	Conclusion and Future works	101
8.1	Discussions and Summary	101
8.2	Extensions and Future Work	103

List of Figures

3.1	Structure of the ELM	26
3.2	(a) Rotronic CL11 sensor for CO_2 , relative humidity and temperature and (b) Lutron MHB-382SD sensor for pressure levels	28
3.3	Process Instrumentation Lab, NTU. (a) Office space and storage space of the venue (b) One set of sensors installed in the office space (c) Overhead cameras to record occupancy ground truth	29
3.4	Layout of sensors in Process Instrumentation Lab, NTU . .	30
3.5	WRANK-ELM algorithm	33
3.6	RIG-ELM algorithm	34
4.1	Occupancy estimation results for the five testing days for sensor position A	43
4.2	Occupancy estimation results for the five testing days for sensor position B	44
4.3	Occupancy estimation results for the five testing days for sensor position C	45
4.4	Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (for sensor position A)	46
4.5	Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (for sensor position B)	47
4.6	Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (for sensor position C)	48

4.7	Comparison of temperature data for Positions A, B and C for test day 4	49
4.8	Occupancy estimation results for the five testing days for sensor positions AB	49
4.9	Occupancy estimation results for the five testing days for sensor positions AC	50
4.10	Occupancy estimation results for the five testing days for sensor positions BC	50
4.11	Occupancy estimation results for the five testing days for sensor positions ABC	51
4.12	Estimation accuracies for different sensor combinations	51
4.13	RMSE for different sensor combinations	52
4.14	Presence/absence detection accuracies for different sensor combinations	52
4.15	Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (AB)	53
4.16	Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (BC)	54
4.17	Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (AC)	55
4.18	Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (ABC)	56
4.19	Occupancy estimation results for full-range occupancy for sensor position A	57
4.20	The x-tolerance accuracies for sensor position A	57
4.21	Occupancy estimation results for full-range occupancy for sensor position B	58
4.22	The x-tolerance accuracies for sensor position B	58
4.23	Occupancy estimation results for full-range occupancy for sensor position C	59
4.24	The x-tolerance accuracies for sensor position C	59

4.25	Occupancy estimation results for the five testing days for sensor positions AB	60
4.26	The x-tolerance accuracies for occupancy number estimation with $R = 1$ (at sensor position A)	60
5.1	The structure of the standard extreme learning machine (ELM).	65
5.2	The sigmoid function.	65
5.3	The structure of the feature scaled extreme learning machine (FS-ELM).	67
5.4	One day CO_2 concentration of the test-bed	73
6.1	Hybrid feature selection framework for dynamic occupancy estimation	78
6.2	Structure of the HFS-ELM	81
7.1	Occupancy estimation results for the five testing days for sensor position A	89
7.2	Occupancy estimation results for the five testing days for sensor position B	90
7.3	Occupancy estimation results for the five testing days for sensor position C	91
7.4	Occupancy estimation results for the five testing days for AB	92
7.5	Occupancy estimation results for the five testing days for sensor position BC	93
7.6	Occupancy estimation results for the five testing days for sensor position AC	94
7.7	Occupancy estimation results for the five testing days for sensor position ABC	95
7.8	Occupancy number estimation results for full-range occupancy for $R=15$ (at sensor position A)	96
7.9	The x-tolerance accuracies for occupancy number estimation for $R=15$ (at sensor position A)	96
7.10	Occupancy number estimation results for full-range occupancy for $R=15$ (at sensor position B)	97

7.11	The x-tolerance accuracies for occupancy number estimation for R=15 (at sensor position <i>B</i>)	97
7.12	Occupancy number estimation results for full-range occupancy for R=15 (at sensor position <i>C</i>)	98
7.13	The x-tolerance accuracies for occupancy number estimation for R=15 (at sensor position <i>C</i>)	98
7.14	Occupancy number estimation for full-range occupancy for R=1 for FS-ELM without smoothing, with global smoothing and with local smoothing	99
7.15	The x-tolerance accuracies for occupancy number estimation for R=1 for FS-ELM without smoothing, with global smooth- ing and with local smoothing	99
7.16	Occupancy number estimation results for full-range occupancy for R=1	100
7.17	The x-tolerance accuracies for occupancy number estimation for R=1 (at sensor position <i>A</i>)	100

List of Tables

3.1	Specifications of the environmental sensors	28
3.2	Description of occupancy levels	31
3.3	Features extracted from sensor data	31
3.4	Features selected by the WRANK-ELM for <i>Position A</i> . . .	35
3.5	Features selected by the WRANK-ELM for <i>Position B</i> . . .	35
3.6	Features selected by the WRANK-ELM for <i>Position C</i> . . .	36
3.7	Features selected by the RIG-ELM for <i>Position A</i>	36
3.8	Features selected by the RIG-ELM for <i>Position B</i>	36
3.9	Features selected by the RIG-ELM for <i>Position C</i>	36
4.1	Occupancy level estimation accuracy and RMSE values for the three algorithms	41
4.2	Computation times for WRANK-ELM, RIG-ELM and Zhang et al's method	41
4.3	Occupancy level estimation accuracy and RMSE values for sensor combinations <i>AC</i> and <i>BC</i>	42
4.4	Occupancy level estimation accuracy and RMSE values for sensor combinations <i>BC</i> and <i>ABC</i>	42
4.5	Presence/absence detection accuracies for WRANK-ELM, RIG- ELM and Zhang et al's method	43
4.6	Presence/absence detection accuracies for sensor positions <i>A</i> , <i>B</i> and <i>C</i>	44
4.7	Comparison of 3 and 4-tolerance accuracies (at sensor position <i>A</i>)	45

6.1	Definitions of static features	77
6.2	Definitions of dynamic features	80
6.3	Feature combinations selected by the hybrid framework . . .	85
7.1	Parameter settings for HFS-ELM	87
7.2	Parameter settings for FS-ELM	87
7.3	Occupancy level estimation accuracy and RMSE values for HFS-ELM, FS-ELM, Zhang et al.'s method and the standard ELM	89
7.4	Presence/absence detection accuracies for HFS-ELM, FS-ELM, Zhang et al.'s method and the standard ELM	90
7.5	Occupancy level estimation accuracy and RMSE values for HFS-ELM, FS-ELM, Zhang et al.'s method and the standard ELM	91
7.6	Occupancy level estimation accuracy and RMSE values for HFS-ELM, FS-ELM, Zhang et al.'s method and the standard ELM	92
7.7	Presence/absence detection accuracies for HFS-ELM, FS-ELM, Zhang et al.'s method and the standard ELM for sensor com- binations	93
7.8	Comparison of accuracies (tolerance 3 and 4) for 15 minutes resolution and 1 minute resolution (at sensor position A) . .	94

Abbreviations

ACMV	Air-Conditioning and Mechanical Ventilation
ANN	Artificial Neural Networks
BEMS	Building Energy Management Systems
DCV	Demand-Controlled Ventilation
ELM	Extreme Learning Machine
FS-ELM	Feature-Scaled Extreme Learning Machine
HFS-ELM	Hybrid Feature-Scaled Extreme Learning Machine
PID	dynamic features of the FS-ELM
PIR	Passive Infra-Red
RIG	Relative Information Gain
RIG-ELM	RIG-based ranking algorithm using ELM
SVM	Support Vector Machines
WRANK-ELM	Wrapper-based ranking algorithm using ELM

Chapter 1

Introduction

1.1 Scope and Overview

According to the Building and Construction Authority of Singapore, the building sector consumes up to 38% of the nation's electricity, of which 50% can be attributed to the occupants [77]. In light of our limited resources, this poses an important challenge for Building Energy Management Systems (BEMS): on one hand, they must maintain thermal comfort of occupants, while on the other, conserve energy. This requires a greater consideration of the occupancy information. In fact, occupancy-driven systems promise significant energy savings, as has been repeatedly demonstrated for intelligent lighting systems [13], demand-controlled ventilation [91, 65, 2, 97], occupancy based temperature control [33] and airflow control [68]. There is thus considerable research on estimating occupancy in real-time. However, many of the approaches reported in the literature are intrusive to the privacy of occupants, such as the use of cameras [32] and wearable sensors [63]. The works dealing with non-intrusive approaches are still limited. A non-intrusive approach is to use motion sensors [70]. However, these are useful only for binary detection, not for estimating the overall occupancy level, especially for spaces with a large number of occupants.

A promising non-intrusive approach of estimating occupancy that has emerged recently is the use of environmental sensors. Environmental parameters such as carbon dioxide, temperature and humidity can have a significant correlation with occupancy in a closed space [111]. However, to obtain accurate occupancy information, it is essential to derive useful features from the environmental parameters, since the raw data can be noisy. Additionally, to

maximize accuracy and minimize computation time, only the most relevant features ought to be selected for estimating occupancy. Thus, feature selection techniques are a central part of this problem.

Feature selection techniques are largely of two kinds: filter and wrapper approaches. The filter approach measures the relevance of features independently of the classifier used for estimation, while the wrapper approach uses the classifier itself. The result is that the filter approach is fast, while the wrapper approach is generally more accurate, since it is optimized for the particular classifier used [57]. Another class of algorithms is of hybrid feature selection, which combines the advantages of filters and wrappers. Hybrid algorithms have shown promise in a number of applications [53, 8, 103], but have not been used for occupancy estimation problems. Most works on occupancy estimation have used filter approaches, while a recent work used a wrapper approach [72].

Feature selection techniques when coupled with data-driven models such as ELM, Artificial Neural Networks (ANN), Support Vector Machines (SVM) etc. can enable us to estimate the occupancy in real-time. In the literature, a number of such schemes have been proposed [28, 31, 60], which can be classified into two categories. The schemes of the first category employ *static* features, which are defined instantaneously. A static feature may be a function of some past values, but the information is condensed into single time instants. These features are typically used to train machine learning models. Largely all the reported works on occupancy estimation utilize these features. The schemes of the second category employ *dynamic* features, which are defined in a past time window. These features contain greater information about the past context of a system. The use of these features for occupancy estimation has to date been very limited.

In this work, we seek to contribute to both paradigms of feature-selection based occupancy estimation. In the context of static features, we introduce the use of wrapper and hybrid feature selection to address some limits on accuracy and computation time of past works. In the context of dynamic features, new schemes are devised to exploit these features for occupancy estimation.

Additionally, a lot of the reported work is limited to low-occupancy settings with simple occupancy profiles, ranging from 5 to 10 occupants. In this work, we study the effectiveness of feature-selection based approaches in a

high occupancy setting.

1.2 Major Contributions

Our main contributions are as follows:

- *Occupancy estimation using static features:* We present two novel feature selection algorithms for estimating occupancy from static features, namely WRANK-ELM and RIG-ELM. Moving away from the filter paradigm of most previous works, our algorithms are wrapper and hybrid feature selection algorithms. The proposed wrapper, called WRANK-ELM, uses the ELM classifier to identify a feature combination that yields the best occupancy estimation accuracy. Our proposed hybrid, RIG-ELM, is a filter-wrapper algorithm that uses the RIG criterion and the ELM classifier to find the best feature combination. Our algorithms, due to their use of the classifier in the feature selection, can obtain a better occupancy estimation accuracy than filter approaches. Additionally, the proposed algorithms exploit the fast learning speed of the ELM. Also, both algorithms use a top-down search of an ordered list of features. In contrast to the exhaustive combinatorial search used in past works, this is a far more efficient approach. Our proposed algorithms can therefore evaluate large feature sets with a small computation time. We compare the performance of the two algorithms with experimental results to understand the trade-off in accuracy and computation time.
- *Occupancy estimation using CO_2 -based dynamic models:* We present a novel machine-learning based approach to estimate occupancy from dynamic features of CO_2 . Our method is called the Feature-Scaled Extreme Learning Machine (FS-ELM), which is a discrete-time dynamic model of occupancy. Specifically, the occupancy is modeled as a function of the CO_2 data, venting level and the past occupancy level in a moving time horizon. The FS-ELM is a novel ELM architecture that contains an additional layer called the ‘feature layer’ to extract dynamic features from its inputs. Additionally, we incorporate a scaling strategy into the FS-ELM architecture to balance the relative influence of its inputs. We also present a smoothing strategy for CO_2 data that notably improves the performance of the FS-ELM. We showed

through experimental results that the FS-ELM is highly accurate even in complex occupancy estimation scenarios.

- *Occupancy estimation using multiparameter dynamic features:* We extend the FS-ELM to multiple parameters and features in the form of the Hybrid Feature-Scaled Extreme Learning Machine (HFS-ELM). The HFS-ELM is a machine learning framework for extracting dynamic features from any type of parameter. We define feature selection matrices to extract a diverse set of features. We incorporate the HFS-ELM into a hybrid feature selection framework, which is novel in its use of both static and dynamic features. Through experimental results, we demonstrate the excellent performance of the HFS-ELM both in terms of accuracy and computation time.

Additionally, for all the proposed algorithms, we present the results for multiple sensors, sensor combinations, occupancy levels and time resolutions.

1.3 Outline of the thesis

Chapter 1 introduces the scope of this thesis and provides an overview of the motivation behind this research work. Also, the main contributions are stated and a brief outline of this thesis is provided.

Chapter 2 reviews the existing literature on occupancy estimation in terms of the various sensing technologies used. We then briefly review the types of feature selections and some recent algorithms reported in the literature. We then look at the application of the ELM and of dynamic models for occupancy estimation.

Chapter 3 presents novel wrapper and hybrid feature selection techniques for occupancy estimation. In this chapter, we elaborate on the experimental data collected and the features extracted from these data. We then explain the motivation behind the proposed algorithms, elaborating on the computational cost and limits on accuracy of some previous works. We then analyse the features selected by the algorithms.

Chapter 4 presents experimental results for the proposed wrapper and hybrid algorithms. We consider the estimation of both pre-defined occupancy levels and of actual occupancy numbers. We compare the performance of the proposed algorithms with past work.

Chapter 5 introduces a novel technique to estimate occupancy from CO_2 -based dynamic features. We develop a novel machine learning architecture that is a variant of the standard ELM. The proposed method extracts dynamic features from CO_2 data and incorporates a scaling strategy to balance the influence of inputs of varying magnitude. Also, we present a smoothing strategy to improve the estimation of the proposed method.

Chapter 6 presents a feature selection framework for dynamic features. The framework is a filter-wrapper hybrid that finds a feature combination that yields the best occupancy estimation accuracy. To work within this framework, we present an extension of the FS-ELM called the HFS-ELM. The HFS-ELM is a generalized machine learning architecture that can extract a large and diverse set of features from any type of parameter. We study the features selected by this framework.

Chapter 7 presents the experimental results for the FS-ELM and HFS-ELM for occupancy number and occupancy level estimation. We present the results for multiple sensors, sensor combinations and time resolutions.

Chapter 8 summarizes this thesis and presents suggestions for future work.

Chapter 2

Literature Review

2.1 Introduction

In this section, we review the state-of-the art in indoor occupancy estimation. In Section 2.2, various sensor-based techniques that have been used for occupancy estimation are studied, ending with a discussion on environmental sensors. In Section 2.3, filter, wrapper and hybrid feature selection methods are reviewed. In Section 2.4, the use of the ELM for occupancy estimation is studied. In 2.5, the use of dynamic models for occupancy estimation is reviewed.

2.2 Sensors for occupancy estimation

2.2.1 Binary sensors

The most widely used approach to occupancy sensing in non-domestic buildings has been the deployment of PIR sensors, mostly to optimize the use of lighting. The PIR sensor typically functions as a timer that signals for the lighting to turn off if it does not detect the presence of an occupant. Through some generic assessments for commercial buildings, ASHRAE estimates an average of 30% energy savings using occupancy sensors [1]. A number of PIR-based occupancy detection systems have been reported in the literature. For instance, Delaney et al. [25] used a PIR sensor network in an office space to detect occupants for lighting control. They found that energy savings of 7.9% could be achieved in comparison to a standard manual switch lighting

control system, although this would clearly vary with occupant behavior. Garg and Bansal [37] attempted to minimize the problem of false-offs by adapting the time delay of the PIR sensors to the known activity patterns of an occupant. They modeled the inactivity period of the occupant using a Pareto distribution. However, they assumed the same trend to repeat itself every day. To compensate for their shortcomings, PIR sensors are often coupled with other sensors. Dodier et al. [27] moved away from the paradigm of using a single expensive detector to deploying multiple detectors and using a complex inference algorithm to make a decision. With three PIR sensors and a detector that sensed when the telephone handset was “off-hook”, they used a Bayesian belief network to probabilistically infer occupancy. A Hidden Markov Model (HMM) was used to model the occupied state of an office room. Ground truth occupancy was recorded using a camera and manually counted from the images. This work was limited to detection and could not determine the number of occupants. Padmanabh et al. [80] created a PIR and sound sensor network in a conference room to minimize energy wastage during unoccupied periods. They tested the system for two months and managed an average of 13% energy savings. Lu et al. [69] used PIR sensors and magnetic reed switches for occupancy detection in a home. They used a Hidden Markov Model (HMM) to determine whether the occupants were at home, away or asleep. They achieved a classification accuracy of 88% for occupancy detection. Also, their system performed better than a typical reactive thermostat in terms of energy savings. An issue with the ground truth data they collected was that it consisted of best estimates aided by interviews with the residents to clarify ambiguous data. It was thus not as reliable as data from a camera. PIR sensors were part of a test-bed in the work of Dong et al. [28] that was used for occupancy estimation, in combination with other environmental sensors such as for CO_2 , temperature, humidity and Volatile Organic Compounds (VOC). After a feature selection process based on the mutual information values of extracted features and occupancy levels, they shortlisted PIR sensor data as being relevant for occupancy estimation.

PIR sensors have a few shortcomings. Firstly, they cannot detect immobile occupants. Secondly, they cannot detect multiple occupants passing through their field of view. Thirdly, they are suitable only for presence and absence detection. Fourthly, they can be falsely triggered by heat currents from HVAC systems [99]. To get an accurate representation of the entire room’s

occupancy level, a large network of motion sensors would be required.

To some extent, ultrasonic sensors address the problems with PIR sensors. In contrast to the passive nature of PIR sensors, ultrasonic sensors do not require line of sight and continuous movement [60]. While their effectiveness for presence detection has been demonstrated [102], like PIR sensors, they are limited to binary detection and are susceptible to false-offs [41]. Yavari et al. showed that Doppler radar sensors [110] are effective at distinguishing human and non-human periodic motions, which could reduce the rate of false-offs. However, no investigation of detection accuracy was carried out. Additionally, these sensors are also limited to binary detection.

2.2.2 Wearable sensors

Wearable sensors have shown to be effective in tracking occupants. They are frequently used in monitoring patients for health care. Sokwoo et al. [87] developed a sensor that would allow for twenty-four hour monitoring of the blood flow of patients. It was in the form of a ring to be worn on the finger. This form of sensing suits a dedicated application such as the monitoring of elderly patients. Similarly, sensors in the form of wristwatches have also been designed [27]. A ‘smart shirt’ that monitors vital signs has found applications in defense, health monitoring, public safety and sports [83]. In [113], RFID tags worn by occupants were used for localization with the aid of a support vector machine (SVM) algorithm. They performed regional, as opposed to point-to-point, localization using a round-robin comparison, which improved the robustness of the system to the changing environment. The occupants were tracked with an accuracy of 93%. The latency and scalability of their system would require improvement to suit non-domestic buildings. Gillot et al. [39] achieved a tracking accuracy to within 15 cm in three dimensions with ultra-wideband RFID tags. Li et al. [63] used RFID tags to monitor occupancy in real-time and identify the thermal zones in which the occupants were positioned in an office building. They employed a K-nearest neighbor algorithm for the tracking, achieving an average zone detection accuracy of 88% for seated occupants. When the occupants were moving, however, the detection accuracy reduced to 62%. The authors noted an issue with the RFID system in that increasing the tag density also increases the risk of radio signal collision, thus compromising the tracking accuracy. Also, the tracking accuracy degraded when occupants moved close to the boundaries

of the sensing area.

Wearable sensors have their advantages. They are mobile and can measure otherwise inaccessible physiological parameters [58]. They also allow for tracking, not mere estimation. However, aside from the specific issues with RFID tags mentioned above, wearable sensors are intrusive, which limits their application in a building environment.

2.2.3 Device and appliance monitoring

Melfi et al. [73] presented a method to estimate the occupancy based on the number of mobile phones connected to the wireless network inside a room. They identified a strong correlation between the number of active Wi-Fi hosts and the electricity use inside the test venues. They also monitored the outgoing telephone calls, triggering of security door sensors and badge in/out systems. Additionally, they recorded keyboard and mouse use at the office workstations. They found that the Wi-Fi monitoring method suffered from low accuracy due to coverage overlap of wireless Access Points (AP) and inconsistent Wi-Fi connectivity of mobile phones. In order to address these limitations, Balaji et al. [9] augmented information from Wi-Fi logs with metadata information such as occupant identity, WiFi MAC address and AP location. They achieved detection accuracies of 83%. However, they noted that due to various reasons, such as power management, low connectivity or alternate internet connections on desktops, many users were not logged on to the Wi-Fi, affecting the accuracy of the system. Other Wi-Fi based approaches suffer from the same limitations [78, 51, 90, 52, 38, 34, 79] As an alternative to Wi-Fi, some works proposed bluetooth-based detection [35, 20, 21]. A key advantage of this approach is that no processing has to be done on the mobile phone, thus enabling low power and low capacity mobile phones to be used as well. However, these approaches remain dependent on the presence and active function of smartphones. They are also not strictly non-intrusive, since device-specific information is solicited.

2.2.4 Camera-based sensors

Cameras offer a highly accurate method of estimating occupancy. For instance, İçoğlu and Mahdavi [49] created a camera-based occupancy estimation system for a sentient building. Sentient buildings contain an ex-

tensive sensory infrastructure that allows for real-time monitoring and updates of important components and processes. Their system implemented an internet-based distributed data fusion algorithm to sense occupants. The cameras wirelessly transmitted to computers scattered through the building, which in turn transmitted to the main server. The advantage of the distributed structure is that the configuration was flexible, that is the number of cameras and IPU's active could be chosen according to coverage and speed requirements, respectively. The drawback though is that the scheme is computationally quite expensive with multiple cameras due to the many components that work in parallel. Another video-based real-time occupancy estimation scheme was implemented in [101]. Video cameras recorded the flow of occupants. The people count was updated every time an occupant passed the camera's field of view. For emergency situations, they added an Extended Kalman Filter based on a nonlinear dynamic stochastic state-space model for the movement of people, which improved the estimation accuracy. Benezeth et al.[10] developed an image-based occupancy detection system using static cameras. They also characterized the activity of the occupants. They analyzed the image in three stages, which were background subtraction, tracking and recognition. For background subtraction, they modeled each pixel in the image by a Gaussian probability density function and monitored for changes. For tracking, they distinguished connected components in the image and applied a tracking algorithm. The recognition of objects was performed using a cascade of boosted classifiers. They achieved a detection rate of up to 97%. The system faltered when the contrast between the occupant and the background was not high and also when he/she took an unusual posture. Silvestre and Perez [95] presented an analysis of energy savings possible by controlling the lighting based on surveillance systems in manufacturing industries. They suggested that 70% energy savings could be achieved compared to conventional lighting control systems.

Liu et al. [66] presented a strategy to fuse the detection results from different vision sensors. Their approach was to first apply a Haar-Adaboost classifier to detect the occupant's heads and to apply contour analysis to remove the non-head objects from the images. They then modeled the detection results of the various vision sensors as the observation variables of a Dynamic Bayesian Network (DBN), from which they inferred the overall occupancy. They presented the results for a single entrance only and for a limited time window of 15 minutes. For the results to be conclusive, at least an entire

work-day should be considered.

Shih et al. [94] used depth sensors and pan-tilt-zoom (PTZ) cameras for tracking and counting occupants in a building. A particle-filter based tracking algorithm was proposed. They suggested to use the skeletal, contour and shape features of the occupant for recognition. However, no estimation results were provided. Diraco et al. [26] presented an occupant localization scheme that does not require tracking, by using features extracted from 3D depth sensor data in combination with a Real AdaBoost classifier. By eliminating the need for tracking, significant computational resources were saved.

Erickson et al. [33] deployed a network of low-power cameras to collect ground truth occupancy data, which they used to create an inhomogeneous HMM of the occupancy. They showed that with real-time occupancy data, up to 42% energy savings would be possible in their test area.

While vision-based systems often provide high accuracy, the issues of intrusion of privacy, high computational cost and the large storage requirements of the data collected are significant concerns. They can only be deployed in selected venues where both permissions and resources are abundant. For a more detailed discussion of privacy in wireless network infrastructures, the reader is referred to [96].

2.2.5 Environmental sensors

Occupancy estimation with environmental parameters requires the extraction and selection of useful features from the data. These features can then be applied to machine learning to generate occupancy estimates. For instance, Zhang et al.[111] measured environmental parameters such as CO_2 , temperature, humidity and acoustic levels in an office space. From these data, they generated various features that could capture the variations in occupancy. They then analyzed each environmental parameter and its features individually, shortlisting the best for each. Then, they analyzed the best features of each parameter in combination with each other, finally shortlisting the best features overall. The merit of the features was measured using Relative Information Gain (RIG). In [28], these features were tested with a number of machine learning algorithms. Their best results were with the Hidden Markov Model (HMM). The accuracy was 75% on average for

a maximum of four occupants. Ai et al. [3] used a similar combination of sensors with an Autoregressive Hidden Markov Model (AHMM), achieving a slightly smoother profile than with a standard HMM.

Mamidi et al. [71] deployed CO_2 , temperature, humidity, light and motion sensors in two lab spaces with up to ten occupants each. They created a feature set that contained, in addition to the raw sensor data, moving averages and correlation values of CO_2 data. Specifically, they included a variety of time windows for the moving averages and correlation values. These features were used to train a number of machine learning methods, of which the multilayer perceptron yielded the highest accuracy. For the ground truth occupancy data, the authors used counter apps on iPads placed at the doorways of the labs. They suggested that the resulting occupancy data was reasonably consistent with the images of a camera they installed for verification. We tested this approach in the Process Instrumentation lab at NTU. It was frequently observed that occupants would forget to input their entry or exit into the lab, despite extensive signage. The method does not have the guaranteed reliability needed of ground truth data, even if it does not compromise the accuracy significantly. Additionally, if the authors had selected the most effective features instead of using all of them for training the machine learning algorithms, as they did, it would have considerably reduced the computational burden.

Hailemariam et al. [42] embedded CO_2 , light, sound, power use and motion sensors into a single cubicle for seven days. They used the data only for occupancy detection (presence or absence), not estimation. For feature selection, they used decision trees. They found that while the use of motion features alone produced good detection accuracy, the inclusion of features derived from other sensors did not improve the overall accuracy. This was likely due to overfitting, which is a familiar problem with decision trees. The features of CO_2 did not yield high scores based on the weighted feature occurrence of the decision trees. The implication of this contradicts most other works on occupancy detection, which rank CO_2 among the highest in their respective feature selection techniques. Worner et al. [107] used a Netatmo weather station to collect CO_2 , temperature, relative humidity, pressure and noise data to automatically detect occupancy in a two person office. This would help save energy by running the heating only when the occupants were away. They used simple heuristics to generate an occupancy profile based on the sensor data. No assessment of the accuracy was provided.

In [31], a sensor network monitoring CO_2 , relative humidity, temperature, sound level and PC case temperature was deployed in an office space. Features extracted from the data were evaluated using Symmetric Uncertainty (SU), which is also based on IG. Those features that had the same SU value were regarded as redundant and were removed. Then, a Correlation Feature Selection (CFS) filter was used to find the best combination of features across sensing domains. The best features overall were used to train an Artificial Neural Network (ANN) to estimate the occupancy. Their accuracy was also limited to 75% for a maximum of six occupants.

Yang et al.[108] measured up to eleven environmental parameters in both single and multi occupancy rooms. They took an approach similar to [28], using IG to assess the parameters individually and then in combination with each other. The best parameters were used with various machine learning algorithms to perform the estimation, with decision trees yielding the best accuracies. However, they did not extract features from the data, instead using the sensor data in its raw form.

Amayri et al. [4] used luminance, temperature, relative humidity, motion, CO_2 concentration, power consumption, door and window positions and acoustic signals to extract occupancy. To assess the features, they used information gain incorporated into a decision tree. They found the motion, acoustics and power consumption data to be most informative according to their approach. To further enhance the accuracy, they applied a random forest technique to multiple decision trees. They estimated the occupancy at 30 minute intervals, achieving an accuracy of 70-75% on working days. The occupancy generally did not exceed four. Zikos et al.[114] used pressure mats and double-beam, acoustic, PIR and CO_2 sensors. They extracted two kinds of features from the data: a simple moving average and weighted moving average. They used Conditional Random Fields (CRF) to estimate the occupancy from the sensor data. They found that the larger the number of sensors they used, the higher was the accuracy. However, concerns of cost, privacy and intrusiveness were also important in determining the practicality of various sensors. For example, they found that double beam sensors contributed the highest to increasing the accuracy. But these sensors are obtrusive to occupants and may not be suitable for every indoor space.

Cheney et al. [14] derived features from electrical power, dew point temperature and indoor CO_2 concentration inside a home. They used a HMM to predict the occupancy profile from these features. For the emission model of

the HMM, they used the Dempster-Shafer theory, which deals with missing sensor data by assigning “ignorance” to the particular sensor of the actual state of the system in the affected time period. They noted that a drawback of using electrical power was that electricity usage is quite low at night, making it difficult to determine night-time occupancy. Also, the ground truth data was limited to four days and was provided by the occupants, in this case the residents of the home. Obtaining ground truth from occupants can be unreliable, as is discussed in Section 3.3.2.

Candanedo et al. [12] monitored CO_2 , temperature, humidity and light levels in an office space. In addition, they added the time and date stamps of the data as features. Also, a humidity ratio that incorporated pressure information was considered. The sensors were placed on a desk. They used Linear Discriminant Analysis (LDA), Classification and Regression Trees (CART), Random Forest (RF) models and Gradient Boosting Machines (GBM) to evaluate the features. They found the best accuracy was achieved using a combination of all features and with the RF models. This work was limited to a two-occupant room and to detection of occupant presence only.

Ryu et al. [89] used time stamps, power consumption data and CO_2 data to estimate the occupancy in an experimental room. For CO_2 data, they used the first order difference, 15 minute moving average, 15 min rate of change and ratio of indoor and outdoor CO_2 concentration features. As a preprocessing step, they discarded features with low values of IG with occupancy. They then used the CART model for occupancy estimation and the HMM for occupancy prediction. They considered the case of a 5-occupant testing room. They found that the CO_2 data, especially the indoor to outdoor concentration ratio, gave the best prediction accuracy. However, they noted that the HMM did not perform well on low occupancy days.

It is evident from the above works that multiple parameters have shown promise for occupancy estimation problems. Taking into consideration the diversity of these parameters, a more generalized approach is needed to judge the relative effectiveness of parameters. Additionally, the reported works are limited to using static features, as opposed to the more informative dynamic features. In this work, we combine static and dynamic features into a framework that can assess a diverse set of parameters and features.

2.2.6 Applications of sensor-based occupancy detection

The occupancy detection schemes studied in the previous sections have a number of important applications. For instance, occupancy numbers can enhance demand-controlled ventilation (DCV), which is defined by ASHRAE as being the regulation of outdoor-air flow based on the estimated occupancy and ventilation requirements of the occupied zone [6]. DCV is important to maintain indoor air quality and the comfort of occupants. Presently, DCV systems infer the occupancy from the CO_2 levels alone [67]. A number of concerns with these systems have been identified, such as the overall accuracy, location of sensors in the indoor space and the number of sensors required [29]. There is a need to investigate accurate and cost-effective ways of estimating the occupancy from CO_2 data that would not require a dense sensor network.

Lighting control is another important application of occupancy detection. Lighting energy demand can constitute up to 25-35% of the total energy consumption in an office building [82] and thus there is great potential for energy savings in this area. To achieve this, automatic lighting control using occupancy sensors has been widely implemented, with energy savings of 20-60% reported for various office settings and sensor types [19, 104, 36]. Motion sensors are widely used today in commercial lighting control [24]. Considering their limitations discussed in Section 2.2.1, there is still room for improvement in accuracy and reliability of these systems, which can be addressed by a fusion of various sensors.

Applications of occupancy information in security systems has also been investigated. For example, a recent patented technology uses occupancy sensors to reduce the instances of false alarms in security systems [61]. A monitoring system for occupants evacuating in the event of a fire alarm was investigated in [101].

2.3 Feature selection methods

2.3.1 Filter methods

Feature selection is often combined with machine learning to remove irrelevant and redundant features, both to allow the algorithm to process faster

and to ensure better accuracy. Once features are extracted from the raw data, a few different approaches can be taken when selecting the features that would give the best classification accuracy with a machine-learning algorithm. There are essentially two paradigms to feature selection for machine learning, the filter model and the wrapper model [57]. These are discussed below.

In the filter model, the feature space is reduced progressively by eliminating lower-ranked features according to a merit criterion. In the wrapper model, the algorithm used to make the final inference is used in the feature selection. The criterion for eliminating redundant features in this model is generally of two kinds: consistency-based and correlation-based. Consistency measures search for a minimum number of features that distinguish classes within the data as consistently as the main feature set can [22]. If two instances have the same feature values but different class labels, it is identified as an inconsistency. The issue with this approach is that many features may be required to achieve consistency, thus making the exhaustive search intractable. Additionally, the algorithm may continue to add features to repair a single inconsistency (i.e. with one particular class label) thus over-fitting the training data [30]. Correlation measures instead evaluate how well the features are correlated with the target classes. This work aims to investigate the viability of using environmental parameters for occupancy estimation, which is why a correlation measure is used in the proposed algorithms.

2.3.2 Wrapper methods

In the wrapper model, the learning algorithm is used for the feature selection. That is, the different feature subsets in the feature space are tested progressively with the learning algorithm until the optimal subset is found. Since the learning algorithm is applied to every possible subset, the computational burden can become quite high, especially for large datasets. The advantage of the filter approach over the wrapper model is that the execution time is faster, which is especially significant for large datasets. Also, since the filter selects features independently of the machine learning algorithm, re-execution is not necessary. The key drawbacks of the filter model are that it does not take into consideration the influence of the feature subsets on the classification accuracy, nor does it take into account the biases of the machine learning algorithms [57]. In the wrapper model, the selection is op-

timized for the particular learning algorithm used, therefore often yielding better results than the filter model [43].

2.3.3 Hybrid methods

A valuable trade-off of accuracy and computation time is offered by hybrid feature selection algorithms. These algorithms generally contain a filter and a wrapper, usually in a two-stage algorithm, thus benefiting from the speed of filters and the accuracy of wrappers. A number of filter-wrapper combinations have been reported in the literature. Ruiz et al. [88] presented two frameworks for hybrid algorithms. In the first framework, features were ranked according to their relevance using either a filter or a wrapper. Then, starting with the top-ranked feature, lower ranked features that increased the accuracy significantly (as measured by a t-test) were added to the feature subset. In the other framework, they assessed subsets of increasing sizes, discarding those that measured poorly with either a filter or a wrapper. They tested their frameworks with a CFS-based filter and with wrappers that used Naïve Bayes and C4.5 [85] as the learning algorithms. They found that the hybrid algorithms could achieve speeds of up to ten times faster than the equivalent wrappers while maintaining a comparable accuracy. Sebban et al. [93] proposed an algorithm that used Relative Certainty Gain (RCG) as the criterion that was optimized with a 1-nearest neighbor graph framework. They found that the RCG was a better criterion to optimize than the accuracy of the 1-NN rule. However, this may be specific to the 1-NN and may not hold for all machine learning algorithms. Naseriparsa et al. [76] used IG to rank features, followed by a genetic search using the Naïve Bayes algorithm. In their filter phase, they discarded features that were below a threshold. This may have eliminated some features that, while not ranked highly by the filter, would still improve the accuracy in combination with other higher ranked features. Apolloni et al. [7] used an IG-based filter and a Binary Differential Evolution (BDE)-based wrapper on a microarray dataset. Instead of a threshold value, they used a certain percentage of features from the ranked list. However, the problem of discarding possibly useful features remains. Recently, Chen et al. [16] used a number of filters, such as mRmR [84], IG, Relief and t-test in combination with the ELM for early diagnosis of Parkinson's disease. They demonstrated that applying the filters to the PD dataset prior to classification improved the estimation

accuracy of the ELM.

2.4 Occupancy estimation using ELM

The ELM learns much faster than other machine learning algorithms such as ANN, SVM and HMM that have been used for occupancy estimation. Only a few works have explored the potential of the ELM for occupancy estimation. Masood et al. [72] presented a wrapper algorithm for feature selection based on the ELM. The algorithm first identified the best features corresponding to each individual environmental parameter. Then, the best features of each parameter were combined together and a combinatorial search was performed to find the best feature combination. This approach is feasible only for a small set. For a larger feature set, the computational burden would become prohibitively large. The authors of [112] developed a Class Incremental Extreme Learning Machine (CLIEM) to perform activity recognition from sensor data collected from smart phones. They addressed the problem of adding new activities to the classifier without having to re-train the model. This meant that the need to store the original sensor data when performing recognition with newly added activities, which would not be feasible for a mobile phone, was eliminated. The method did not perform better than the default approach of batch learning with both the original and new datasets combined. In [116], the ELM was used for localization of occupants in an indoor environment. The ELM was trained using Received Signal Strength (RSS) values from RFID tags attached to occupants as inputs and the location of the occupants as outputs. They found the ELM algorithm to provide better localization accuracy and to be more robust than a Weighted Path Loss (WPL) algorithm, but the WPL was seen to be faster. The authors therefore integrated the ELM and WPL algorithms for localization in [117], which enhanced the localization accuracy by 33% and allowed for a reduction in RFID sensors. However, the intrusive nature of RFID tags, in fact all wearable sensors, remains an issue. In [115], the authors used an online sequential ELM for localization. They intended to reduce the extensive time and labor required to collect RSS data for offline training, as well as function better in the face of the changes in the real-time environment beyond the capability of fingerprinting methods. The OS-ELM also consists of an offline training phase, but only as an initial model. The online phase received further RSS signals and updated the initial model.

This in effect reduced the resources needed for data collection. Additionally, the localization accuracy improved 30% over batch ELM.

2.5 Occupancy estimation using dynamic models

Among all the environmental parameters, CO_2 concentration is the one that most correlates with the number of occupants. The indoor occupancy level can be solved from the mass balance equation[54, 5, 105, 11]

$$m \frac{dc_t}{dt} = -Q(c_t - c_t^{\text{sup}}) + P o_t \quad (2.1)$$

where m is the mass of indoor air, Q is the airflow rate, P is the CO_2 generation rate per person, o_t , c_t , and c_t^{sup} are the number of occupants, CO_2 concentration of indoor air and supply air at the time instant t , respectively. This physical model assumes that the indoor CO_2 concentration is uniform. In real applications, this method suffers from three issues: 1) unpredictable opening of doors and windows [75], 2) uncertainties involved with the CO_2 concentration measurements, and 3) error of the calculated differential due to inaccuracies in the CO_2 measurement (the accuracy of the commonly used CO_2 sensors is around a few tens ppm). An approximated explicit solution in terms of the number of occupants obtained from the mass balance equation (2.1) has been used in [5, 11]. If using the approximated solution, we can avoid calculating the differential of CO_2 concentration but many other parameters are required to be measured. For more detail about the approximated solution, see [11]. However, the first two issues are not easy to overcome when using this method.

A recent work investigated the dynamic CO_2 concentration of a small seminar room under various scenarios in terms of different ventilation flow rates and all potential changes of occupancy level [40]. The occupancy changes can be estimated via the analysis of the dynamic CO_2 concentration. However, this work only considered the case that the number of occupants was increasing, and this method is impractical for the large rooms in which the number of occupants may have a large change in a short time interval.

In [30], a linear model of occupancy level in terms of CO_2 concentration and temperature was constructed. This method was tested in a room with

a maximum of 4 occupants. The number of occupants was estimated with a 5 minute time delay with an accuracy of up to 88.8%. In [4], the measurement of CO_2 was combined with indoor power consumption, PIR sensors and a microphone for occupancy estimation. The testing average error was 0.19 occupants for a room with maximum 4 occupants. The estimator was designed by solving a deconvolution problem. In this strategy, the deconvolution process suffers from truncation errors and the error of the identified CO_2 dynamic model. In this work, we present a machine-learning based approach to directly identify the occupancy estimator, thus avoiding the deconvolution process.

Part I

Occupancy estimation using static features

Chapter 3

Occupancy estimation using wrapper and hybrid feature selection

3.1 Introduction

To estimate occupancy from environmental sensors, it is essential to extract and select useful features from the raw data. In this chapter, two novel feature selection algorithms for occupancy estimation are presented. In Section 3.2, a brief background about the concepts behind RIG and ELM which are used in our algorithms is given. In Section 3.3, the sensor test-bed and data collection is elaborated on. An efficient way of collecting ground truth occupancy data is presented. In Section 3.4, the extracted features are defined and the proposed wrapper and hybrid feature selection algorithms are presented. Then, in Section 3.5, the features selected by the algorithms are analyzed.

3.2 Background

3.2.1 Relative Information Gain (RIG)

In this work, the correlation between environmental parameters and occupancy is studied. To do this, we model the environmental parameters and occupancy as random variables. To measure the correlation between these

variables, we apply the concept of entropy from Information Theory, which measures the uncertainty of a random variable. Let y be a random variable representing the occupancy. The entropy of y is defined as

$$H(y) = - \sum_{i=0}^n P(y_i) \log P(y_i) \quad (3.1)$$

where n is the sample space of y .

Let x be a random variable representing a vector of features extracted from an environmental parameter (more details about feature extraction are given in 3.4.1). The conditional entropy of y given x is defined as

$$H(y|x) = \sum_{j=0}^m P(x_j) H(y|x_j) \quad (3.2)$$

where m is the sample space of x .

The reduction in entropy of y given x indicates the additional information about y gained from x . It is thus termed ‘information gain’ and is defined as

$$IG(x, y) = H(y) - H(y|x) \quad (3.3)$$

The Relative Information Gain (RIG) is expressed as [28]:

$$RIG(x, y) = \frac{IG(x, y)}{H(y)} * 100\% \quad (3.4)$$

In Section 3.4.3, we incorporate the RIG into an algorithm to determine the features most correlated with occupancy.

3.2.2 Extreme Learning Machine (ELM)

Extreme Learning Machine (ELM) was introduced as a learning algorithm for single-hidden-layer feed-forward neural networks (SLFNs) [48]. It has shown many advantages for multi-class classification and regression problems, especially because of its extremely fast training speed. The training of the ELM is computationally efficient due to the random assignment of input layer parameters that do not have to be adapted to the data. Among the best characteristics of the ELM are its universal approximation capability, improved generalization performance, and that it yields a global optimum

for the convex optimization problem and its closed form solution eliminating iterative learning [48]. While both the input and output layer parameters are tuned in conventional ANNs, the ELM randomly assigns the former. This eliminates the need for iterative training. The gradient-based learning algorithms that are employed in training ANNs, most commonly the back-propagation method, involve a trade-off in the choice of learning rate (small values mean slow convergence while larger values lead to instability and divergence) and may get trapped at local minima. In the ELM, the input data is projected onto a high dimension space using the hidden neurons and the randomly initialized parameters. The nonlinear data is transformed to a linear pattern, which is used to model the outputs using a least-squares method [50]. With linear least squares used for training and a convex objective function, the ELM learns extremely fast and is not prematurely terminated due to local minima.

The output of the ELM in its basic form is identical to that of a SLFN and can be expressed as

$$\mathbf{y}_j = \sum_{i=1}^{N_H} \beta_i \phi_i(\mathbf{w}_i \cdot \mathbf{x}_j + b_i) \quad (3.5)$$

where $\mathbf{y}_j = [y_{1j}, y_{2j}, \dots, y_{mj}]$ is the output vector of the ELM for the j^{th} sample, $\mathbf{x}_j = [x_{1j}, x_{2j}, \dots, x_{pj}]$ is the input vector for the j^{th} sample, $\beta_j = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]$ is the output weight vector connecting the i^{th} hidden node and m outputs, ϕ_i is the activation function of the i^{th} hidden node, $\mathbf{w}_i = [w_{i1}, w_{i2}, \dots, w_{ip}]$ is the input weight vector connecting the i^{th} hidden node and p inputs and b_i is the bias of the i^{th} hidden node. The structure of the ELM is shown in Figure 3.1.

Equation (3.6) can be expressed compactly in matrix form as:

$$\mathbf{Y} = \mathbf{H}\beta \quad (3.6)$$

where

$$\mathbf{H} = \begin{bmatrix} \phi_1(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & \phi_{N_n}(\mathbf{w}_{N_n} \cdot \mathbf{x}_1 + b_{N_n}) \\ \vdots & \ddots & \vdots \\ \phi_1(\mathbf{w}_1 \cdot \mathbf{x}_N + b_1) & \cdots & \phi_{N_n}(\mathbf{w}_{N_n} \cdot \mathbf{x}_N + b_{N_n}) \end{bmatrix} \quad (3.7)$$

Figure 3.1: Structure of the ELM

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1 \\ \vdots \\ \boldsymbol{\beta}_{N_n} \end{bmatrix} = \begin{bmatrix} \beta_{11} & \cdots & \beta_{1m} \\ \vdots & \ddots & \vdots \\ \beta_{N_n1} & \cdots & \beta_{N_nm} \end{bmatrix} \quad (3.8)$$

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_N \end{bmatrix} = \begin{bmatrix} y_{11} & \cdots & y_{m1} \\ \vdots & \ddots & \vdots \\ y_{1N} & \cdots & y_{mN} \end{bmatrix} \quad (3.9)$$

This work uses classification as opposed to regression. The output $\mathbf{Y}$ thus refers to class labels. Since the weights and biases are randomly generated and fixed, the ELM solves the following optimization problem:

$$\min_{\boldsymbol{\beta}} \|\mathbf{H}\boldsymbol{\beta} - \mathbf{Y}\| \quad (3.10)$$

This is a linear parameter estimation problem, whose least-squares solution can be obtained as

$$\hat{\boldsymbol{\beta}} = \mathbf{H}^\dagger \mathbf{Y} \quad (3.11)$$

where $\mathbf{H}^\dagger$ is the Moore-Penrose generalized inverse of the matrix $\mathbf{H}$. Thus, the ELM algorithm is as follows:

- 1) Set the number of hidden neurons and randomly choose the weights $\mathbf{w}_i$ and biases b_i .

- 2) Compute the hidden layer output matrix $\mathbf{H}$ using (3.7).
- 3) Compute $\hat{\beta}$ using (3.11).

3.3 Data collection

3.3.1 Environmental sensors

In this work, we recorded the CO_2 , relative humidity, temperature and pressure levels in the office space of the Process Instrumentation Lab at Nanyang Technological University (NTU), Singapore. The lab contains an open space with 24 cubicles and 11 open seats. The occupants consist of 9 PhD students and 11 research staff. About 6 to 10 of them are regularly present during working hours. Additionally, there are 6 PCs for final year undergraduate students and 5 PCs open to all students. The room is conditioned using both Active Chilled Beam (ACB) and the conventional Variable Air Volume (VAV) systems, and is mechanically ventilated. The measurements of CO_2 , relative humidity and temperature were done using the CL11 sensor from Rotronic. The data is stored in its internal memory and transferred to the laptop in Excel format using a USB cable. Pressure levels were measured using the MHB-382SD sensor from Lutron. The data was stored in a detachable memory card and transferred to the laptop in Excel format. Both types of sensors were purchased off-the-shelf and come with a plug-in power adapter. The sampling time for both sensors was set to one minute. The resolution and accuracy of the sensors are shown in Table 3.1. The sensors are attached to a plastic stand (as shown in Figure 3.3 (b)) at a height of 1.1m from the ground. The layout of the sensors is shown in Figure 3.4. As is shown in the figure, there are three sensor positions. The choice of the number and placement of sensors was considering their non-intrusiveness to occupants, in addition to allowing a comparison between the central position (Position *A*) and the two corner positions (Positions *B* and *C*). The main door opens to an outer office space for administrative staff. Another door opens to a lab area, while the third door remains closed. All the windows are closed. The size of the zone is 20m long, 9.3m wide and 2.6m high.

Preprocessing of the data was done in MATLAB. This involved removing missing values, synchronizing the time stamps of the sensors and synchronizing the sensor data with occupancy values. Since the response of building

Table 3.1: Specifications of the environmental sensors

Sensor	Measured parameter	Resolution	Accuracy
Rotronic CL11	CO_2	1 ppm	$\pm 5\%$ of the measured value
Lutron MHB-382SD	Temperature	$0.05^\circ C$	$\pm 0.3^\circ K$
	Relative Humidity	0.1 % RH	$< 2.5\%$ RH
	Pressure	0.1 hPa	± 2 hPa

(a)

(b)

Figure 3.2: (a) Rotronic CL11 sensor for CO_2 , relative humidity and temperature and (b) Lutron MHB-382SD sensor for pressure levels

control systems is slow, we used 15 minutes resolution of the data, which is adequate [81, 64]. For comparison, we also consider a 1 minute resolution. We subsequently refer to the resolution as ‘ R ’.

3.3.2 Ground truth data

Ground truth data is generally collected by counting the number of occupants from minute-by-minute images of the room [114, 12] or by live observation [55][18]. This is quite cumbersome. Other works use smartphones to collect ground truth data [15, 98, 86]. However, the smartphones may not always be switched on, thus compromising the accuracy of the data. Another work placed colored strips in the field of view of a camera, using image processing to detect when occupants passed the strips. An issue with this approach is that multiple occupants moving together may not be detected well [106]. The same issue applies to obtaining ground truth from motion sensors, as was done in [44]. Alternatively, occupants may be asked to record their entry and exit, usually on a touch-screen device [109, 71, 56].

Figure 3.3: Process Instrumentation Lab, NTU. (a) Office space and storage space of the venue (b) One set of sensors installed in the office space (c) Overhead cameras to record occupancy ground truth

This method is quite unreliable, as occupants may occasionally forget, especially when they are many. For example, in [100], more than 66% of the intended ground truth data was lost since it was dependent on the feedback of occupants. Instead, we detect the entry and exit of occupants using cameras installed at the doorways. Using motion detection software, we generate images of the entry and exit of occupants, from which we deduce the occupancy of the room. Note that the sensitivity of the motion detection software can be set quite high, which ensures that 100% of occupant entries and exits are detected. Using this approach, the number of images that need to be viewed to note the occupancy count is considerably reduced. For a typical work day, between 8:00 am and 11.30 pm, there were only about 350

Figure 3.4: Layout of sensors in Process Instrumentation Lab, NTU

instances of entry or exit of occupants. While, the conventional minutely processed image approach would need to consider 930 instances. Thus, the motion detection approach is a significantly more efficient way of creating ground truth data. In this work, we use overhead IP cameras installed at the doors (one per door) to record the entry and exit of occupants. The benefits of overhead cameras in preserving the privacy of occupants have been previously shown in [59]. We use iSpy 64 software to create images upon motion detection and manually counted the entries and exits. In this manner, 31 days of data was prepared. The maximum occupancy in the data collection period was 28.

From a practical standpoint, knowing the level of occupancy would be sufficient for the ACMV control systems of buildings. In this work, we make a balanced division of the occupancy into four levels as shown in Table 3.2.

Table 3.2: Description of occupancy levels

Occupancy level	Number of occupants
Zero	0 (Empty)
Low	1-6
Medium	7-14
High	Over 14

3.4 Feature selection

3.4.1 Feature extraction

The raw environmental parameter data is not sufficient to give good classification results. This is because firstly, it may be noisy and secondly, the correlations between the raw data and the occupancy may not be so obvious. It is thus important to extract relevant features from the raw data. We define three groups of features shown in Table 3.3. The first order shifted difference (fd) and second order difference (sd) are chosen to capture the temporal variations in the data. The moving average ($mavg$) takes into account the time delay in the build-up and decay of the environmental parameters. Considering the multiple difference ' d ' values (for fd and sd) and step ' m ' values for moving average, we obtain a total of 84 features including the raw data. Note that in the case of the difference features, when all samples are not available in the beginning (such as at $i = 1$ for fd), we treat the missing sample the same as the neighbouring one. Thus, $fd(1) = 0$. As for $mavg$, when there are missing samples in the beginning, we take the average of the available samples. For example, $mavg(2) = (\sum_{j=1}^2 raw(j))/2$.

Table 3.3: Features extracted from sensor data

#	Feature descriptions
1	$raw(n)$
2	$fd(n) = raw(n) - raw(n - d); d \in \{1\} \cup \{5\delta : \delta = [1\ 6]\}$
3	$sd(n) = raw(n) - 2raw(n - d) + raw(n - 2d); d \in \{1\} \cup \{5\delta : \delta = [1\ 6]\}$
4	$mavg(n) = (\sum_{j=n-(m-1)}^n raw(j))/m; m \in \{5\delta : \delta = [1\ 6]\}$

3.4.2 Wrapper-based ranking using ELM (WRANK-ELM)

Wrapper algorithms generally achieve high accuracy, but tend to be computationally expensive. This is due to the repeated training of the classifier that is required while assessing features. This is especially true for classifiers that train over several iterations, such as Artificial Neural Networks (ANN). To address this, we propose the WRANK-ELM, which capitalizes on the extremely fast training speed of the ELM. The aim of the wrapper is to find a feature combination that maximizes the classifier accuracy. Some previous works that used feature selection for occupancy estimation take an exhaustive search approach. For example, in [111], the feature selection algorithm performs two levels of exhaustive search: first independently for feature combinations of each environmental parameter, then collectively with the best features of each parameter combined together. With P parameters and N features (including the raw data) associated with each parameter, this requires assessing $P * \sum_{n=2}^N C(N, n)$ combinations in the first stage where n is the number of features in a combination for a particular parameter. In the second stage, $\sum_{i=2}^{B*P} C(B * P, i)$ combinations have to be assessed, where B is the number of features taken as the ‘best’ per parameter. With $P = 4$ and $N = 21$ as in our case, and taking $B = 3$ as in [111], this would mean assessing a total of 332,635 combinations. This is prohibitively large. It would be particularly cumbersome to apply a wrapper paradigm to this approach, since a very large number of models of the classifier would have to be trained. Instead, WRANK-ELM progressively searches through an ordered list of features to identify the best feature combination.

The algorithm works in two stages. In the first stage, individual features of each parameter are evaluated. The best B features for each parameter are identified. Based on the mean accuracy achieved with each set of B features, the individual sensing domains (a domain corresponding to a parameter) are ranked. For example, if the mean accuracy of the set of B features of CO_2 is the highest, CO_2 is taken as the top-ranked domain. The $B * P$ features are then combined into a single set, in order of the domain rankings. In the second stage, subsets of the $B * P$ features are evaluated, starting from the top-ranked feature and progressively adding lower ranked features. The feature subset that gives the highest estimation accuracy is identified as the best subset. The algorithm evaluates $P * N$ features in the first stage

```

/* Individual feature ranking
1  for  $i = 1$  to  $P$ 
2    for  $j = 1$  to  $N$ 
3       $A_{ij} \leftarrow ELM(f_{ij}, O)$ 
4       $A_{ij} \in \mathbf{A}_i$ 
5    end for
6     $(\bar{A}_{ij}, \bar{f}_{ij}) \leftarrow \text{sort}(\mathbf{A}_i, \text{descending})$ 
7  end for
/* Combination search
8   $\text{DomainRanks} \leftarrow \text{sort}(\text{mean}(\mathbf{A}_i), \text{descending})$ 
9   $f_M = \{\bar{f}_r^B, r = \text{DomainRanks}(1) : \text{DomainRanks}(\text{end})\}$ 
10 for  $c = 1$  to  $B * P$ 
11    $S^c \in f_M; \text{size}(S_c) = c$ 
12    $A^c \leftarrow ELM(S^c, O)$ 
13 end for
14  $S^{c*} \leftarrow \text{argmax}_S A$ 

```

Figure 3.5: WRANK-ELM algorithm

and $P * B$ features in the second stage. Taking $B = N$, only a total of 168 classifier models need to be trained. The WRANK-ELM algorithm is illustrated in Figure 3.5. The number of hidden nodes during the feature evaluations was set as 50 and for the final estimation as 100. We set a simple scaling factor of 0.0101 for the features. The final output is the mean of five iterations.

3.4.3 Filter-wrapper hybrid algorithm (RIG-ELM)

Hybrid algorithms in general can offer a valuable trade-off between accuracy and computation time compared to wrapper algorithms. Our proposed hybrid, RIG-ELM, is illustrated in Figure 3.6. With P parameters and N types of features, we have a set F of $P \times N$ unordered features. The algorithm creates a set R containing the RIG values of each feature of F with occupancy. This is therefore the filter part of the algorithm. The RIG values are then sorted in descending order to form the ordered set of features F_o . Then, the best B features are passed to the wrapper part of the algorithm. In the wrapper part, we search through the ordered list of features by starting

with the top ranked feature, then progressively add lower ranked features. That is, set S^1 contains the top-ranked feature in F_o , while S^2 contains the two top-ranked features. Each iteration thus contains a unique subset of features. We test every combination S^c of c features per combination as the input of an ELM and note the corresponding occupancy estimation accuracy. The best combination is S^{c*} , for which the accuracy is the highest.

The RIG-ELM algorithm requires the assessment of only $P * N$ features in the filter stage and only B feature subsets in the second stage. Thus, despite the wrapper part of the algorithm, the computation time is kept low.

```

/* Individual feature ranking
1 for  $i = 1$  to  $P \times N$ 
2    $R^i \leftarrow RIG(F^i, O)$ 
3 end for
4  $(R_o, F_o) \leftarrow sort(R, descending)$ 
/* Combination search
5 for  $c = 1$  to  $B$ 
6    $S^c \leftarrow \{F_o^i\}; i \in [1, N]$ 
7    $A^c \leftarrow ELM(S^c, O)$ 
8 end for
9  $S^{c*} \leftarrow argmax_{S^c} A$ 

```

Figure 3.6: RIG-ELM algorithm

3.5 Analysis of selected features

The features selected by the WRANK-ELM are shown in Tables 3.4 to 3.6. The parameters are arranged in order of their domain rankings, with CO_2 being ranked the highest. The results indicate that of all the parameters, the features of pressure are selected the least. Out of all the features, the second order difference features are ranked the highest and practically never discarded. The moving average features are discarded by the RIG-ELM.

The features selected by the RIG-ELM are shown in Tables 3.7 to 3.9. Only CO_2 features are selected by the algorithm. The moving average feature is discarded. The shifted difference features appear most prominently in all cases.

Table 3.4: Features selected by the WRANK-ELM for *Position A*

	<i>Position A</i>
<i>CO₂</i>	<i>raw_CO₂, sd1, sd5, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10, fd15, fd20, fd25, fd30</i> <i>m5, m10, m15, m20, m25, m30</i>
<i>RH</i>	<i>raw_RH, sd1, sd5, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10, fd15, fd20, fd25, fd30</i> <i>m5, m10, m15, m20, m25, m30</i>
<i>Temp</i>	<i>raw_Temp, sd1, sd5, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10, fd15, fd20, fd25, fd30</i> <i>m5, m10, m15, m20, m25, m30</i>
<i>Press</i>	<i>sd1, sd5, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10</i>

Table 3.5: Features selected by the WRANK-ELM for *Position B*

	<i>Position B</i>
<i>CO₂</i>	<i>raw_CO₂, sd1, sd5, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10, fd15, fd20, fd25, fd30</i> <i>m5, m10, m15, m20, m25, m30</i>
<i>RH</i>	<i>raw_RH, sd1, sd5, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10, fd15, fd20, fd25, fd30</i> <i>m5, m10, m15, m20, m25, m30</i>
<i>Temp</i>	<i>raw_Temp, sd1, sd5, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10, fd15, fd20, fd25, fd30</i> <i>m5, m10, m15, m20, m25, m30</i>
<i>Press</i>	<i>raw_Press, sd1, sd5, sd10, sd15, sd20, sd30</i> <i>m5, m10, m15, m20, m25, m30</i>

3.6 Summary

In this chapter, we presented two novel feature selection algorithms for estimating occupancy from environmental parameters. In contrast to the filter-based methods of past works, the proposed algorithms are of the wrapper and hybrid feature selection class. The wrapper algorithm is the WRANK-ELM, which uses the ELM classifier, while RIG-ELM is a filter-wrapper hybrid that combines the RIG criterion and the ELM. The algorithms prepare an ordered feature set based on their respective criteria. They then progressively search through the ordered set for the best feature combination. Compared to the exhaustive combinatorial search of past works, our approach ensures that the algorithms are fast, thus being capable of handling a large set of features.

Table 3.6: Features selected by the WRANK-ELM for *Position C*

	<i>Position C</i>
<i>CO₂</i>	<i>raw_CO₂, sd1, sd5, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10, fd15, fd20, fd25, fd30</i> <i>m5, m10, m15, m20, m25, m30</i>
<i>RH</i>	<i>raw_RH, sd1, sd5, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10, fd15, fd20, fd25, fd30</i> <i>m5, m10, m15, m20, m25, m30</i>
<i>Temp</i>	<i>raw_Temp, sd1, sd5, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10, fd15, fd20, fd25, fd30</i> <i>m5, m10, m15, m20, m25, m30</i>
<i>Press</i>	<i>raw_Press, sd1, sd5, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5</i> <i>m5, m10, m15, m20, m25</i>

Table 3.7: Features selected by the RIG-ELM for *Position A*

	<i>Position A</i>
<i>CO₂</i>	<i>raw_CO₂, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10</i>

Table 3.8: Features selected by the RIG-ELM for *Position B*

	<i>Position B</i>
<i>CO₂</i>	<i>raw_CO₂, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10</i>

Table 3.9: Features selected by the RIG-ELM for *Position C*

	<i>Position C</i>
<i>CO₂</i>	<i>raw_CO₂, sd10, sd15, sd20, sd25, sd30</i> <i>fd1, fd5, fd10</i>

To collect experimental data, we set up CO_2 , humidity, temperature and pressure sensors in an office space. From the sensor data, three groups of features were extracted out of which the shifted difference feature was seen to be most effective. After feature selection WRANK-ELM shortlisted a combination of all parameters, while RIG-ELM shortlisted features of CO_2 . In the next chapter, we assess the occupancy performance of the proposed algorithms.

Chapter 4

Results and Discussion: Part I

4.1 Introduction

In this section, the occupancy estimation results for WRANK-ELM and RIG-ELM are presented. In Section 4.2, the results for estimating occupancy levels at individual sensor positions and for sensor combinations are presented. In Section 7.4, the results for estimating actual occupancy numbers for two different resolutions of time are presented.

4.2 Occupancy level estimation

4.2.1 Individual sensor locations

In this section, the results for occupancy estimation for the proposed algorithms are presented. We define two performance indices to evaluate the results:

- Accuracy of estimating the occupancy level (A_L): This is the percentage of correct estimation of the actual occupancy level and is defined by:

$$A_L = \frac{S - \sum_{k=1}^S |o_k - \hat{o}_k|}{S} \quad (4.1)$$

where S is the number of samples, o is the actual occupancy level and $\hat{o}$ is the estimated occupancy level.

- Root mean squared error ($RMSE$): This is the mean magnitude of

the estimation error defined by:

$$RMSE = \sqrt{\frac{\sum_{k=1}^S (o_k - \hat{o}_k)^2}{S}} \quad (4.2)$$

We compare the proposed algorithms with that of Zhang et al.[111], which uses an RIG-based filter technique to exhaustively search for the best combination of features. To keep the computation time from becoming prohibitively large, we set the maximum combination size to six. To perform the estimation with Zhang et al.’s approach, we used the standard ELM with 450 hidden nodes. Note that scaling of the features is not discussed by Zhang et al., thus we do not employ a scaling factor.

The results for occupancy level estimation are summarized in Table 4.1. The WRANK-ELM achieves an excellent accuracy of up to 79.17% (Position *A*), nearly 12% higher than Zhang et al.’s method. The RIG-ELM also achieves an excellent accuracy of up to 77.92% (Position *B*). The *RMSE* values of WRANK-ELM and RIG-ELM remain below 0.44, while that of Zhang et al.’s method is above 0.5. Overall, the best accuracy is achieved with the WRANK-ELM at Position *A*.

The occupancy trends for the actual occupancy, WRANK-ELM, RIG-ELM and Zhang et al.’s method are shown in Figures 4.1 to 4.3 and the confusion matrices showing recognition rates are shown in Figures 4.4 to 4.6. The trends show that the WRANK-ELM gives the most stable response, with practically no fluctuations. Both algorithms are significantly smoother than Zhang et al.’s method. From the confusion matrices for Position *A* (Figure 4.4), we note that the ‘zero’ state is never classified as medium or high by WRANK-ELM and RIG-ELM, while this happens 2.7% of the time for Zhang et al.’s method. From Figure 4.1, we see that this happens close to midnight on Day 3. The ‘zero’ state is confused as ‘low’ by all three algorithms, but only 3.79% and 4.35% of the time by WRANK-ELM and RIG-ELM, compared to 17.83% of the time by Zhang et al.’s method. From Figure 4.1, we see that this happens at around 8 am and 11 pm every day, which corresponds to the beginning and end of the day’s occupancy. On Day 5, there is a prolonged ‘low’ detection between 6 am and 9 am by Zhang et al.’s method. This could lead to a significant wastage of energy. In comparison, the shorter duration false detections by the proposed algorithms can be easily addressed using a scheduled operation of the ACMV control.

The ‘low’ state is classified as ‘zero’ 5.13%, 7.46% and 8.51% of the time by the WRANK-ELM, RIG-ELM and Zhang et al.’s method respectively. From 4.1, we see that this happens only close to the end occupancy (around 11pm) for the WRANK-ELM. For the RIG-ELM, this repeats around 6 pm at Days 1,2,3 and 4. This time corresponds to the end of working hours of most research staff, leaving behind mostly PhD students who stay behind. The ACMV control system could be set to ignore the instantaneous drop from ‘low’ to ‘zero’ at this time, acting to switch off the air conditioning only if the ‘zero’ state crosses a certain threshold of time. The ‘high’ state is never confused as ‘zero’ by WRANK-ELM, while the RIG-ELM does this once at Day 3 around 10 am. Considering this to be a peak working hour, the control system can ignore such instantaneous false detections.

The confusion matrices for Position *B* (Figure 4.5) do not show a notable difference from Position *A*. As for Position *C*, the only notable difference is the slight drop in accuracy for the WRANK-ELM. Studying the confusion matrix in Figure 4.6, we see that this is largely due to the confusion of the ‘low’ state as ‘zero’. We see from the occupancy trend in Figure 4.3 that the disparity is confined to Day 4, where the WRANK-ELM falters between 8pm and 10pm. This could possibly be due to the localization effect arising from the corner placement of Position *C*. The sensors are placed in a narrow space between an occupant cubicle and a cubicle partition for another row of cubicles. Thus, the effect of this single occupant may affect the general trend of the entire room captured by the sensors. This is corroborated by the raw data trends for Day 4. For example, the temperature trends for Day 4 are shown in Figure 4.7. It is evident that the temperature readings for Position *C* are significantly offset compared to Positions *A* and *B*.

The computation times of the three algorithms are shown in Table 4.2. Both the proposed algorithms are considerably faster than Zhang et al.’s method, with the RIG-ELM being over 19 times faster. This is because the proposed algorithms do not require an exhaustive combinatorial analysis.

Presence/absence detection has important applications such as lighting control [12]. Table 4.6 shows the accuracy of presence and absence detection for the three algorithms and three sensor locations. The WRANK-ELM achieves a high accuracy of 96.88% for Position *A*, over 6% higher than Zhang et al.’s method. However, the accuracy is reduced at Position *C*, possibly due to the localization effect at the corner discussed earlier. The RIG-ELM achieves consistently good accuracies for all sensor positions.

Table 4.1: Occupancy level estimation accuracy and RMSE values for the three algorithms

	<i>Position A</i>		<i>Position B</i>		<i>Position C</i>	
	$A_L(\%)$	$RMSE$	$A_L(\%)$	$RMSE$	$A_L(\%)$	$RMSE$
WRANK-ELM	79.17	0.3678	78.75	0.3683	75.63	0.4452
RIG-ELM	76.88	0.4333	77.92	0.4128	77.08	0.4357
Zhang et al.	67.71	0.5332	67.50	0.5821	68.75	0.5585

Table 4.2: Computation times for WRANK-ELM, RIG-ELM and Zhang et al's method

	WRANK-ELM	RIG-ELM	Zhang et al.
<i>Time (seconds)</i>	130	23	466

4.2.2 Sensor combinations

In this section, we investigate the effect of combining sensor positions A , B and C . Four combinations are studied: AB , BC , AC and ABC . The occupancy level estimation accuracies, RMSE values and presence/absence detection accuracies are shown in Figures 4.12 to 4.14, respectively. The accuracy at AB is highest, but only negligibly more than A . Despite the slightly reduced accuracy of C , the combinations AC , BC and ABC remain high (close to 79%). This indicates that the sensor combinations mitigate the localization effect at C discussed earlier. The combinations represent an averaging of the raw data at different positions, which reduces the offset in the data at Position C illustrated earlier in Figure 4.7.

The confusion matrices for the sensor combinations are shown in Figures 4.15 to 4.18. For the ‘medium’ level, the WRANK-ELM achieves recognition rates of over 75%, while Zhang et al’s method is around 35%. For the ‘low’ level, the recognition rate of the RIG-ELM is as high as 91.26% (BC , Figure 4.17), while Zhang et al’s is 68.07%. To ensure energy savings, the confusion in classifying low levels as high levels should be minimized. In that regard, ABC is the most suitable combination, since the rate of classifying a ‘low’ as ‘medium’ and ‘high’ is the least.

Table 4.3: Occupancy level estimation accuracy and RMSE values for sensor combinations AC and BC

	<i>Positions AB</i>		<i>Position AC</i>	
	$A_L(\%)$	$RMSE$	$A_L(\%)$	$RMSE$
<i>WRANK – ELM</i>	79.79	0.3595	78.96	0.3729
<i>RIG – ELM</i>	77.50	0.4170	78.54	0.4149
<i>Zhang et al.</i>	66.25	0.5769	67.08	0.5409

Table 4.4: Occupancy level estimation accuracy and RMSE values for sensor combinations BC and ABC

	<i>Position BC</i>		<i>Positions ABC</i>	
	$A_L(\%)$	$RMSE$	$A_L(\%)$	$RMSE$
<i>WRANK – ELM</i>	79.17	0.3619	78.96	0.3642
<i>RIG – ELM</i>	77.71	0.4151	76.46	0.4286
<i>Zhang et al.</i>	66.46	0.5569	66.67	0.5471

4.3 Occupancy number estimation

4.3.1 15-minute resolution

In this section, we present the results with full-range occupancy, i.e. the actual number of occupants. Since the number of occupants is high (up to 28), it is very difficult to classify the number of occupants exactly. Therefore, the accuracy A_L defined in Section 4.2 will not reflect the performance of the estimator well. For example, if $o_k = 20$, the estimate $\hat{o}_k = 21$ is considered completely incorrect when using A_L as an assessment criterion. Also, in many real applications like indoor air-conditioning or lighting control systems, distinguishing between single occupants would have no practical difference for the operation of the control system. Therefore, to better assess the performance of the occupancy estimator, we introduce the following *x-tolerance accuracy*:

$$\tau(\hat{\mathbf{o}}, x) := \frac{\sum_{k=1}^S X(|o_k - \hat{o}_k|, x)}{S} \quad (4.3)$$

where

$$X(|o_k - \hat{o}_k|, x) = \begin{cases} 1, & \text{if } |o_k - \hat{o}_k| \leq x \\ 0, & \text{otherwise.} \end{cases}$$

Table 4.5: Presence/absence detection accuracies for WRANK-ELM, RIG-ELM and Zhang et al's method

	<i>Position AB</i>	<i>Position AC</i>	<i>Position BC</i>	<i>Position ABC</i>
<i>WRANK – ELM</i>	96.46	96.67	96.46	96.25
<i>RIG – ELM</i>	95.21	95.63	96.67	95.83
<i>Zhang et al.</i>	92.50	91.46	91.04	89.58

Figure 4.1: Occupancy estimation results for the five testing days for sensor position A

We can easily find that the *accuracy* is a special case of the *x-tolerance accuracy*, i.e. $\tau(\hat{\mathbf{o}}, 0) = A_L$. The tolerance x should be properly selected based on the specific applications.

The occupancy trends for the three algorithms, along with the respective x -tolerance accuracies, are shown in Figures 4.19 to 4.24. The 3 and 4-tolerance accuracies are generally around 80% and 85% , respectively, for both the WRANK-ELM and RIG-ELM. The algorithms do track the occupancy profile reasonably well. However, the response experiences fluctuations, especially around 3pm when there are many abrupt variations in occupancy.

Figure 4.2: Occupancy estimation results for the five testing days for sensor position B

Table 4.6: Presence/absence detection accuracies for sensor positions A , B and C

	<i>Position A</i>	<i>Position B</i>	<i>Position C</i>
WRANK-ELM	96.88	96.67	93.54
RIG-ELM	95.63	94.58	95.42
Zhang et al.	90.42	92.50	92.92

4.3.2 1-minute resolution

In this section, we present the results for occupancy number estimation with $R = 1$. This is a very challenging scenario, since the maximum occupancy is high and the occupancy profile is complex. We present the results for sensor position A . The occupancy trends are shown in Figure 4.25. The 3 and 4 tolerance accuracies are 74.89% and 79.56% for the WRANK-ELM and 73.67 and 78.6 for the RIG-ELM. Both algorithms achieve better accuracy than Zhang et al.'s method. There are a number of fluctuations in the response. Table 4.7 compares the 3 and 4-tolerance accuracies for $R = 15$ and $R = 1$. There is an expected decline in the accuracies. However, these are reasonable results considering the complexity of the problem.

Figure 4.3: Occupancy estimation results for the five testing days for sensor position *C*

Table 4.7: Comparison of 3 and 4-tolerance accuracies (at sensor position *A*)

	15m		1m	
	x=3	x=4	x=3	x=4
WRANK-ELM	79.79	86.04	74.89	79.56
RIG-ELM	77.29	84.58	73.67	78.6
Zhang et al.	72.08	79.58	67.5	72.78

4.4 Summary

In this chapter, we presented experimental results for the WRANK-ELM and RIG-ELM. Both algorithms achieved excellent accuracies nearly 12% higher than past work. For presence detection, accuracies over 96% were obtained. Also, the computation times of the proposed algorithm were considerably smaller, over 19 times in case of RIG-ELM. It was seen that the corner location presented a slight reduction in accuracy, which was identified as being possibly caused by the close proximity of the adjacent occupant to the sensor. When the corner sensor data was combined with other sensors, this effect was mitigated. For the difficult scenario of occupancy number estimation, the response was reasonably accurate but showed fluctuations.

zero	94.87	3.79	0.00	0.00
low	5.13	81.06	21.43	0.80
medium	0.00	15.15	71.43	44.80
high	0.00	0.00	7.14	54.40
	zero	low	medium	high

(a) WRANK-ELM

zero	92.04	4.35	0.00	0.00
low	7.46	82.61	30.00	1.61
medium	0.00	13.04	60.00	45.97
high	0.50	0.00	10.00	52.42
	zero	low	medium	high

(b) RIG-ELM

zero	88.30	17.83	2.70	0.00
low	8.51	68.22	37.84	4.76
medium	2.66	9.30	40.54	50.79
high	0.53	4.65	18.92	44.44
	zero	low	medium	high

(c) Zhang et al.

Figure 4.4: Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (for sensor position A)

zero	94.85	4.48	0.00	0.00
low	4.64	80.60	21.43	0.81
medium	0.52	14.93	67.86	45.16
high	0.00	0.00	10.71	54.03
	zero	low	medium	high

(a) WRANK-ELM

zero	90.59	6.03	0.00	0.00
low	8.42	82.76	20.83	0.88
medium	0.50	11.21	66.67	43.86
high	0.50	0.00	12.50	55.26
	zero	low	medium	high

(b) RIG-ELM

zero	89.29	13.64	0.00	0.00
low	9.18	70.91	41.30	7.03
medium	1.53	11.82	34.78	50.00
high	0.00	3.64	23.91	42.97
	zero	low	medium	high

(c) Zhang et al.

Figure 4.5: Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (for sensor position B)

zero	89.16	7.44	0.00	0.00
low	9.85	80.17	17.39	2.26
medium	0.99	12.40	73.91	46.62
high	0.00	0.00	8.70	51.13
	zero	low	medium	high

(a) WRANK-ELM

zero	90.00	0.93	0.00	0.00
low	7.62	87.96	31.58	0.81
medium	2.38	11.11	55.26	46.77
high	0.00	0.00	13.16	52.42
	zero	low	medium	high

(b) RIG-ELM

zero	91.49	12.50	7.02	0.00
low	8.51	70.54	36.84	6.50
medium	0.00	12.50	40.35	47.97
high	0.00	4.46	15.79	45.53
	zero	low	medium	high

(c) Zhang et al.

Figure 4.6: Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (for sensor position C)

Figure 4.7: Comparison of temperature data for Positions A, B and C for test day 4

Figure 4.8: Occupancy estimation results for the five testing days for sensor positions AB

Figure 4.9: Occupancy estimation results for the five testing days for sensor positions *AC*

Figure 4.10: Occupancy estimation results for the five testing days for sensor positions *BC*

Figure 4.11: Occupancy estimation results for the five testing days for sensor positions ABC

Figure 4.12: Estimation accuracies for different sensor combinations

Figure 4.13: RMSE for different sensor combinations

Figure 4.14: Presence/absence detection accuracies for different sensor combinations

zero	94.36	4.44	0.00	0.00
low	5.13	80.74	14.29	0.82
medium	0.51	14.81	78.57	43.44
high	0.00	0.00	7.14	55.74
	zero	low	medium	high

(a) WRANK-ELM

zero	92.39	6.61	0.00	0.00
low	6.09	82.64	23.26	1.68
medium	1.02	10.74	62.79	45.38
high	0.51	0.00	13.95	52.94
	zero	low	medium	high

(b) RIG-ELM

zero	87.38	9.52	0.00	0.00
low	10.68	73.33	39.13	5.69
medium	1.94	10.48	28.26	55.28
high	0.00	6.67	32.61	39.02
	zero	low	medium	high

(c) Zhang et al.

Figure 4.15: Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (AB)

zero	95.31	5.26	0.00	0.00
low	4.69	81.95	14.81	1.56
medium	0.00	12.78	74.07	46.09
high	0.00	0.00	11.11	52.34
	zero	low	medium	high

(a) WRANK-ELM

zero	90.82	1.83	0.00	0.00
low	8.70	87.16	21.74	0.85
medium	0.48	11.01	65.22	44.92
high	0.00	0.00	13.04	54.24
	zero	low	medium	high

(b) RIG-ELM

zero	89.01	16.81	0.00	0.00
low	8.90	67.23	37.29	4.50
medium	0.52	10.92	40.68	52.25
high	1.57	5.04	22.03	43.24
	zero	low	medium	high

(c) Zhang et al.

Figure 4.16: Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (BC)

zero	94.82	5.26	0.00	0.00
low	5.18	81.20	17.86	0.79
medium	0.00	13.53	75.00	45.24
high	0.00	0.00	7.14	53.97
	zero	low	medium	high

(a) WRANK-ELM

zero	93.50	2.91	0.00	0.00
low	6.50	91.26	26.67	0.85
medium	0.00	5.83	55.00	48.72
high	0.00	0.00	18.33	50.43
	zero	low	medium	high

(b) RIG-ELM

zero	88.48	17.65	0.00	0.00
low	9.95	68.07	39.53	5.51
medium	1.57	10.08	34.88	51.97
high	0.00	4.20	25.58	42.52
	zero	low	medium	high

(c) Zhang et al.

Figure 4.17: Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (AC)

zero	93.88	4.38	0.00	0.00
low	5.61	79.56	13.04	0.81
medium	0.51	16.06	78.26	44.35
high	0.00	0.00	8.70	54.84
	zero	low	medium	high

(a) WRANK-ELM

zero	92.08	3.64	0.00	0.00
low	6.93	85.45	31.91	0.83
medium	0.50	10.91	53.19	47.93
high	0.50	0.00	14.89	51.24
	zero	low	medium	high

(b) RIG-ELM

zero	85.71	17.80	2.56	0.00
low	10.71	71.19	33.33	4.72
medium	2.55	8.47	35.90	52.76
high	1.02	2.54	28.21	42.52
	zero	low	medium	high

(c) Zhang et al.

Figure 4.18: Confusion matrix showing recognition rate percentages for WRANK-ELM, RIG-ELM and Zhang et al's method (ABC)

Figure 4.19: Occupancy estimation results for full-range occupancy for sensor position *A*

Figure 4.20: The x-tolerance accuracies for sensor position *A*

Figure 4.21: Occupancy estimation results for full-range occupancy for sensor position B

Figure 4.22: The x-tolerance accuracies for sensor position B

Figure 4.23: Occupancy estimation results for full-range occupancy for sensor position C

Figure 4.24: The x-tolerance accuracies for sensor position C

Figure 4.25: Occupancy estimation results for the five testing days for sensor positions AB

Figure 4.26: The x-tolerance accuracies for occupancy number estimation with $R = 1$ (at sensor position A)

Part II

Occupancy estimation using dynamic features

Chapter 5

Occupancy estimation using CO_2 -based dynamic features: FS-ELM

5.1 Introduction

In this chapter, an occupancy estimator that works with CO_2 -based dynamic features is presented. A variation of the Extreme Learning Machine (ELM) called *feature scaled ELM* (FS-ELM) is proposed. In Section 5.2, the dynamic CO_2 -based model used in this work is introduced. In Section 5.3, the basis and structure of the FS-ELM is explained. Then, in Section 5.4, a smoothing strategy for the raw data is presented to improve the performance of the FS-ELM.

5.2 Indoor Occupancy estimation from CO_2 data

Various dynamic models of the indoor CO_2 concentration have been discussed in the literature [30, 92, 54, 5, 11, 40]. In summary, we introduce the following generalized discrete-time state space model

$$c_k = g(\mathbf{c}_{k-l:k-1}, \mathbf{o}_{k-l:k}, \mathbf{v}_{k-l:k}) \quad (5.1)$$

where c_k is the CO_2 concentration around an indoor sensor node at time instant t_k , $g(\cdot)$ is an unknown function, and $\mathbf{c}_{k-l:k-1} = [c_{k-l}, c_{k-l+1}, \dots, c_{k-1}]^T$

is the sequence of CO₂ concentration at the past time horizon $[t_{k-l}, t_{k-1}]$. Similarly, $\mathbf{o}_{k-l:k} = [o_{k-l}, o_{k-l+1}, \dots, o_k]^T$ and $\mathbf{v}_{k-l:k} = [v_{k-l}, v_{k-l+1}, \dots, v_k]^T$ are the sequence of indoor occupants number and venting level, respectively.

Considering the CO₂ dynamic model (5.1), a generalized occupancy estimator can be described as

$$o_k = f(\mathbf{c}_{k-l:k}, \mathbf{o}_{k-l:k-1}, \mathbf{v}_{k-l:k}) = f(\mathbf{x}_k) \quad (5.2)$$

where $f(\cdot)$ is the model to be identified, and

$$\mathbf{x}_k = [\mathbf{c}_{k-l:k}^T \quad \mathbf{o}_{k-l:k-1}^T \quad \mathbf{v}_{k-l:k}^T]^T \in \mathbb{R}^n \quad (5.3)$$

is the input of the occupancy estimator. Here $n = 3l + 2$ is the dimension of the input vector. The occupancy estimator is a regression model between $\mathbf{x}_k$ and the number of current occupants o_k .

The main work of the estimator design is to identify the model $f(\cdot)$, which is a regression problem. With enough training samples, which can be measured off-line, many machine learning techniques like ANN [30, 31, 28], SVM [30, 28], ELM [16], and deep learning [62] can be used to solve this regression problem. In this work, we select the ELM due to its simplicity, computational efficiency, and flexibility. Next, we present a strategy to incorporate dynamic features into the ELM architecture.

5.3 FS-ELM for occupancy estimator identification

5.3.1 The standard ELM for occupancy estimation

When using the standard ELM, the input-to-hidden layer connection weights are randomly generated without tuning and are independent of the training data. This is why the ELM is computationally much more efficient than many other machine learning techniques in the training process. The general ELM structure was presented in Section 3.2.2. For the purpose of this discussion, we refer to the simplified structure in Fig. 5.1.

The output of the SLFN, i.e. the number of occupants, can be formulated

Figure 5.1: The structure of the standard extreme learning machine (ELM).

as

$$o_k = f(\mathbf{x}_k) = \sum_{i=1}^L \beta_i h(\mathbf{w}_i^T \mathbf{x}_k + b_i) \quad (5.4)$$

where L is the number of hidden neurons, $\mathbf{w}_i \in \mathbb{R}^n$ consists of the random weights from the input layer to the i -th hidden neuron, b_i is the random bias of the input of i -th hidden neuron, $h(\cdot)$ is a known activation function called *ELM random feature mapping* [46], and $\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_L]^T$ represents the hidden-to-output connection weights.

Figure 5.2: The sigmoid function.

As shown in Fig. 5.1, z_{ki} is the input of the activation function, and $z_{ki} = \mathbf{w}_i^T \mathbf{x}_k + b_i$. A commonly used activation function is the sigmoid function

$$h(z) = \frac{1}{1 + \exp(-z)}$$

which is shown in Fig. 5.2. We can find from Fig. 5.2 that $h(z_{ki}) \approx 0$ and $h(z_{ki}) \approx 1$ for all $z_{ki} < -5$ and $z_{ki} > 5$, respectively. Therefore, for any $\mathbf{x}_k$ which can guarantee $|z_{ki}| > 5$, the output of the hidden neuron will be 0 or 1, and the result is non-sensitive to the exact value of $\mathbf{x}_k$. In other words, ELM cannot distinguish the input layer data $\mathbf{x}_k$ if $\mathbf{x}_k$ can lead to $|z_{ki}| > 5$.

Let $\mathbf{z}_k = [z_{k1}, z_{k2}, \dots, z_{kL}]^T$, it is clear that

$$\mathbf{z}_k = \mathbf{W}\mathbf{x}_k + \mathbf{b} \quad (5.5)$$

where $\mathbf{W} = [\mathbf{w}_1 \ \mathbf{w}_2 \ \dots \ \mathbf{w}_L]^T$ and $\mathbf{b} = [b_1 \ b_2 \ \dots \ b_L]^T$. To make the hidden neurons more meaningful, i.e. to make sure that the output of hidden neurons can reflect the diversity of input layer data, we should guarantee that

$$\|\mathbf{z}_k\|_\infty < 5 \quad (5.6)$$

This requirement implies that the input-to-hidden layer connection weights are dependent on the training set, and thus they cannot be randomly set.

In addition, in our chosen dynamic model, the magnitude of each type of data may differ significantly from others. The indoor CO_2 concentration is typically from several hundred to a few thousand ppm (parts-per-million, i.e. the unit of CO_2 concentration), while the values of both the occupancy level and venting level are much smaller. In such a case, if the input-to-hidden weight matrix is randomly generated the influence of CO_2 concentration will overwhelm that of occupancy levels and venting levels, which would adversely affect the training of the regression model. Therefore, a scaling strategy suited to the ELM architecture is important. Also, since the raw data is noisy, features need to be extracted from the input data.

To address these requirements, we introduce two steps to modify the standard ELM: 1) add a feature layer between the input layer and the hidden layer, and 2) present a scaling strategy for the randomly generated feature-to-hidden layer connection weights. We termed the modified ELM as *feature scaled extreme learning machine* (FS-ELM). The structure of the proposed FS-ELM is shown in Fig. 5.3, and the details of the FS-ELM are shown below.

5.3.2 The feature layer of the FS-ELM

The feature layer extracts dynamic features from the input layer data, i.e. $\mathbf{x}_k$. For the FS-ELM, we define the following dynamic features:

- data: the sequence of CO_2 concentration in a past time horizon, i.e. $\mathbf{c}_{k-l:k}$.

Figure 5.3: The structure of the feature scaled extreme learning machine (FS-ELM).

- Integration of the differential: the sequence of integration of the first-order difference of CO_2 concentration in a time interval, i.e. $\mathbf{c}_{k-l+s:k}^I = \mathbf{c}_{k-l+s:k} - \mathbf{c}_{k-l:k-s}$.
- Difference of data: the sequence of first-order difference of CO_2 data, i.e. $\mathbf{c}_{k-l+1:k}^D = \mathbf{c}_{k-l+1:k} - \mathbf{c}_{k-l:k-1}$.

We call the three features as ‘PID of CO_2 data’. Actually both $\mathbf{c}_{k-l+s:k}^I$ and $\mathbf{c}_{k-l+1:k}^D$ are the difference of CO_2 data in the time domain, but the former is with a larger sampling time (s). Therefore, we can view the entries of $\mathbf{c}_{k-l+s:k}^I$ as the integration of the differential of CO_2 data in a time interval with length s .

The PID of CO_2 data together with the past occupancy level and the venting level constitute the feature layer, as shown in the second layer in Fig. 5.3. For simplicity, we denote the feature layer data as the vector

$$\mathbf{y}_k = [\mathbf{c}_{k-l:k}^T \quad (\mathbf{c}_{k-l+s:k}^I)^T \quad (\mathbf{c}_{k-l+1:k}^D)^T \quad \mathbf{o}_{k-l:k-1}^T \quad \mathbf{v}_{k-l:k}^T]^T \quad (5.7)$$

where $\mathbf{y}_k \in \mathbb{R}^{n_f}$ and $n_f = 5l - s + 3$. From (5.3) and (5.7) we can find that

$$\mathbf{y}_k = \mathbf{W}_1 \mathbf{x}_k = \text{blkdiag}\{\mathbf{W}_{\text{PID}}, \mathbf{I}_{2l+1}\} \mathbf{x}_k \quad (5.8)$$

where $\mathbf{I}_n$ is the n -dimensional identity matrix, $\text{blkdiag}\{\cdot\}$ is the block diagonal matrix operator, and

$$\mathbf{W}_{\text{PID}} = \begin{bmatrix} \mathbf{I}_{l+1} & \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_{l-s+1} \end{bmatrix} - \begin{bmatrix} \mathbf{I}_{l-s+1} \\ \mathbf{0} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_l \end{bmatrix} - \begin{bmatrix} \mathbf{I}_l \\ \mathbf{0} \end{bmatrix} \end{bmatrix}^T$$

where $\mathbf{0}$ represents a proper dimensional matrix with all zero entries.

5.3.3 The hidden layer of the FS-ELM

We can view the feature layer as a preprocessing of the input layer data, and the last three layers in Fig. 5.3 (i.e., the feature layer, the hidden layer and the output layer) are the same as the structure of the standard ELM. However, the feature-to-hidden layer connection weights of FS-ELM are not a random matrix. The input of the activation function is

$$\mathbf{z}_k = \mathbf{W}_2 \mathbf{y}_k + \mathbf{b} = \mathbf{R} \mathbf{S} \mathbf{y}_k + \mathbf{b} \quad (5.9)$$

where $\mathbf{W}_2 = \mathbf{R} \mathbf{S} \in \mathbb{R}^{L \times n_f}$ is the feature-to-hidden layer weight matrix, $\mathbf{R} \in \mathbb{R}^{L \times n_f}$ is a random matrix, and $\mathbf{S} \in \mathbb{R}^{n_f \times n_f}$ is a diagonal matrix. Here, $\mathbf{S}$ is unknown *a priori* and we can design it to scale $\mathbf{z}_k$ and guarantee the requirement of (5.6).

The output of the activation function, i.e. the output of hidden layer is

$$\mathbf{h}(\mathbf{z}_k) = [h(z_{k1}) \ h(z_{k2}) \ \cdots \ h(z_{kL})]$$

where the *ELM random feature mapping* $h(\cdot)$ is the sigmoid function.

5.3.4 FS-ELM training

The output of FS-ELM can be written as

$$o_k = \mathbf{h}(\mathbf{z}_k) \boldsymbol{\beta} = \mathbf{h}(\mathbf{R} \mathbf{S} \mathbf{W}_1 \mathbf{x}_k + \mathbf{b}) \boldsymbol{\beta} \quad (5.10)$$

where $\mathbf{W}_1$ is known *a priori*. Note that the entries of $\mathbf{b}$ and $\mathbf{R}$ are randomly generated. Therefore, the training process is to find the scale matrix $\mathbf{S}$ and the hidden-to-output layer weights $\boldsymbol{\beta}$.

Firstly, we determine the scale matrix. To meet the requirement (5.6), we can simply set

$$\mathbf{S} = \frac{5}{z'_{\max}} \mathbf{I}_{n_f} \quad (5.11)$$

where $z'_{\max} = \mathbf{max}\{\|\mathbf{z}'_k\|_{\infty}, k \leq N\}$, N is the number of training samples, and

$$\mathbf{z}'_k = \mathbf{R}\mathbf{y}_k = \mathbf{R}\mathbf{W}_1\mathbf{x}_k$$

Apparently, ignoring the bias vector $\mathbf{b}$, if $\mathbf{S} = \frac{5}{z'_{\max}} \mathbf{I}_{n_f}$ we can easily find that $\|\mathbf{z}_k\|_{\infty} \leq 5$ for all $k \leq N$. However, we can use the scale matrix $\mathbf{S}$ to adjust the influence of each feature. If such a diagonal matrix is carefully designed, we can obtain a better regression model.

We denote the scale matrix by

$$\mathbf{S} = \text{blkdiag}\{\alpha_P \mathbf{I}_{l+1}, \alpha_I \mathbf{I}_{l-s+1}, \alpha_D \mathbf{I}_l, \alpha_o \mathbf{I}_l, \alpha_v \mathbf{I}_{l+1}\} \quad (5.12)$$

where the parameters α_P , α_I , α_D , α_o , and α_v are positive scalable parameters, which can be tuned for better estimator design. If the scalable parameters

$$\boldsymbol{\alpha} = [\alpha_P \ \alpha_I \ \alpha_D \ \alpha_o \ \alpha_v]^T$$

are known, we can directly obtain $\mathbf{S}$.

Substituting (5.7) and (5.12) into (5.9) yields

$$\mathbf{z}_k = \mathbf{z}_k^P + \mathbf{z}_k^I + \mathbf{z}_k^D + \mathbf{z}_k^o + \mathbf{z}_k^v + \mathbf{b}$$

where

$$\begin{aligned} \mathbf{z}_k^P &= \alpha_P \mathbf{R}_1 \mathbf{c}_{k-l:k} \\ \mathbf{z}_k^I &= \alpha_I \mathbf{R}_2 \mathbf{c}_{k-l+s:k}^I \\ \mathbf{z}_k^D &= \alpha_D \mathbf{R}_3 \mathbf{c}_{k-l+1:k}^D \\ \mathbf{z}_k^o &= \alpha_o \mathbf{R}_4 \mathbf{o}_{k-l:k-1} \\ \mathbf{z}_k^v &= \alpha_v \mathbf{R}_5 \mathbf{v}_{k-l:k} \end{aligned}$$

and $\mathbf{R} = [\mathbf{R}_1 \ \mathbf{R}_2 \ \mathbf{R}_3 \ \mathbf{R}_4 \ \mathbf{R}_5]$. It is clear that the scalable parameters can adjust the influence of different features on $\mathbf{z}_k$. If all the five scalable parameters are equal like (5.11), the entries of $\mathbf{z}_k^P$ will overwhelm those of the other four components because the value of CO_2 data is much larger

than the data of the other features. In that case, compared with the CO₂ data the other four features are insignificant.

In practice, we use the following three steps to determine the scalable parameters:

1. Generate the random matrix $\mathbf{R}$ and calculate $z_{\max}^{P'}, z_{\max}^{I'}, z_{\max}^{D'}, z_{\max}^{o'}$, and $z_{\max}^{v'}$ from the training data. Here

$$z_{\max}^{P'} = \mathbf{max}\{\|\mathbf{z}_k^{P'}\|_{\infty} = \|\mathbf{R}_1 \mathbf{c}_{k-l:k}\|_{\infty}, k \leq N\}$$

and $\mathbf{z}_k^{P'} = \frac{1}{\alpha_P} \mathbf{z}_k^P$. Like $z_{\max}^{P'}$, we have the corresponding similar definitions of $z_{\max}^{I'}, z_{\max}^{D'}, z_{\max}^{o'}$, and $z_{\max}^{v'}$.

2. Set $z_{\max}^P, z_{\max}^I, z_{\max}^D, z_{\max}^o$, and $z_{\max}^v$ based on the following equation

$$z_{\max}^P + z_{\max}^I + z_{\max}^D + z_{\max}^o + z_{\max}^v < 5 \quad (5.13)$$

where $z_{\max}^P = \mathbf{max}\{\|\mathbf{z}_k^P\|_{\infty}, k \leq N\}$.

3. Evaluate the scalable parameters by $\alpha_s = \frac{z_{\max}^s}{z_{\max}^{s'}}$ where 's' corresponds to the subscripts/superscripts 'P', 'I', 'D', 'o', 'v', respectively.

Apparently, the five parameters in (5.13) stand for the influence of the five features on the hidden neurons (i.e., $\mathbf{z}_k$ and $\mathbf{h}(\mathbf{z}_k)$). In practice, we can tune the five parameters and check the effectiveness of the occupancy regression model via cross validation to find a group of effective $\boldsymbol{\alpha}$. With the tuned scalable parameters $\boldsymbol{\alpha}$ we can obtain the corresponding scale matrix $\mathbf{S}$ from (5.12). It is obvious that (5.11) is a special case of (5.12). On one hand the scale matrix in (5.12) scales $\|\mathbf{z}_k\|_{\infty}$ to a proper value (less than 5), on the other hand it balances the influence of different features.

Next, we determine the hidden-to-output layer connection weights. Considering the training data set $\{(\mathbf{x}_k, o_k) | k = 1, 2 \cdots N\}$, (5.10) can be rewritten as the following compact form:

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{O} \quad (5.14)$$

where $\mathbf{H}$ is the hidden layer output matrix

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}(\mathbf{z}_1) \\ \vdots \\ \mathbf{h}(\mathbf{z}_N) \end{bmatrix} = \begin{bmatrix} h(z_{11}) & \cdots & h(z_{1L}) \\ \vdots & \vdots & \vdots \\ h(z_{N1}) & \cdots & h(z_{NL}) \end{bmatrix}$$

and $\mathbf{O} = [o_1, o_2 \cdots o_N]^T$. With the designed scalable parameters, $\mathbf{H}$ can be easily calculated. Using the least-squares technique, the solution of (5.14) is given by $\boldsymbol{\beta} = \mathbf{H}^\dagger \mathbf{O}$

To improve the generalization performance and make the solution robust, a regularization term is added into to the least-squares solution. As seen from [47], based on the l_2 norm optimization equations, the solution of output weights becomes

$$\boldsymbol{\beta} = (\gamma \mathbf{I} + \mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{O} \quad (5.15)$$

where γ is a regularization parameter.

$\mathbf{W}_1$ is known *a priori*. With scalable parameters $\boldsymbol{\alpha}$ we can obtain $\mathbf{S}$. $\mathbf{R}$ and $\mathbf{b}$ are randomly generated and $\boldsymbol{\beta}$ can be obtained from (5.15). With $\mathbf{W}_1, \mathbf{R}, \mathbf{b}, \mathbf{S}$, and $\boldsymbol{\beta}$, from (5.10) we can find the occupancy estimator

$$o_k = f(\mathbf{x}_k) = \mathbf{h}(\mathbf{R} \mathbf{S} \mathbf{W}_1 \mathbf{x}_k + \mathbf{b}) \boldsymbol{\beta} \quad (5.16)$$

Like the standard ELM, in the training process the main computational cost is attributed to solving the inverse matrix in (5.15). Hence, the FS-ELM retains the computational efficiency of the standard ELM.

5.4 Smoothed CO_2 data for occupancy estimation

In this section, we present a smoothing strategy for the CO_2 data. Using the smoothing, we present occupancy estimation results for 1-minute resolution in Section 7.4.2.

5.4.1 Globally smoothed CO_2 data

It is shown in Fig. 5.4 that the measured CO_2 concentration suffers from spikes, which may be induced by the measurement noise, irregular indoor air movement and occupants' irregularly approaching the sensor. The influence of the irregular air and occupants movement on the CO_2 concentration of a certain location is still an open problem. Obviously, the measured CO_2 concentration with spikes cannot reflect the CO_2 level of the whole indoor space, which should be used for occupancy estimation. Therefore, to offset

the negative influence of the spikes in the measured CO_2 concentration, we smooth the measured data.

We denote all the measured CO_2 concentration data by a vector $\mathbf{c} = [c_1, c_2, \dots, c_N]^T$. The smoothed CO_2 data $\mathbf{c}_s$ can be obtained by minimizing the following energy function

$$J(\mathbf{c}_s) = \|\mathbf{c} - \mathbf{c}_s\|_2^2 + \lambda \|\nabla \mathbf{c}_s\|_2^2 \quad (5.17)$$

where $\nabla \mathbf{c}_s$ represents the gradient of $\mathbf{c}_s$,

$$\nabla = \begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix} \in \mathbb{R}^{N-1 \times N},$$

and λ is a weighting factor which controls the balance between the two terms in (5.17). Increasing λ results in more smoothing of the output $\mathbf{c}_s$.

Taking the derivative of $J(\mathbf{c}_s)$ in (5.17) with respect to $\mathbf{c}_s$ and setting it to zero yields

$$(\mathbf{I} + \lambda \Delta) \mathbf{c}_s = \mathbf{c} \quad (5.18)$$

where $\Delta = \nabla^T \nabla$ is a Laplacian matrix, and $\mathbf{I} + \lambda \Delta$ is a tridiagonal matrix. Hence (5.18) can be solved via the Thomas algorithm in $O(N)$ operations [23]. From (5.18), we can easily obtain the smoothed CO_2 concentration

$$\mathbf{c}_s = (\mathbf{I} + \lambda \Delta)^{-1} \mathbf{c} \quad (5.19)$$

The smoothed CO_2 concentration is shown in Fig. 5.4. Such a smoother is commonly used for image/signal edge-preserving smoothing [74, 17].

Using the smoothed CO_2 data we can train a new occupancy estimator via the FS-ELM introduced in Section III, which we denote by

$$o_k = f_s(\mathbf{x}_k^{\text{gs}}) = f_s(\mathbf{c}_{k-l:k}^{\text{gs}}, \mathbf{o}_{k-l:k-1}, \mathbf{v}_{k-l:k}) \quad (5.20)$$

where the superscript ‘gs’ means the ‘global smoothing’, which will be discussed later.

5.4.2 Locally smoothed CO_2 data

When using the identified occupancy estimator, $\mathbf{o}_{k-l:k-1}$, i.e. the exact number of occupants in the time horizon $[t_{k-l}, t_{k-1}]$, is generally unknown.

Figure 5.4: One day CO_2 concentration of the test-bed

In practice, we can use the estimated occupancy level $\hat{\mathbf{o}}_{k-l:k-1}$ instead of $\mathbf{o}_{k-l:k-1}$. Therefore, the practically used estimator (5.16) becomes

$$\hat{o}_k = f(\hat{\mathbf{x}}_k) = f(\mathbf{c}_{k-l:k}, \hat{\mathbf{o}}_{k-l:k-1}, \mathbf{v}_{k-l:k}) \quad (5.21)$$

Similarly, the occupancy estimator (5.20) trained by the globally smoothed CO_2 data becomes

$$\hat{o}_k^{\text{gs}} = f_s(\hat{\mathbf{x}}_k^{\text{gs}}) = f_s(\mathbf{c}_{k-l:k}^{\text{gs}}, \hat{\mathbf{o}}_{k-l:k-1}^{\text{gs}}, \mathbf{v}_{k-l:k}) \quad (5.22)$$

where $\mathbf{c}_{k-l:k}^{\text{gs}}$ consists of the $(k-l)$ -th to k -th entries of the smoothed CO_2 data

$$\mathbf{c}^{\text{gs}} = (\mathbf{I} + \lambda \Delta_M)^{-1} \mathbf{c}^{\text{u}} \quad (5.23)$$

Here, M implies the dimension of the the Laplacian matrix Δ_M , and $\mathbf{c}^{\text{u}} \in \mathbb{R}^M$ is measured CO_2 data for an entire day, which is unknown in real-time when using the estimator (5.22).

Equations (5.19) and (5.23) show that we need global information of the measured CO_2 data (i.e. all CO_2 data used for training or a whole day of CO_2 data for testing/using) to find the smoothed CO_2 concentration. In the process of training the occupancy estimator, all measured CO_2 data are *a priori* known and we can easily obtain the smoothed CO_2 data from (5.19). When using the occupancy estimator, however, the entire day's CO_2 data is unknown in real-time. Only the current and past CO_2 data are measured and future data are unknown. In real applications, only the locally smoothed CO_2 data can be obtained in real-time .

In practice, $\mathbf{c}^u$ and $\mathbf{c}^{gs}$ in (5.23) and $\mathbf{c}_{k-l:k}^{gs}$ in (5.22) is unavailable in real-time and hence we cannot use (5.22). An intuitive idea is to replace $\mathbf{c}_{k-l:k}^{gs}$ by the locally smoothed CO_2 data $\mathbf{c}_{k-l:k}^{ls}$, and the estimator is described as

$$\hat{o}_k^{ls} = f_s(\hat{\mathbf{x}}_k^{ls}) = f_s(\mathbf{c}_{k-l:k}^{ls}, \hat{\mathbf{o}}_{k-l:k-1}^{ls}, \mathbf{v}_{k-l:k}) \quad (5.24)$$

where $\mathbf{c}_{k-l:k}^{ls}$ consists of the $(k-l)$ -th to the k -th entries of the local smoothed CO_2 data

$$\mathbf{c}_{1:k}^{ls} = (\mathbf{I} + \lambda \Delta_k)^{-1} \mathbf{c}_{1:k} \quad (5.25)$$

It is shown in Fig. 5.4 that around 720 minutes, the measured CO_2 data has no spikes, and the black line, i.e., the locally smoothed 720 minutes CO_2 data, almost coincides with the globally smoothed CO_2 data. The green line in Fig. 5.4 shows that the locally smoothed 680 minutes CO_2 data is different from the globally smoothed data in the region near the end point, but the difference is smaller than the spikes around the region of 680 minutes. In practice, the end point exactly corresponds to the current time instant. It is not difficult to see from Fig. 5.4 that the locally smoothed data almost coincides with the globally smoothed data except the region near the end point, and the difference in this region is relatively small compared with the magnitude of the spikes. Therefore, considering the globally smoothed data is unavailable in real-time, we use the locally smoothed data instead.

However, directly using the estimator in (5.24) presents two problems. Firstly, the regression model $f_s(\cdot)$ is identified from the globally smoothed CO_2 data but the real-time locally smoothed CO_2 data is slightly different from the globally smoothed data. Secondly, $\hat{\mathbf{o}}_{k-l:k-1}^{ls}$ is a part of the input data to estimate $\hat{o}_k^{ls}$. If the estimated occupancy level $\hat{o}_{k-l:k-1}^{ls}$ is not accurate due to the difference between the locally and globally smoothed CO_2 data, the error will be transmitted to $\hat{o}_k^{ls}$. The estimation error is easily accumulated, and the accumulated error makes the results of (5.24) meaningless when k is large.

To remove the accumulated error, at the time instant t_k , we carry out the following three steps:

1. Evaluate the locally smoothed CO_2 data $\mathbf{c}_{1:k}^{ls}$ from (5.25).
2. Evaluate the occupancy level $\hat{o}_{l+1:k|k}^s$. For all $l+1 \leq i \leq k$,

$$\hat{o}_{i|k}^s = f_s(\mathbf{c}_{i-l:i|k}^s, \hat{\mathbf{o}}_{i-l:i-1|k}^s, \mathbf{v}_{i-l:i}) \quad (5.26)$$

where $\mathbf{c}_{i-l:i|k}^s$ consists of the $(i-l)$ -th to the i -th entries of $\mathbf{c}_{1:k}^{ls}$ obtained from step 1).

3. The estimated occupancy level at t_k is $\hat{o}_{k|k}^s$.

At each sampled time instant we need to estimate all the past occupancy levels. But in the process of estimating $\hat{o}_{k|k}^s$ we do not use $\{\hat{o}_{k-l|k-l}^s, \dots, \hat{o}_{k-1|k-1}^s\}$. Therefore, the estimation error cannot be transmitted and we can avoid the accumulation of the estimation error.

5.5 Summary

In this chapter, a novel ELM architecture to estimate occupancy from dynamic features was presented. The FS-ELM has two main differences with the standard ELM: 1) the FS-ELM has one more layer, i.e. the feature layer, which extracts dynamic features from the input layer data; and 2) the random hidden layer weight matrix is scaled based on the input layer data. The proposed FS-ELM retains the computational efficiency of the standard ELM. Later in this work, we present experimental results to validate the FS-ELM.

Chapter 6

Feature selection with the Hybrid Feature-Scaled ELM (HFS-ELM)

6.1 Introduction

In this chapter, the FS-ELM is extended to multiple parameters and features. We call this generalized structure HFS-ELM. The HFS-ELM is incorporated into a hybrid feature selection framework, which uses both static and dynamic features. In Section 6.2, the Hybrid Feature Selection Framework is elaborated on, as part of which we introduce the HFS-ELM in subsection 6.2.2. In Section 6.3, the features selected by the framework are studied.

6.2 Hybrid Feature Selection Framework

The aim of the feature selection framework is to shortlist a set of features that give the best occupancy estimates with a classifier. The proposed framework has two main components: a *filter* component, which uses a filter method of feature selection and a *wrapper* component, which implements a wrapper method of feature selection with a dynamic occupancy estimator. It is thus a kind of filter-wrapper hybrid. The framework is illustrated in Figure 6.1. The filter component extracts feature sets F_{P1} to F_{PN} corresponding to N parameters. These feature sets are ranked in order of their correlation with occupancy, which we measure using the RIG criterion [111] in this work. The best B features of each parameter are then combined into a single set. This set is then ranked using RIG to yield an ordered multi-parameter feature set.

This set is passed to the wrapper component, which transforms the features into their dynamic form (more details about dynamic features are provided later). The wrapper then searches for the combination of dynamic features that yields the best occupancy estimation accuracy. The components of the framework are explained in more detail in subsequent sections.

6.2.1 The filter component of the framework

6.2.1.1 Extraction of static features

By static features, we mean that the length of the sampling window $l = 1$. Therefore, the occupancy at time instant n is a function of the feature values at n only, without taking past values of the features into consideration. The advantage of static features is that they are small in size and can be processed with computationally light feature selection techniques. We define a set of static features $F_s = \{f_{gu}^p(n)\}$, where $g \in [1 F]$, $p \in [1 N_P]$ and u is the number of distinct values of a difference or step factor for feature group g . In this work, we define three groups of features (i.e. $F = 3$): the *first order shifted difference* (with N_s^p values of the difference ' s '), the *second order difference* (with N_t^p values of the difference ' t ') and *moving average* (with N_m^p values of the step ' m '). The definitions are shown in Table 6.1. In this work, we take $s_i^p = \{1\} \cup \{5\delta : \delta = [1 6]\}$, $t_j^p = \{1\} \cup \{5\delta : \delta = [1 6]\}$ and $m_k^p = \{5\delta : \delta = [1 6]\}$.

Table 6.1: Definitions of static features

#	Feature Descriptions
1	$f_{1i}^p(n) = raw(n) - raw(n - s_i^p); i \in [1 N_s^p]$
2	$f_{2j}^p(n) = raw(n) - 2raw(n - t_j^p) + raw(n - 2t_j^p); j \in [1 N_t^p]$
3	$f_{3k}^p(n) = (\sum_{\hat{n}=n-(m_k^p-1)}^n raw(\hat{n}))/m_k^p; k \in [1 N_m^p]$

6.2.1.2 RIG-based feature selection

The aim of the RIG-based feature selection is to reduce the set F_s to a smaller set of features. In order to do this, we rank features in order of their correlation with occupancy as measured by their RIG values. We split the feature selection into two levels. First, we create a set $F_B = \{F_B^{par}; par = [1 p]\}$, such that F_B^{par} consists of the best B features, in descending order of

Figure 6.1: Hybrid feature selection framework for dynamic occupancy estimation

their RIG values, for parameter par . Second, we rank the features of F_B to obtain the set $\bar{F}_B$. The set $\bar{F}_B$ is thus a multiparameter set of $B * p$ features ranked in order of their RIG values.

The RIG-based feature selection significantly reduces the size of the features passed on to the wrapper component of the framework. This greatly reduces the overall computation time of the feature selection framework. Originally, the total number of static features, including the raw data, is $p + \sum_{p=1}^{N_P} (N_s^p + N_t^p + N_m^p)$. The RIG-based feature selection reduces the number of features to $B * p$. Thus, the reduction is $(1 - \frac{B * p}{p + \sum_{p=1}^{N_P} (N_s^p + N_t^p + N_m^p)}) * 100\%$, which for the feature set defined in Section 6.2.1.1 is 85.71%.

6.2.2 The wrapper component of the framework: Hybrid Feature Scaled Extreme Learning Machine (HFS-ELM)

The HFS-ELM is a generalized version of the FS-ELM presented in Chapter 5 which can work with multiple parameters and features. Combined with the hybrid feature selection framework, the HFS-ELM offers a reliable method to study the relative worth of parameters and features for occupancy estimation.

The architecture of the HFS-ELM is shown in Figure 6.2. The various layers of the HFS-ELM are described below.

1) The input layer:

The input of the HFS-ELM is defined as follows:

$$x_n = [P_{n-l:n}^{1T} \dots P_{n-l:n}^{N_P T} \ o_{n-l:n-1}^T \ v_{n-l:n}^T]^T \quad (6.1)$$

where P refers to the raw data of an environmental parameter, o is the occupancy and v refers to the venting levels. The data at every time instant n is sampled within a past time horizon defined by l .

2) The feature layer:

The feature layer extracts dynamic features from the raw data input. By dynamic features, we mean that the features are sampled at every instant n within a past time window of size l . Henceforth, we deal with the set of $B * p$ features selected by the RIG filter. We use the bar notation to indicate

Table 6.2: Definitions of dynamic features

#	Feature	Description
1	First order shifted difference	$f_{1_{\Delta_{1i}^p}}^p = P_{n-l+\bar{s}_i^p:n} - P_{n-l:n-\bar{s}_i^p}$ $\Delta_{1i}^p = n - l + \bar{s}_i^p : n; i \in [1 \bar{N}_s^p]$
2	Second order difference	$f_{2_{\Delta_{2j}^p}}^p = P_{n-l+\bar{t}_j^p:n} - 2P_{n-l+\bar{t}_j^p:n-\bar{t}_j^p} + P_{n-l:n-l+\bar{t}_j^p}$ $\Delta_{2j}^p = n - l + 2\bar{t}_j^p : n; j \in [1 \bar{N}_t^p]$
3	m-step moving average	$\bar{f}_{3_{\Delta_{3k}^p}}^p = \sum_{v=0}^{\bar{m}_k^p-1} P_{n-(l-\bar{m}_k^p+v+1):n-v};$ $\Delta_{3k}^p = n - (l - \bar{m}_k^p + 1) : n; k \in [1 \bar{N}_m^p]$

the features that have passed through the filter component. The definitions of the dynamic features are shown in Table 6.2.

Let $\{\bar{s}_i^p\}, i \in [1 \bar{N}_s^p]$ be the set of difference values of the first order shifted difference feature for the parameter p . Let $\{\bar{t}_j^p\}, j \in [1 \bar{N}_t^p]$ be the set of difference values of the second order difference feature. Let $\{\bar{m}_k^p\}, k \in [1 \bar{N}_m^p]$ be the set of step values of the moving average feature. We define the following feature selection matrices to extract the dynamic features from the raw data.

$$W_{raw}^p = I_{l+1} \quad (6.2)$$

$$\begin{aligned}
W_{sd}^p = & \left[\begin{bmatrix} \mathbf{0} \\ I_{l-\bar{s}_1^p+1} \end{bmatrix} - \begin{bmatrix} I_{l-\bar{s}_1^p+1} \\ \mathbf{0} \end{bmatrix} \dots \right. \\
& \left. + \begin{bmatrix} \mathbf{0} \\ I_{l-\bar{s}_i^p+1} \end{bmatrix} - \begin{bmatrix} I_{l-\bar{s}_i^p+1} \\ \mathbf{0} \end{bmatrix} \right] \quad (6.3)
\end{aligned}$$

$$\begin{aligned}
W_{fd}^p = & \left[\begin{bmatrix} \mathbf{0} \\ I_{l-2\bar{t}_1^p+1} \end{bmatrix} - 2 \begin{bmatrix} I_{l-2\bar{t}_1^p+1} \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} I_{l-2\bar{t}_1^p+1} \\ \mathbf{0} \end{bmatrix} \dots \right. \\
& \left. + \begin{bmatrix} \mathbf{0} \\ I_{l-2\bar{t}_j^p+1} \end{bmatrix} - 2 \begin{bmatrix} \mathbf{0} \\ I_{l-2\bar{t}_j^p+1} \end{bmatrix} - \begin{bmatrix} I_{l-2\bar{t}_j^p+1} \\ \mathbf{0} \end{bmatrix} \right] \quad (6.4)
\end{aligned}$$

Figure 6.2: Structure of the HFS-ELM

$$W_{avg}^p = \left[\left[\begin{array}{c} \mathbf{0} \\ I_{l-\bar{m}_1^p+2} \end{array} \right] + \left[\begin{array}{c} \mathbf{0} \\ I_{l-\bar{m}_1^p+2} \\ \mathbf{0} \end{array} \right] + \dots \left[\begin{array}{c} I_{l-\bar{m}_1^p+2} \\ \mathbf{0} \end{array} \right] \right] \dots \\ + \left[\left[\begin{array}{c} \mathbf{0} \\ I_{l-\bar{m}_k^p+1} \end{array} \right] + \left[\begin{array}{c} \mathbf{0} \\ I_{l-\bar{m}_k^p+1} \\ \mathbf{0} \end{array} \right] + \dots \left[\begin{array}{c} I_{l-\bar{m}_k^p+1} \\ \mathbf{0} \end{array} \right] \right] \right] \quad (6.5)$$

$$W_p = [W_{raw}^p \ W_{sd}^p \ W_{fd}^p \ W_{avg}^p] \quad (6.6)$$

where p is the parameter number. For $p = [1 \ N_p]$, the combined feature extraction matrix is

$$W = blkdiag\{W_1, W_2 \dots W_{N_p}\} \quad (6.7)$$

The output of the feature layer is

$$y_n = [[P_{n-l:n}^{1T} \quad [(\bar{f}_{1_{\Delta_{11}^1}}^1)^T \dots (\bar{f}_{1_{\Delta_{1i}^1}}^1)^T] \dots [(\bar{f}_{F_{\Delta_{F1}^1}}^1)^T \dots \\ (\bar{f}_{F_{\Delta_{Fk}^1}}^1)^T]] \dots [P_{n-l:n}^{N_P T} \quad [(\bar{f}_{1_{\Delta_{11}^{N_P}}}^{N_P})^T \dots \\ (\bar{f}_{1_{\Delta_{1i}^{N_P}}}^{N_P})^T] \dots [(\bar{f}_{F_{\Delta_{F1}^{N_P}}}^{N_P})^T \dots (\bar{f}_{F_{\Delta_{Fk}^{N_P}}}^{N_P})^T]] \\ o_{n-l:n-1}^T \ v_{n-l:n}^T]^T \quad (6.8)$$

From (6.1) and (8), we can infer that

$$y_n = blkdiag\{W, I_{2l+1}\}x_n \quad (6.9)$$

3) The hidden layer:

The output of the hidden layer is

$$\mathbf{h}(\mathbf{z}_n) = [h(\mathbf{z}_{n1}) \ h(\mathbf{z}_{n2}) \dots h(\mathbf{z}_{nL})] \quad (6.10)$$

where the input $\mathbf{z}_n$ of the activation function is expressed as:

$$\mathbf{z}_n = \mathbf{W}_h \mathbf{y}_n + \mathbf{b} \quad (6.11)$$

where $\mathbf{W}_h = \mathbf{R}\mathbf{S}$ is the feature-to-hidden layer weight matrix. Here, $\mathbf{R}$ is a random matrix and $\mathbf{S}$ is a diagonal matrix, which can be designed in accordance with the relative magnitude of the input parameters. The matrix S balances the influence of different features. We denote the scalable parameters of S as α_{raw} , α_{sd} , α_{fd} and α_{mavg} and define the components of S as follows:

$$\Lambda_{raw}^p = \alpha_{raw} I_{l+1} \quad (6.12)$$

$$\Lambda_{sd}^p = blkdiag\{\alpha_{sd} I_{l-\bar{s}_1^p+1}, \dots, \alpha_{sd} I_{l-\bar{s}_i^p+1}\} \quad (6.13)$$

$$\Lambda_{fd}^p = blkdiag\{\alpha_{fd} I_{l-2\bar{t}_1^p+1}, \dots, \alpha_{fd} I_{l-2\bar{t}_j^p+1}\} \quad (6.14)$$

$$\Lambda_{mavg}^p = blkdiag\{\alpha_{mavg} I_{l-\bar{m}_1^p+1}, \dots, \alpha_{mavg} I_{l-\bar{m}_k^p+1}\} \quad (6.15)$$

Then, the scaling matrix for parameter p is defined as:

$$S^p = blkdiag\{\Lambda_{raw}^p, \Lambda_{fd}^p, \Lambda_{sd}^p, \Lambda_{mavg}^p\} \quad (6.16)$$

The overall scaling matrix S is thus:

$$S = blkdiag\{S^1, S^2, \dots, S^{N_p}\} \quad (6.17)$$

To obtain S , we only need to define the scalable parameters α_{raw} , α_{sd} , α_{fd} and α_{mavg} . The scalable parameters need to only guarantee that $\|\mathbf{z}_n\| < 5$, since for all input values beyond 5, the sigmoid activation function outputs 1.

4) The output layer:

For the training data set $\{(x_n, o_n) | n = 1, 2, \dots, N\}$, the output of the HFS-ELM can be expressed as follows:

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{O} \quad (6.18)$$

where $\mathbf{H}$ is the hidden layer output matrix expressed as:

$$\mathbf{H} = \begin{bmatrix} ? \mathbf{h}(\mathbf{z}_1) \\ \vdots \\ ? \mathbf{h}(\mathbf{z}_N) \end{bmatrix} = \begin{bmatrix} h(z_{11}) & \cdots & h(z_{1L}) \\ \vdots & \vdots & \vdots \\ h(z_{N1}) & \cdots & h(z_{NL}) \end{bmatrix}$$

and $\mathbf{O} = [o_1, o_2, \dots, o_N]^T$. The least-squares solution of (6.18) is given by $\boldsymbol{\beta} = \mathbf{H}^\dagger \mathbf{O}$.

A regularization term can be added to the least-squares solution to make the solution robust and enhance the generalization performance [45]. The solution of output weights thus becomes

$$\boldsymbol{\beta} = (\gamma \mathbf{I} + \mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{O} \quad (6.19)$$

Finally, the occupancy estimator is obtained as follows:

$$o_n = \mathbf{h}(\mathbf{z}_n) \boldsymbol{\beta} = \mathbf{h}(\mathbf{R} \mathbf{S} \mathbf{W}_{in} \mathbf{x}_n + \mathbf{b}) \boldsymbol{\beta} \quad (6.20)$$

The HFS-ELM retains the computational efficiency of the standard ELM, since the computational cost can chiefly be attributed to solving the inverse matrix in (6.19).

6.3 Analysis of selected features

In this section, the feature combinations selected by the hybrid framework for sensor positions A , B and C , respectively are presented. These are shown in Table 6.3. Here, 'raw' refers to the raw value of the parameter, 'fd' refers to the first order shifted difference feature, 'sd' to the second order difference feature and 'm' refers to the moving average feature. The numbers refer to the difference values in the case of fd and sd and to the step value in the case of m . Note that there are two stages of feature selection, the first for each individual parameter and the second for the parameters in combination. The features in bold are those features that pass both stages of feature selection, thus constituting the final, 'best' feature set. It can be seen that for Positions A and B , only the CO_2 features are selected finally, while for Position C , CO_2 and RH features are selected. Temperature and pressure features are ultimately discarded by the algorithm, indicating that the influence of occupants on these parameters is compensated by the ACMV

system. Also note that largely the same feature sets are selected for each sensor position. In the next section, the occupancy estimation results with the selected features are presented.

Table 6.3: Feature combinations selected by the hybrid framework

	<i>Position A</i>	<i>Position B</i>	<i>Position C</i>
<i>CO₂</i>	<i>fd25,fd30,sd1</i>	<i>fd25,fd30,sd1</i>	<i>raw_CO2,fd25,fd30</i>
<i>RH</i>	<i>raw_RH, fd1, m30</i>	<i>raw_RH, fd1, fd5</i>	<i>raw_RH,fd1, fd5</i>
<i>Temp</i>	<i>raw_T, fd1, fd5</i>	<i>raw_T, fd1, fd5</i>	<i>raw_T, fd1, fd5</i>
<i>Press</i>	<i>raw_P, fd1, fd5</i>	<i>raw_P, fd1, fd5</i>	<i>raw_P, fd1, fd5</i>

6.4 Summary

In this chapter, we presented a novel feature selection framework for dynamic features. The framework is a filter-wrapper hybrid. The filter part was shown to reduce the initial feature set by 85.71%, so as to make the wrapper implementation feasible. As part of the wrapper component, we extended the FS-ELM to a generalized machine learning structure that can accommodate multiple parameters and features. We explained the architecture of the HFS-ELM in detail. The feature selection matrices of the HFS-ELM allow the extraction of a large and diverse set of features. Feature selection results showed that mostly CO_2 features were shortlisted by the framework, with relative humidity also being selected to a minor degree. In the next chapter, we validate the algorithms with experimental results.

Chapter 7

Results and Discussion: Part II

7.1 Introduction

In this section, the occupancy estimation results for FS-ELM and HFS-ELM are presented. In Section 7.2, the parameters selected for the FS-ELM and HFS-ELM are explained. In Section 7.3, the results for estimating occupancy levels at individual sensor positions and for sensor combinations are presented. In Section 7.4, the results for estimating actual occupancy numbers for two different resolutions of time are presented.

7.2 Parameter settings

We used the experimental dataset described in Section 3.3. For both ELM and FS-ELM, all entries of the random matrices $\mathbf{W}$ and $\mathbf{R}$ were set to follow uniform distribution in between -1 and 1, i.e. $w_{ij} \sim \mathcal{U}(-1, 1)$ and $r_{ij} \sim \mathcal{U}(-1, 1)$. All entries of the bias vector were set by $b_i \sim \mathcal{U}(-0.1, 0.1)$. All the required parameters in the training process are shown in Tables 7.1 and 7.2.

The scalable parameters α were determined based on the method provided in Section 5.3.4. We simply set the parameters $z_{\max}^P, z_{\max}^I, z_{\max}^D, z_{\max}^o$, and $z_{\max}^v$, which stand for the influence of the five features on the hidden neurons, as 1, 1, 1, 1, and 0.1, respectively. If we set a larger $z_{\max}^v$ such as 0.5, the solutions of the FS-ELM may have a jump after 10:00 PM (the ventilation system is turned off), like the results of the standard ELM, which will be mentioned later. If the random matrix $\mathbf{R}$ changed, the obtained scalable

parameters would slightly change. The scalable parameters in Table 7.1 are the approximated solutions based on one random matrix $\mathbf{R}$. We have checked that if we regenerated $\mathbf{R}$ whose entries $r_{ij} \sim \mathcal{U}(-1, 1)$, the parameters $z_{\max}^P, z_{\max}^I, z_{\max}^D, z_{\max}^O$, and $z_{\max}^V$ were still around 1, 1, 1, 1, and 0.1, respectively. The parameters for HFS-ELM are shown in Table 7.2. Note that L_{sel} is the number of hidden nodes during feature selection, while L_{estim} is the number of hidden nodes for the final estimation. The scaling factors denoted by the superscript '1' are for CO_2 and pressure values, since they are of a similar order of magnitude. Similarly, the scaling factors denoted by the superscript '2' are for temperature and relative humidity values. We note that all the parameters in Tables 7.1 and 7.2 can still be further tuned for better performance of the estimator.

Table 7.1: Parameter settings for HFS-ELM

Parameter	Value
l	4
L_{sel}	100
L_{estim}	450
γ	0.001
$[\alpha_{raw}^1, \alpha_{fd}^1, \alpha_{sd}^1, \alpha_{mavg}^1]$	[0.0001, 0.0012, 0.005, 0.0001]
$[\alpha_{raw}^2, \alpha_{fd}^2, \alpha_{sd}^2, \alpha_{mavg}^2]$	[0.001, 0.012, 0.05, 0.001]
$[\alpha_O, \alpha_V]$	[0.004, 0.008]

Table 7.2: Parameter settings for FS-ELM

Parameter	Value
l	4
s	1
L	450
γ	0.001
$[\alpha_P, \alpha_I, \alpha_D, \alpha_o, \alpha_v]$	[0.0001, 0.0012, 0.005, 0.004, 0.008]

7.3 Occupancy level estimation

7.3.1 Individual sensor positions

In this section, the results for occupancy level estimation are presented. We evaluate the results using the accuracy A_L and $RMSE$ (described in equations). We compare our approach with that of Zhang et al.[111], which uses

the RIG criterion to exhaustively search for the best combination of static features. To keep the computation time from becoming prohibitively large, we set the maximum combination size to six. To perform the estimation with Zhang et al.'s approach, we used the standard ELM with 450 hidden nodes.

The results for occupancy level estimation are summarized in Table 7.3. The FS-ELM and HFS-ELM achieve excellent accuracies of up to 82.29% (Position *C*), almost 20% higher than Zhang et al.'s method and over 25% higher than the standard ELM. The accuracies remain about the same for all sensor positions. The RMSE values remain below 0.46 for FS-ELM and HFS-ELM, while they are above 0.64 for Zhang et al.'s method and the standard ELM. In Section 4.2, we noted that in the corner positions, the influence of occupants is more localized. In case of static features only, this effect was seen in a reduction in accuracy at that position. However, as the results indicate, FS-ELM and HFS-ELM are not influenced by this phenomenon. This can be attributed to the fact that dynamic features take into consideration a large past window of time. Therefore, short-term fluctuations do not adversely affect the estimation.

The occupancy trends for the ground truth, HFS-ELM and Zhang et al.'s method are shown in Figures 7.1 to 7.3 for Positions *A*, *B* and *C*, respectively. The FS-ELM and HFS-ELM closely tracks the actual occupancy profile. Notably, there are no false zero detections in the case of the FS-ELM and HFS-ELM, while there are a number of such instances with Zhang et al.'s method and the standard ELM. Also, the tracking of the FS-ELM and HFS-ELM is quite smooth and practically free from fluctuations. In detecting the initial and final occupancies, there is sometimes a slight offset. From a control point of view, this could be easily worked around by scheduling the ACMV system to switch on and off at fixed times. Promisingly, short-range changes in occupancy level, such as for Day 3 at 3pm, are tracked well by the FS-ELM and HFS-ELM. Overall, as both the performance indices and the occupancy trends indicate, the performance of the proposed algorithms is smooth and consistent across different locations.

In Table 7.4, we present the accuracies for detection of presence/absence of occupants. This is an important criterion for applications such as lighting control [12]. The FS-ELM and HFS-ELM achieve excellent accuracies of 97.92% and 98.12%, respectively. The accuracies are consistent across the three sensor positions.

The computation time of the FS-ELM is 2 seconds, the HFS-ELM is approximately 17 seconds, while that of Zhang et al.'s method is approximately 7.5 minutes. The high efficiency of the proposed algorithms makes them highly suitable for real-time applications. The advantage of the HFS-ELM is that it can be extended to any type of parameter and can evaluate a large and diverse set of features.

Table 7.3: Occupancy level estimation accuracy and RMSE values for HFS-ELM, FS-ELM, Zhang et al.'s method and the standard ELM

	<i>Position A</i>		<i>Position B</i>		<i>Position C</i>	
	$A_L(\%)$	$RMSE$	$A_L(\%)$	$RMSE$	$A_L(\%)$	$RMSE$
HFS-ELM	81.46	0.4306	79.37	0.4541	82.29	0.4208
FS-ELM	81.25	0.4330	81.25	0.4402	82.29	0.4208
Zhang et al.	67.71	0.6724	67.50	0.6519	68.75	0.6471
Standard ELM	64.38	0.6755	57.29	0.6997	65.63	0.7188

Figure 7.1: Occupancy estimation results for the five testing days for sensor position A

7.3.2 Sensor combinations

In this section, we study the results with sensor combinations AB , BC , AC and ABC . The results are shown in Tables 7.5 and 7.6. For all the combinations, the accuracy is above 80%. The HFS-ELM achieves the highest

Figure 7.2: Occupancy estimation results for the five testing days for sensor position *B*

Table 7.4: Presence/absence detection accuracies for HFS-ELM, FS-ELM, Zhang et al.'s method and the standard ELM

	<i>Position A</i>	<i>Position B</i>	<i>Position C</i>
HFS-ELM	98.12	97.71	97.71
FS-ELM	97.92	97.71	97.71
Zhang et al.	90.42	92.50	92.92
Standard ELM	87.71	77.92	86.67

accuracy of 83.33% for *AC*. The occupancy trends for the sensor combinations are shown in Figures 7.4 to 7.7. The response for the FS-ELM and HFS-ELM is smooth and practically free of false zero detections, unlike Zhang et al.'s method and the standard ELM. The offset in estimating initial and final occupancies is small. This is corroborated by the presence/absence detection accuracies shown in Table 7.7. The accuracies are over 97% for the FS-ELM and HFS-ELM.

Figure 7.3: Occupancy estimation results for the five testing days for sensor position C

Table 7.5: Occupancy level estimation accuracy and RMSE values for HFS-ELM, FS-ELM, Zhang et al.'s method and the standard ELM

	AB		BC	
	$A_L(\%)$	$RMSE$	$A_L(\%)$	$RMSE$
HFS-ELM	80.63	0.4402	81.46	0.4306
FS-ELM	80.83	0.4449	82.29	0.4282
Zhang et al.	66.25	0.6708	66.46	0.6551
Standard ELM	51.46	0.7388	43.13	0.7826

7.4 Occupancy number estimation

7.4.1 15-minute resolution

In this section, we study the performance of the proposed algorithms for estimating actual occupancy numbers. We use the x-tolerance accuracy defined in Section 4.3 to measure the performance. The occupancy trends and x-tolerance accuracies for positions A , B and C are shown in Figures 7.8 to 7.13. Note that the maximum occupancy is 28 and the actual occupancy profile is complex for all testing days. We note that the zero tolerance accuracies are low for all algorithms. However, considering the 3 and 4-tolerance accuracies, the performance is remarkably good. The FS-ELM

Table 7.6: Occupancy level estimation accuracy and RMSE values for HFS-ELM, FS-ELM, Zhang et al.’s method and the standard ELM

	<i>AC</i>		<i>ABC</i>	
	$A_L(\%)$	$RMSE$	$A_L(\%)$	$RMSE$
HFS-ELM	83.33	0.4082	80.63	0.4402
FS-ELM	81.67	0.4354	81.04	0.4425
Zhang et al.	67.08	0.6739	66.67	0.6785
Standard ELM	68.13	0.6455	55.83	0.6967

Figure 7.4: Occupancy estimation results for the five testing days for *AB*

and HFS-ELM achieve 3 and 4 tolerance accuracies of around 90% and 95%, respectively, while Zhang et al.’s method is limited to around 75% and 82%, respectively. The standard ELM reaches a 4-tolerance accuracy of 79% for position *B*, but the response is quite unstable as is evident by the low performance at position *A*.

7.4.2 1-minute resolution

In this section, we study the estimation of occupancy numbers with 1-minute resolution. This is a very challenging scenario for our high-occupancy setting. At this resolution, the noise in the data is greater. Therefore, we apply the smoothing strategy presented in Section 5.4. The occupancy number estimation results for the FS-ELM without smoothing, with global smoothing

Figure 7.5: Occupancy estimation results for the five testing days for sensor position *BC*

Table 7.7: Presence/absence detection accuracies for HFS-ELM, FS-ELM, Zhang et al.'s method and the standard ELM for sensor combinations

	<i>AB</i>	<i>BC</i>	<i>AC</i>	<i>ABC</i>
HFS-ELM	97.29	97.29	97.29	97.71
FS-ELM	97.71	97.71	97.92	97.50
Zhang et al.	92.50	91.04	91.46	89.58
Standard ELM	74.58	64.58	88.75	78.75

and with local smoothing are presented in Figure 7.14. We show the results with sensor position *A* only, since it has been established that the variation in positions is small. As is seen from the figure, the tracking of the FS-ELM is notably improved by smoothing. The difference is most prominently seen beyond 9pm. The x-tolerance accuracies are shown in Figure 7.15. There is a significant difference between the accuracies with and without smoothing. While the global smoothing performs the best, it cannot be performed in real-time, since the entire day's is required. The practical option is to perform local smoothing.

Incorporating the local smoothing strategy into the HFS-ELM and FS-ELM, we compare the performance with Zhang et al.'s method and the standard ELM. As seen from the occupancy trend in Figure 7.16, the FS-ELM and HFS-ELM perform remarkably well, with 3-tolerance accuracies of 87.58%

Figure 7.6: Occupancy estimation results for the five testing days for sensor position *AC*

and 88.17% and 4-tolerance accuracies of 93.60% and 93.75%, respectively. In comparison, the response with Zhang et al.'s method and the standard ELM contains multiple fluctuations. Table 7.8 summarizes the results for $R = 15$ mins and $R = 1$ min at sensor position *A*. It is evident that the 3 and 4-tolerance accuracies do not decrease significantly with a change of resolution.

Table 7.8: Comparison of accuracies (tolerance 3 and 4) for 15 minutes resolution and 1 minute resolution (at sensor position *A*)

	15m		1m	
	x=3	x=4	x=3	x=4
HFS-ELM	89.79	95.42	88.17	93.60
FS-ELM	90.83	96.04	87.58	93.75
Zhang et al.	75	82.5	67.50	72.78
ELM	63.33	72.08	62.74	69.85

7.5 Summary

In this chapter, we presented experimental results for the FS-ELM and HFS-ELM. The occupancy level estimation accuracy with both algorithms exceeds

Figure 7.7: Occupancy estimation results for the five testing days for sensor position *ABC*

82%. For presence detection, accuracies over 97% were obtained. The algorithms far outperformed the past work and the standard ELM, while being computationally more efficient. The accuracies were consistently high for various sensor combinations. Notably, the algorithms perform very well under the difficult scenario of occupancy number estimation with a 1 minute resolution.

Figure 7.8: Occupancy number estimation results for full-range occupancy for $R=15$ (at sensor position A)

Figure 7.9: The x-tolerance accuracies for occupancy number estimation for $R=15$ (at sensor position A)

Figure 7.10: Occupancy number estimation results for full-range occupancy for $R=15$ (at sensor position B)

Figure 7.11: The x-tolerance accuracies for occupancy number estimation for $R=15$ (at sensor position B)

Figure 7.12: Occupancy number estimation results for full-range occupancy for $R=15$ (at sensor position C)

Figure 7.13: The x-tolerance accuracies for occupancy number estimation for $R=15$ (at sensor position C)

Figure 7.14: Occupancy number estimation for full-range occupancy for $R=1$ for FS-ELM without smoothing, with global smoothing and with local smoothing

Figure 7.15: The x-tolerance accuracies for occupancy number estimation for $R=1$ for FS-ELM without smoothing, with global smoothing and with local smoothing

Figure 7.16: Occupancy number estimation results for full-range occupancy for $R=1$

Figure 7.17: The x-tolerance accuracies for occupancy number estimation for $R=1$ (at sensor position A)

Chapter 8

Conclusion and Future works

8.1 Discussions and Summary

In this work, we addressed the problem of estimating the occupancy of an indoor space in real-time, which could be useful for demand-driven air conditioning, security and other building automation applications. From a detailed review of past works in Chapter 2, we determined that a variety of sensors have been used for occupancy estimation, but most of them tend to be intrusive to occupant privacy. Out of the non-intrusive methods reported in the literature, environmental sensors have shown promise. However, since their use is relatively new, there is great potential for improving the accuracy and computation time obtained by previous works. In the context of features extracted from environmental parameters, we identified that most past works worked within the paradigm of static features. Working within paradigm, we recognised the potential of wrapper and hybrid feature selection techniques to achieve better occupancy estimation results. Additionally, we noted the potential of making greater use of dynamic features for occupancy estimation.

In Chapter 3, we proposed two novel feature selection algorithms based on the static feature paradigm. One is a wrapper algorithm called WRANK-ELM, which uses the ELM classifier and an efficient two-stage search to identify the best combination of features. The hybrid algorithm is called RIG-ELM, which is a filter-wrapper that combines the RIG criterion with the ELM classifier to search for the best feature combination. To test the algorithms, we collected data through a test-bed of CO_2 , relative humidity, temperature and pressure sensors in an office space. These sensors are cost

effective, easy to install and are already present in most ACMV systems. The office is a high occupancy setting with a maximum of 28 occupants. To determine the ground truth occupancy, we proposed an efficient strategy using motion detection at the entrances of the office. With this experimental data, the proposed algorithms ranked CO_2 features to be the most relevant and pressure data to be the least relevant. With the selected features, the proposed algorithms achieved excellent accuracies of over 79% in case of the WRANK-ELM and over 77% in case of the RIG-ELM. In comparison to a past work, these accuracies are over 12% higher. Additionally, the efficient search strategy of the two algorithms ensures a fast computation time, over 19 times higher than the exhaustive search approach used in past works. Testing the results at different sensor locations, we noted that a corner placement had a slight adverse effect on the results of the WRANK-ELM, possibly due to the localization effect in that position, as indicated by the noise in the raw data. However, when combined with data from other sensors, this effect was mitigated.

In Chapters 5 and 6, we presented novel techniques to estimate occupancy from dynamic features. We proposed the FS-ELM, which is a novel machine learning approach that models the occupancy in terms of dynamic features extracted from CO_2 , venting levels and occupancy in a past time horizon. A scaling strategy was incorporated into the architecture of the FS-ELM to balance the relative influence of features. A smoothing strategy was also presented to smooth noisy input data in real-time, which was shown to notably improve the performance of the FS-ELM. It was also shown that the FS-ELM retains the computational efficiency of the standard ELM. We then extended the FS-ELM to accommodate multiple parameters and features into a novel machine learning framework called the HFS-ELM. The HFS-ELM can extract a diverse set of dynamic features as defined by its feature selection matrices. We then proposed a novel feature selection framework that incorporates the HFS-ELM into a filter-wrapper hybrid algorithm. Testing the HFS-ELM and FS-ELM on the experimental data, we achieved excellent accuracies for all the considered scenarios. Specifically, the response for the most challenging problem of occupancy number estimation at a 1 minute resolution was smooth, with 3 and 4-tolerance accuracies of 88% and 93%, respectively. In light of these excellent results, we make the case in this work for a greater use of dynamic features for occupancy estimation problems. Aside from the high accuracy, the main advantage of the proposed

hybrid framework that incorporates the HFS-ELM is the generalization to any type of parameter, feature and machine learning technique. Multiple parameters have been investigated by past works. The HFS-ELM offers a reliable method to assess any of these parameters.

It is evident from the results presented in Chapters 4 and 7 that the achieved accuracy is high for individual sensor locations, with only a marginal improvement achieved by combining the sensors. In the case of the FS-ELM and HFS-ELM, due to the negligible difference between the accuracies at the three sensor positions A, B and C , no outstanding advantage of one position over the other is evident from these results. However, it was seen in Chapter 4 that the WRANK-ELM experienced a slight decline in accuracy at Position C , which on studying the raw data was attributed to the close proximity of the occupant to the sensor in the corner location. At Positions A and B , the sensors are placed in a wider space and are thus less susceptible to noise from the adjacent occupants. For the specific venue considered in this work, Position A is the most non-intrusive to occupants. Additionally, it is evident from Sections 3.5 and 6.3 that pressure data is not significantly correlated with occupancy. Thus, one set of the integrated CO_2 , temperature and humidity sensor placed centrally in the office offers a good solution for occupancy estimation. The integrated sensor used in this work (Rotronic CL11) costs 355 USD. For commercial implementation, the algorithms need to be ported to the lowest-cost microprocessors, which can be investigated in future work.

8.2 Extensions and Future Work

We suggest four key areas to extend this work in the future. Firstly, the general framework of the proposed algorithms can be incorporated into various machine learning architectures aside from the ELM. For instance, for the HFS-ELM, the feature layer can be manipulated to be the input of SVMs or decision trees. Additionally, deep learning networks have gained popularity in recent years. The deep structure of these networks can automatically generate features. However, the computation time of these networks can be long. These issues can be investigated in the context of occupancy estimation.

Secondly, future studies could focus on developing features that take the

unique characteristics of the parameters into consideration. For example, the time constant of the response of CO_2 to changes in occupancy can be used to devise features. Also, other methods to measure the merit of features can be investigated, such as minimum Relevance and Maximum Redundancy (mRMR).

Thirdly, the methods proposed in this work can be tested in even larger venues such as lecture theaters and auditoriums. The energy wastage that results from binary estimates can be particularly high in such venues. A study of the specific challenges of these venues, such as the high peak loads, optimal sensor selection and sensor placement are useful avenues to investigate.

Fourthly, the integration of the proposed methods with demand-driven ACMV control systems needs to be investigated. Presently, BEMS rarely take into account more than binary occupancy estimates in their controls. Additionally, the occupancy estimation system can be expanded to recognize user activities to further the intelligence of the control system. The capability of the system can be further increased to zone-based detection to enable zone-specific air conditioning. Another promising avenue is to integrate occupancy estimation with occupant feedback to ensure maximum comfort. Also, by testing with real systems, a true gauge of the energy savings possible with occupancy-driven control can be made.

Author's publications

Journal articles

1. Mustafa K.Masood, Yeng Chai Soh and Chaoyang Jiang. "Occupancy estimation from environmental parameters using wrapper and hybrid feature selection", accepted by *Applied Soft Computing*.
2. Mustafa K.Masood, Chaoyang Jiang and Yeng Chai Soh. "A novel feature selection framework with Hybrid Feature-Scaled Extreme Learning Machine (HFS-ELM) for indoor occupancy estimation", submitted to *Energy and Buildings*.
3. Chaoyang Jiang, Mustafa K. Masood and Yeng Chai Soh. "Indoor occupancy estimation from carbon dioxide concentration", *Energy and Buildings* 131 (2016): 132-141

Other Journal articles (not included in thesis)

1. Zhenghua Chen, Mustafa K. Masood and Yeng Chai Soh. "A fusion frame-work based occupancy estimation with environmental parameters in office buildings", *Energy and Buildings* 133 (2016): 790-798.
2. Zhenghua Chen, Qingchang Zhu, Mustafa K. Masood and Yeng Chai Soh. "Environmental Sensors based Occupancy Estimation in Buildings via IHMM-MLR", accepted by *IEEE Transactions on Industrial Informatics*.
3. Zhenghua Chen, Qingchang Zhu, Mustafa K. Masood and Yeng Chai Soh. "Building Occupancy Estimation with Environmental Sensors via CDBLSTM", accepted by *IEEE Transactions on Industrial Electronics*.
4. Qingchang Zhu, Zhenghua Chen, Mustafa K. Masood and Yeng Chai Soh. "Occupancy estimation with environmental sensing via non-iterative LRF feature learning in time and frequency domains", accepted by *Energy and Buildings*.

Bibliography

- [1] R. a. A. C. E. American Society of Heating. Ashrae standard 90.1, 1989.
- [2] Kaiser Ahmed, Jarek Kurnitski, and Piia Sormunen. Demand controlled ventilation indoor climate and energy performance in a high performance building with air flow rate controlled chilled beams. *Energy and Buildings*, 109:115–126, DEC 15 2015.
- [3] Bing Ai, Zhaoyan Fan, and Robert X. Gao. Occupancy Estimation for Smart Buildings by an Auto-Regressive Hidden Markov Model. In *2014 American Control Conference (ACC)*, Proceedings of the American Control Conference, 2014. American Control Conference, Portland, OR, DEC 04-06, 2014.
- [4] Manar Amayri, Abhay Arora, Stephane Ploix, Sanghamitra Bandhyopadhyay, Quoc-Dung Ngo, and Venkata Ramana Badarla. Estimating occupancy in heterogeneous sensor environment. *Energy and Buildings*, 129:46–58, OCT 1 2016.
- [5] Guillaume Ansanay-Alex. Estimating occupancy using indoor carbon dioxide concentrations only in an office building: a method and qualitative assessment. In *Proceedings of the 11th REHVA World Congress-Energy Efficient, Smart and Healthy Buildings (Prague, Czech Republic, June 16–19, 2013). Clima STP, Prague, Czech*, 2013.
- [6] ANSI/ASHRAE. Standard 62.1-2010 ventilation for acceptable indoor air quality, 2010.
- [7] Javier Apolloni, Guillermo Leguizamón, and Enrique Alba. Two hybrid wrapper-filter feature selection algorithms applied to high-dimensional microarray experiments. *Applied Soft Computing*, 38(C):922–932, January 2016.

- [8] B. M. Aslahi-Shahri, R. Rahmani, M. Chizari, A. Maralani, M. Es-lami, M. J. Golkar, and A. Ebrahimi. A hybrid method consisting of GA and SVM for intrusion detection system. *Neural Computing & Applications*, 27(6):1669–1676, AUG 2016.
- [9] Bharathan Balaji, Jian Xu, Anthony Nwokafor, Rajesh Gupta, and Yuvraj Agarwal. Sentinel: occupancy based hvac actuation using existing wifi infrastructure within commercial buildings. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, page 17. ACM, 2013.
- [10] Yannick Benezeth, Helene Laurent, Bruno Emile, and Christophe Rosenberger. Towards a sensor for detecting human presence and characterizing activity. *Energy and Buildings*, 43(2):305–314, 2011.
- [11] Davide Cali, Peter Matthes, Kristian Huchtemann, Rita Streblow, and Dirk Muller. Co 2 based occupancy detection algorithm: Experimental analysis and validation for office and residential buildings. *Building and Environment*, 86:39–49, 2015.
- [12] Luis M. Candanedo and Veronique Feldheim. Accurate occupancy detection of an office room from light, temperature, humidity and CO2 measurements using statistical learning models. *Energy and Buildings*, 112:28–39, JAN 15 2016.
- [13] Miguel Casals, Marta Gangoells, Nuria Forcada, and Marcel Macarulla. Reducing lighting electricity use in underground metro stations. *Energy Conversion and Management*, 119:130–141, JUL 1 2016.
- [14] Joel Chaney, Edward Hugh Owens, and Andrew D. Peacock. An evidence based approach to determining residential occupancy and its role in demand response management. *Energy and Buildings*, 125:254–266, AUG 1 2016.
- [15] Dong Chen, Sean Barker, Adarsh Subbaswamy, David Irwin, and Prashant Shenoy. Non-intrusive occupancy monitoring using smart meters. In *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, pages 1–8. ACM, 2013.

- [16] Hui-Ling Chen, Gang Wang, Chao Ma, Zhen-Nao Cai, Wen-Bin Liu, and Su-Jing Wang. An efficient hybrid kernel extreme learning machine approach for early diagnosis of parkinson's disease. *Neurocomputing*, 184:131–144, 2016.
- [17] Choongsang Cho and Sangkeun Lee. Effective five directional partial derivatives-based image smoothing and a parallel structure design. *IEEE Transactions on Image Processing*, 25(4):1617–1625, 2016.
- [18] Ken Christensen, Ryan Melfi, Bruce Nordman, Ben Rosenblum, and Raul Viera. Using existing network infrastructure to estimate building occupancy and control plugged-in devices in user workspaces. *International Journal of Communication Networks and Distributed Systems*, 12(1):4–29, 2014.
- [19] TM Chung and John Burnett. On the prediction of lighting energy savings achieved by occupancy sensors. *Energy engineering*, 98(4):6–23, 2001.
- [20] Giorgio Conte, Massimo De Marchi, Alessandro Antonio Nacci, Vincenzo Rana, and Donatella Sciuto. Bluesentinel: a first approach using ibeacon for an energy efficient occupancy detection system. In *BuildSys@ SenSys*, pages 11–19, 2014.
- [21] Andrea Corna, L Fontana, AA Nacci, and Donatella Sciuto. Occupancy detection via ibeacon on android devices for smart building management. In *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition*, pages 629–632. EDA Consortium, 2015.
- [22] Manoranjan Dash, Huan Liu, and Hiroshi Motoda. Consistency based feature selection. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 98–109. Springer, 2000.
- [23] Biswa Nath Datta. *Numerical linear algebra and applications*. Siam, 2010.
- [24] Christel de Bakker, Myriam Aries, Helianthe Kort, and Alexander Rosemann. Occupancy-based lighting control in open-plan office spaces: A state-of-the-art review. *Building and Environment*, 112:308–321, 2017.

- [25] Declan T Delaney, Gregory MP O'Hare, and Antonio G Ruzzelli. Evaluation of energy-efficiency in lighting systems using sensor networks. In *Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, pages 61–66. ACM, 2009.
- [26] Giovanni Diraco, Alessandro Leone, and Pietro Siciliano. People occupancy detection and profiling with 3d depth sensors for building energy management. *Energy and Buildings*, 92:246–266, 2015.
- [27] Robert H Dodier, Gregor P Henze, Dale K Tiller, and Xin Guo. Building occupancy detection through sensor belief networks. *Energy and buildings*, 38(9):1033–1043, 2006.
- [28] Bing Dong, Burton Andrews, Khee Poh Lam, Michael Hoyneck, Rui Zhang, Yun-Shang Chiou, and Diego Benitez. An information technology enabled sustainability test-bed (itest) for occupancy detection through an environmental sensing network. *Energy and Buildings*, 42(7):1038 – 1046, 2010.
- [29] David S Dougan and Len Damiano. Co2-based demand control ventilation: Do risks outweigh potential rewards? *ASHRAE journal*, 46(10):47, 2004.
- [30] Afrooz Ebadat, Giulio Bottegal, Damiano Varagnolo, Bo Wahlberg, and Karl H Johansson. Regularized deconvolution-based approaches for estimating room occupancies. *IEEE Transactions on Automation Science and Engineering*, 12(4):1157–1168, 2015.
- [31] Tobore Ekwevugbe, Neil Brown, Vijay Pakka, and Denis Fan. Real-time Building Occupancy Sensing Using Neural-Network Based Sensor Network. In *7TH IEEE International Conference on Digital Ecosystems and Technologies (DEST)*, IEEE International Conference on Digital Ecosystems and Technologies, pages 114–119, 2013.
- [32] Varick L Erickson, Stefan Achleitner, and Alberto E Cerpa. Poem: Power-efficient occupancy-based energy management system. In *Information Processing in Sensor Networks (IPSN), 2013 ACM/IEEE International Conference on*, pages 203–216. IEEE, 2013.
- [33] Varick L. Erickson, Miguel A. Carreira-Perpinan, and Alberto E. Cerpa. Occupancy Modeling and Prediction for Building Energy Management. *ACM Transactions on Sensor Networks*, 10(3), APR 2014.

- [34] Gabe Fierro, Omar Rehmane, Andrew Krioukov, and David Culler. Zone-level occupancy counting with existing infrastructure. In *Proceedings of the Fourth ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, pages 205–206. ACM, 2012.
- [35] Avgoustinos Filippoupolitis, William Oliff, and George Loukas. Occupancy detection for building emergency management using ble beacons. In *International Symposium on Computer and Information Sciences*, pages 233–240. Springer, 2016.
- [36] Anca D Galasiu, Guy R Newsham, Cristian Suvagau, and Daniel M Sander. Energy saving lighting control systems for open-plan offices: a field study. *Leukos*, 4(1):7–29, 2007.
- [37] Vishal Garg and NK Bansal. Smart occupancy sensors to reduce energy consumption. *Energy and Buildings*, 32(1):81–87, 2000.
- [38] Sunil Kumar Ghai, Lakshmi V Thanayankizil, Deva P Seetharam, and Dipanjan Chakraborty. Occupancy detection in commercial buildings using opportunistic context sources. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on*, pages 463–466. IEEE, 2012.
- [39] M. Gillott, L. T. Rodrigues, and C. Spataru. Low-carbon housing design informed by research. *Proceedings of the Institution of Civil Engineers - Engineering Sustainability*, 163(2):77–87, 2010.
- [40] Mattias Gruber, Anders Trüschel, and Jan-Olof Dalenbäck. Co 2 sensors for occupancy estimations: Potential in building automation applications. *Energy and Buildings*, 84:548–556, 2014.
- [41] X Guo, DK Tiller, GP Henze, and CE Waters. The performance of occupancy-based lighting control systems: A review. *Lighting Research & Technology*, 42(4):415–431, 2010.
- [42] Ebenezer Hailemariam, Rhys Goldstein, Ramtin Attar, and Azam Khan. Real-time occupancy detection using decision trees with multiple sensor types. In *Proceedings of the 2011 Symposium on Simulation for Architecture and Urban Design*, pages 141–148. Society for Computer Simulation International, 2011.

- [43] Mark A Hall. *Correlation-based feature selection for machine learning*. PhD thesis, The University of Waikato, 1999.
- [44] Hwataik Han, K Jang, Changho Han, and Junyong Lee. Occupancy estimation based on co2 concentration using dynamic neural network model. *Proc. AIVC*, 13, 2013.
- [45] G. B. Huang, H. Zhou, X. Ding, and R. Zhang. Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(2):513–529, April 2012.
- [46] Guang-Bin Huang. An insight into extreme learning machines: random neurons, random features and kernels. *Cognitive Computation*, 6(3):376–390, 2014.
- [47] Guang-Bin Huang, Hongming Zhou, Xiaojian Ding, and Rui Zhang. Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(2):513–529, 2012.
- [48] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: theory and applications. *Neurocomputing*, 70(1):489–501, 2006.
- [49] Oğuz İçoğlu and Ardeshir Mahdavi. Violas: A vision-based sensing system for sentient building models. *Automation in construction*, 16(5):685–712, 2007.
- [50] V.M. Janakiraman. *Machine Learning for Identification and Optimal Control of Advanced Automotive Engines*. PhD thesis, 2013.
- [51] Yifei Jiang, Xin Pan, Kun Li, Qin Lv, Robert P Dick, Michael Hannigan, and Li Shang. Ariel: Automatic wi-fi based room fingerprinting for indoor localization. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 441–450. ACM, 2012.
- [52] Ming Jin, Ruoxi Jia, Zhaoyi Kang, Ioannis C Konstantakopoulos, and Costas J Spanos. Presencesense: Zero-training algorithm for individual presence detection based on power monitoring. In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, pages 1–10. ACM, 2014.

- [53] G. Jothi and Hannah H. Inbarani. Hybrid Tolerance Rough Set-Firefly based supervised feature selection for MRI brain tumor image classification. *Applied Soft Computing*, 46:639–651, SEP 2016.
- [54] Swarnendu Kar and Pramod K Varshney. Accurate estimation of gaseous strength using transient data. *IEEE Transactions on Instrumentation and Measurement*, 60(4):1197–1205, 2011.
- [55] Aftab Khan, James Nicholson, Sebastian Mellor, Daniel Jackson, Karim Ladha, Cassim Ladha, Jon Hand, Joseph Clarke, Patrick Olivier, and Thomas Plötz. Occupancy monitoring using environmental and context sensors and a hierarchical analysis framework. In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, BuildSys '14, pages 90–99, New York, NY, USA, 2014. ACM.
- [56] Wilhelm Kleiminger, Christian Beckel, Thorsten Staake, and Silvia Santini. Occupancy detection from electricity consumption data. In *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, pages 1–8. ACM, 2013.
- [57] Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1):273–324, 1997.
- [58] Ilkka Korhonen, Juha Parkka, and Mark Van Gils. Health monitoring in the home of the future. *IEEE Engineering in medicine and biology magazine*, 22(3):66–73, 2003.
- [59] Stelios Krinidis, Georgios Stavropoulos, Dimosthenis Ioannidis, and Dimitrios Tzovaras. A robust and real-time multi-space occupancy extraction system exploiting privacy-preserving sensors. In *Communications, Control and Signal Processing (ISCCSP), 2014 6th International Symposium on*, pages 542–545. IEEE, 2014.
- [60] TM Labeodan, W Zeiler, G Boxem, and Y Zhao. Occupancy measurement in commercial office buildings for demand-driven control applications. *Energy and Buildings*, 2015.
- [61] Michael Lamb and Kenneth Sweeney. Reduced false alarm security system, 2017. US Patent 9,613,524.

- [62] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.
- [63] Nan Li, Gulben Calis, and Burcin Becerik-Gerber. Measuring and monitoring occupancy with an rfid based system for demand-driven hvac operations. *Automation in construction*, 24:89–99, 2012.
- [64] Chenda Liao, Yashen Lin, and Prabir Barooah. Agent-based and graphical modelling of building occupancy. *Journal of Building Performance Simulation*, 5(1):5–25, 2012.
- [65] Xingbin Lin and Josephine Lau. Demand controlled ventilation for multiple zone hvac systems: Co2-based dynamic reset (rp 1547). *HVAC&R Research*, 20(8):875–888, 2014.
- [66] Dixin Liu, Xiaohong Guan, Youtian Du, and Qianchuan Zhao. Measuring indoor occupancy in intelligent buildings using the fusion of vision sensors. *Measurement Science and Technology*, 24(7):074023, 2013.
- [67] Guopeng Liu, Jian Zhang, and A Dasu. Review of literature on terminal box control, occupancy sensing technology and multi-zone demand control ventilation (dcv). *US Department of Energy, Tech. Rep*, 2012.
- [68] L. James Lo and Atila Novoselac. Localized air-conditioning with occupancy control in an open office. *Energy and Buildings*, 42(7):1120–1128, JUL 2010.
- [69] Jiakang Lu, Tamim Sookoor, Vijay Srinivasan, Ge Gao, Brian Holben, John Stankovic, Eric Field, and Kamin Whitehouse. The smart thermostat: using occupancy sensors to save energy in homes. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 211–224. ACM, 2010.
- [70] Sawsan Mahmoud, Ahmad Lotfi, and Caroline Langensiepen. Behavioural pattern identification and prediction in intelligent environments. *Applied Soft Computing*, 13(4):1813–1822, APR 2013.
- [71] Sunil Mamidi, Yu-Han Chang, and Rajiv Maheswaran. Improving building energy efficiency with a network of sensing, learning and prediction agents. In *Proceedings of the 11th International Conference*

- on Autonomous Agents and Multiagent Systems-Volume 1*, pages 45–52. International Foundation for Autonomous Agents and Multiagent Systems, 2012.
- [72] Mustafa K Masood, Yeng Chai Soh, and Victor W-C Chang. Real-time occupancy estimation using environmental parameters. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2015.
- [73] Ryan Melfi, Ben Rosenblum, Bruce Nordman, and Ken Christensen. Measuring building occupancy using existing network infrastructure. In *Green Computing Conference and Workshops (IGCC), 2011 International*, pages 1–8. IEEE, 2011.
- [74] Dongbo Min, Sunghwan Choi, Jiangbo Lu, Bumsuh Ham, Kwanghoon Sohn, and Minh N Do. Fast global image smoothing based on weighted least squares. *IEEE Transactions on Image Processing*, 23(12):5638–5653, 2014.
- [75] Eldar Naghiyev, Mark Gillott, and Robin Wilson. Three unobtrusive domestic occupancy measurement technologies under qualitative review. *Energy and Buildings*, 69:507–514, FEB 2014.
- [76] M. Naseriparsa, A. Bidgoli, and T. Varae. A hybrid feature selection method to improve performance of a group of classification algorithms. *International Journal of Computer Applications*, 69(17):28–35, 2013.
- [77] Singapore National Environment Agency. *The energy efficiency revolution*. E2 Singapore, 2007.
- [78] Nam Tuan Nguyen, Rong Zheng, and Zhu Han. Umli: An unsupervised mobile locations extraction approach with incomplete data. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, pages 2119–2124. IEEE, 2013.
- [79] Kofi Nyarko and Cecelia Wright-Brown. Cloud based passive building occupancy characterization for attack and disaster response. In *Technologies for Homeland Security (HST), 2013 IEEE International Conference on*, pages 748–753. IEEE, 2013.
- [80] Kumar Padmanabh, Adi Malikarjuna V, Sougata Sen, Siva Prasad Kartu, Amrit Kumar, Sunil Kumar Vuppala, Sanjoy Paul, et al. isense:

- a wireless sensor network based conference room management system. In *Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings*, pages 37–42. ACM, 2009.
- [81] Jessen Page, Darren Robinson, Nicolas Morel, and J-L Scartezzini. A generalised stochastic model for the simulation of occupant presence. *Energy and buildings*, 40(2):83–98, 2008.
- [82] Ashish Pandharipande and David Caicedo. Daylight integrated illumination control of led systems based on enhanced presence sensing. *Energy and Buildings*, 43(4):944–950, 2011.
- [83] Sungmee Park and Sundaresan Jayaraman. Enhancing the quality of life through wearable technology. *IEEE Engineering in medicine and biology magazine*, 22(3):41–48, 2003.
- [84] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, 27(8):1226–1238, 2005.
- [85] J Ross Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014.
- [86] Rajib Rana, Brano Kusy, Josh Wall, and Wen Hu. Novel activity classification and occupancy estimation methods for intelligent hvac (heating, ventilation and air conditioning) systems. *Energy*, 93:245–255, 2015.
- [87] Sokwoo Rhee, Boo-Ho Yang, Kuowei Chang, and Hamhiko H Asada. The ring sensor: a new ambulatory wearable sensor for twenty-four hour patient monitoring. In *Engineering in Medicine and Biology Society, 1998. Proceedings of the 20th Annual International Conference of the IEEE*, volume 4, pages 1906–1909. IEEE, 1998.
- [88] R. Ruiz, J. C. Riquelme, J. S. Aguilar-Ruiz, and M. Garcia-Torres. Fast feature selection aimed at high-dimensional data via hybrid-sequential-ranked searches. *Expert Systems with Applications*, 39(12):11094–11102, 2012.
- [89] Seung Ho Ryu and Hyeun Jun Moon. Development of an occupancy prediction model using indoor environmental data based on machine learning techniques. *Building and Environment*, 107:1–9, OCT 2016.

- [90] Manaswi Saha, Shailja Thakur, Amarjeet Singh, and Yuvraj Agarwal. Energylens: combining smartphones with electricity meter for accurate activity detection and user annotation. In *Proceedings of the 5th international conference on Future energy systems*, pages 289–300. ACM, 2014.
- [91] M Schell and D Int-Hout. Demand control ventilation using CO₂. *ASHRAE Journal*, 43(2), FEB 2001.
- [92] Francesco Scotton, Lirong Huang, Seyed A Ahmadi, and Bo Wahlberg. Physics-based modeling and identification for hvac systems. In *European Control Conference*, 2013.
- [93] Marc Sebban and Richard Nock. A hybrid filter/wrapper approach of feature selection using information theory. *Pattern Recognition*, 35(4):835–846, 2002.
- [94] Huang-Chia Shih. Automatic Building Monitoring and Commissioning Via Human Behavior Recognition. In *2016 IEEE 5TH Global Conference on Consumer Electronics*. IEEE, 2016. 5th IEEE Global Conference on Consumer Electronics, Kyoto, JAPAN, OCT 11-14, 2016.
- [95] Javier Silvestre-Blanes and Rubén Pérez-Lloréns. Energy efficiency improvements through surveillance applications in industrial buildings. *Energy and Buildings*, 43(6):1334–1340, 2011.
- [96] Chakchai So-In, Raj Jain, Subharthi Paul, and Jianli Pan. Future wireless networks: key issues and a survey (id/locator split perspective). *International Journal of Communication Networks and Distributed Systems*, 8(1-2):24–52, 2012.
- [97] Zhongwei Sun, Shengwei Wang, and Zhenjun Ma. In-situ implementation and validation of a co₂-based adaptive demand-controlled ventilation strategy in a multi-zone office building. *Building and Environment*, 46(1):124–133, 2011.
- [98] Guoming Tang, Kui Wu, Jingsheng Lei, and Weidong Xiao. The meter tells you are at home! non-intrusive occupancy detection via load curve data. In *Smart Grid Communications (SmartGridComm), 2015 IEEE International Conference on*, pages 897–902. IEEE, 2015.

- [99] T Teixeira, G Dublon, and A Savvides. A survey of human-sensing: Methods for detecting presence, count, location, track. Technical report, ENALAB, 2010.
- [100] Lakshmi V Thanayankizil, Sunil Kumar Ghai, Dipanjan Chakraborty, and Deva P Seetharam. Softgreen: Towards energy management of green office buildings with soft sensors. In *Communication Systems and Networks (COMSNETS), 2012 Fourth International Conference on*, pages 1–6. IEEE, 2012.
- [101] Robert Tomastik, Yiqing Lin, and Andrzej Banaszuk. Video-based estimation of building occupancy during emergency egress. In *2008 American Control Conference*, pages 894–901. IEEE, 2008.
- [102] Mohammad Asif ul Haq, Mohammad Yusri Hassan, Hayati Abdullah, Hasimah Abdul Rahman, Md Pauzi Abdullah, Faridah Hussin, and Dalila Mat Said. A review on lighting control technologies in commercial buildings, their performance and affecting factors. *Renewable and Sustainable Energy Reviews*, 33:268–279, 2014.
- [103] Luxmi Verma, Sangeet Srivastava, and P. C. Negi. A Hybrid Data Mining Model to Predict Coronary Artery Disease Cases Using Non-Invasive Clinical Data. *Journal of Medical Systems*, 40(7), JUL 2016.
- [104] Bill Von Neida, Dorene Manicria, and Allan Tweed. An analysis of the energy and cost savings potential of occupancy sensors for commercial lighting systems. *Journal of the Illuminating Engineering Society*, 30(2):111–125, 2001.
- [105] Shengwei Wang, John Burnett, and Hoishing Chong. Experimental validation of co₂-based occupancy detection for demand-controlled ventilation. *Indoor and Built Environment*, 8(6):377–391, 2000.
- [106] Kevin Weekly, Donghyun Rim, Lin Zhang, Alexandre M Bayen, William W Nazaroff, and Costas J Spanos. Low-cost coarse airborne particulate matter sensing for indoor occupancy detection. In *Automation Science and Engineering (CASE), 2013 IEEE International Conference on*, pages 32–37. IEEE, 2013.
- [107] Dominic Wörner, Thomas von Bomhard, and Felix Wortmann. Occupancy-based heating control for residential buildings using environmental sensors. In *Proceedings of the 5th ACM Workshop on*

- Embedded Systems For Energy-Efficient Buildings*, pages 1–2. ACM, 2013.
- [108] Z. Yang, N. Li, B. Becerik-Gerber, and M. Orosz. A systematic approach to occupancy modeling in ambient sensor-rich buildings. *Simulation-Transactions of the Society for Modeling and Simulation International*, 90(8):960–977, 2014.
- [109] Zheng Yang, Nan Li, Burcin Becerik-Gerber, and Michael Orosz. A multi-sensor based occupancy estimation model for supporting demand driven hvac operations. In *Proceedings of the 2012 Symposium on Simulation for Architecture and Urban Design*, SimAUD '12, pages 2:1–2:8, San Diego, CA, USA, 2012. Society for Computer Simulation International.
- [110] Ehsan Yavari, Hsun Jou, Victor Lubecke, and Olga Boric-Lubecke. Doppler radar sensor for occupancy monitoring. In *Silicon Monolithic Integrated Circuits in RF Systems (SiRF), 2013 IEEE 13th Topical Meeting on*, pages 216–218. IEEE, 2013.
- [111] Rui Zhang, Khee Poh Lam, Yun Shang Chiou, and Bing Dong. Information-theoretic environment features selection for occupancy detection in open office spaces. *Building Simulation*, 5(2):179–188, 8 2012.
- [112] Zhongtang Zhao, Zhenyu Chen, Yiqiang Chen, Shuangquan Wang, and Hongan Wang. A class incremental extreme learning machine for activity recognition. *Cognitive Computation*, 6(3):423–431, 2014.
- [113] Zi-Ning Zhen, Qing-Shan Jia, Chen Song, and Xiaohong Guan. An indoor localization algorithm for lighting control using rfid. In *Energy 2030 Conference, 2008. Energy 2008. IEEE*, pages 1–6. IEEE, 2008.
- [114] Stylianos Zikos, Apostolos Tsolakis, Dimitrios Meskos, Athanasios Tryferidis, and Dimitrios Tzovaras. Conditional Random Fields - based approach for real-time building occupancy estimation with multi-sensory networks. *Automation in Construction*, 68:128–145, AUG 2016.
- [115] Han Zou, Hao Jiang, Xiaoxuan Lu, and Lihua Xie. An online sequential extreme learning machine approach to wifi based indoor positioning.

- In *Internet of Things (WF-IoT), 2014 IEEE World Forum on*, pages 111–116. IEEE, 2014.
- [116] Han Zou, Hengtao Wang, Lihua Xie, and Qing-Shan Jia. An rfid indoor positioning system by using weighted path loss and extreme learning machine. In *Cyber-Physical Systems, Networks, and Applications (CPSNA), 2013 IEEE 1st International Conference on*, pages 66–71. IEEE, 2013.
- [117] Han Zou, Lihua Xie, Qing-Shan Jia, and Hengtao Wang. An integrative weighted path loss and extreme learning machine approach to rfid based indoor positioning. In *Indoor Positioning and Indoor Navigation (IPIN), 2013 International Conference on*, pages 1–5. IEEE, 2013.