Estimating Occupancy In Heterogeneous Sensor Environment

Manar Amayri, Abhay Arora, Stéphane Ploix, Sanghamitra Bandhyopadyay, Quoc-Dung Ngo, Venkata Ramana Badarla

Estimating Occupancy In Heterogeneous Sensor Environment

Manar Amayri[a], Abhay Arora[b], Stephane Ploix[a], Sanghamitra Bandhyopadyay[c], Quoc-Dung Ngo[d], Venkata Ramana Badarla[b]

[a]*G-SCOP lab / Grenoble Institute of Technology, Grenoble, France*
[b]*Indian Institute of Technology, Jodhpur, India*
[c]*Indian Statistical Institute, Kolkota, India*
[d]*Posts and Telecommunications Institute of Technology, Hanoi, Vietnam*

Abstract

A general approach is proposed to determine the common sensors that shall be used to estimate and classify the approximate number of people (within a range) in a room. The range is dynamic and depends on the maximum occupancy met in a training data set for instance. Means to estimate occupancy include motion detection, power consumption, CO_2 concentration sensors, microphone or door/window positions. The proposed approach is inspired by machine learning. It starts by determining the most useful measurements in calculating information gains. Then, estimation algorithms are proposed: they rely on decision tree learning algorithms because these yield decision rules readable by humans, which correspond to nested if-then-else rules, where thresholds can be adjusted depending on the living areas considered. In addition, the decision tree depth is limited in order to simplify the analysis of the tree rules. Finally, an economic analysis is carried out to evaluate the cost and the most relevant sensor sets, with cost and accuracy comparison for the estimation of occupancy. C45 and random forest algorithms have been applied to an office setting, with average estimation error of 0.19-0.18. Over-fitting issues and best sensor sets are discussed.

Keywords: human behavior, building performance, activities recognition, office buildings, machine leaning, data mining.

1. Introduction

Recently, research about building has turned to focusing on occupant behavior. Most of these works deal with the design stage: the aim is to represent the diversity

of occupant behaviors in order to guarantee minimal measured performance. Most of the approaches use statistics about human behavior (Roulet et al., 1991; Page et al., 2007; Robinson and Haldi, 2009). (Kashif et al., 2013) which emphasized that inhabitants' detailed reactive and deliberative behaviors must also be taken into account and a co-simulation methodology proposed to find out the impact of certain actions on energy consumption.
Nevertheless, human behavior is not only interesting during the design step, but also during operation. It is indeed useful for diagnostic analyzes to discriminate human misbehavior from building system performance, and also for energy management where strategies depend on human activities and, in particular, on the number of occupants in a zone. Unfortunately, the number of occupants is not easy to measure. Counting gates are expensive to deploy at room level inside a whole building. This paper tackles this issue. It proposes occupancy estimators combining different common measurements such as CO_2 concentration, motion detection, power consumption,... because only one measurement proved to be not reliable enough to estimate the number of occupants. For instance, CO_2 concentration may be useful but in some configurations, when a window is opened for instance, estimates become unreliable. Motion detection and power consumptions depend on occupant activities. However, altogether, these measurements can be combined to get a more reliable estimator.
Section 2 presents a state of the art about occupancy estimation. Section 3 discusses the proposed process that leads to an occupancy estimator suitable for a specific context. Section 5 points out the most relevant measurements to consider for an estimator. Section 6 focuses on how to obtain rules from a decision tree structure. Section 7 compares estimates of occupancy with actual ones in the context of an office.

2. State of the art

Occupant behavior is one of the major factors influencing building energy consumption. (Honga et al., 2015) introduced methods in modeling occupant behavior and quantifying its impact on building energy use. The major themes include advancements in data collection techniques, analytical and modeling methods, and simulation applications, which provide insights into behavior energy savings potential and impact.
Numerous studies have developed various control systems and modeling methods to better assist occupants to play active roles in buildings. In (Zhao et al., 2014),

electricity metered data of office appliances are used to build the occupant individual and the group behavior models. An application is installed in the occupants computer to reveal presence/absence information, whilst in this paper, the estimation is totally based on common cheap non-intrusive sensors (i.e. motion detector, CO2 concentration, ...).

Works aiming at finding occupancy using common sensors have been already tackled and various methods have been investigated. Methods vary from basic single feature classifiers that distinguish among two classes (presence and absence) to multi-sensor, multi-feature models. A primary approach, which is prevalent in many commercial buildings, is to use passive infrared (PIR) sensors for occupancy. However, motion detectors fail to detect presence when occupants remain relatively still, which is quite common during activities like working on a computer, or regular desk work. Furthermore, drifts of warm or cold air on objects can be interpreted as motion leading to false positive detections. This makes the use of only PIR sensors less attractive for occupancy counting purposes. Conjunction of PIR sensors with other sensors can be useful as discussed in (Agarwal et al., 2010). It makes use of motion sensors and magnetic reed switches for occupancy detection in order to increase the efficiency of HVAC systems in smart buildings. It is quite simple and non-intrusive. Apart from motion, acoustic sensors (Padmanabh et al., 2009) are also used. However, audio signals from the environment can easily fool such sensors and, with no support from other sensors, it can report many false positives. In the same way, other sensors like video cameras (Erickson et al., 2011; Milenkovic and Amft, 2013b), which exploit the huge advances in the field of computer vision and the ever increasing computational capabilities RFID tags (Philipose et al., 2004) installed on ID cards, and sonar sensors (Milenkovic and Amft, 2013a) plugged on monitors to detect the presence of a person at a computer desk, have been used and proved to be much better at solving the problem of occupancy counts, though they may not be used in most office buildings for reasons like privacy and costs. The use of pressure and PIR sensors to determine presence/absence in single desk offices has been discussed in (Nguyen and Aiello, 2012); further tagging of activities is based on this knowledge.

However, for various applications like activity recognition or context analysis within a larger office space, information regarding the presence or absence of people is not sufficient and an estimation of the number of people occupying the space is essential. (Lam et al., 2009) investigates this problem in open offices, estimating occupancy and human activities using a multitude of ambient information, and compare the performance of hidden Markov models, support vector machines and Artificial Neural Networks. Though none of these methods gen-

erates human-understandable rules, this may be helpful to manually adapt and customize estimators without requiring a time-consuming occupancy labeling.
(Ebadat et al., 2013) focuses on how to accurately estimate the number of occupants in a room by processing CO_2 concentration, temperature and HVAC actuation levels in order to identify a dynamic model. In (Dong et al., 2010), hidden Markov models have been used for estimating occupancy using a wireless ambient sensing system as well as wired carbon dioxide sensors and a wired camera network in order to establish actual occupancy levels.
The problem of real time estimation of occupancy in a commercial building has also been investigated in (Liao and Barooah, 2010), where merging sensor data with model predictions was essential. Additionally, real-time estimation of building occupancy is extremely valuable during emergency egress. In (Tomastik et al., 2010), an extended Kalman filter, which combines sensor readings and a dynamic stochastic model of people movements, was used.
An alternate approach aims at understanding the relationships between carbon dioxide concentration, IAQ (Indoor Air Quality) and the number of occupants. Such a physical CO_2 model built on sensor networks has been extensively used in (Aglan, 2003) with smart office projects to improve occupant comfort and minimize building energy use. In this paper, a model has been proposed to find out the usefulness of using CO_2 in occupancy estimation. (Nishi, 2012) uses CO_2 concentration sensors to estimate the number of occupants in a small room with controlled ventilation maintaining a constant CO_2 concentration for environmental comfort. (Tachikawa and akihiro Oda, 2008) verifies the effectiveness of CO_2 sensor installed in KNIVES terminals to estimate the number of people. (Risuleo et al., 2015) presented a modeling of the dynamic relationship between occupancy of a room and CO_2concentration. In fact, since occupancy affects the indoor environment through heat gains and CO_2, its estimation is crucial to determine the evolution of indoor environmental conditions. A CO_2 sensor is used to measure CO_2 concentration for assessing indoor vitiation. (Tachikawa et al., 2008) confirmed the effectiveness of this sensor in estimating the number of people. This estimation has been done by estimating the amount of air ventilation using CO_2 concentration measurements and the actual number of people, then estimating the number of people by measured values of CO_2 concentration and estimated amount of air ventilation. Here, the number of people is one of the parameter needed for effective control.
In (D'Oca and Honga, 2014), a data mining learning process has been proposed to extrapolate office occupancy patterns and working user profiles from big data streams. A data mining schedule learning method has been applied with the open

source data mining program RapidMiner to provide insights into patterns of occupancy in 16 offices.
(Hailemariam et al., 2011) exploits numerous features derived from multiple sensor types. The classifiers use only two classes: one which represents the absence of occupants and one which represents the presence of at least one occupant using decision tree. While in our paper the method aims at counting the number of occupants in a space, so separate classes are used, also overfitting problem are solved with analysing to the readable C4.5 method.
In general, an occupancy counting algorithm that fully exploits all the available information coming from low cost and non-intrusive sensors is an important yet little explored tool to solve problems in office buildings.

3. Data processing

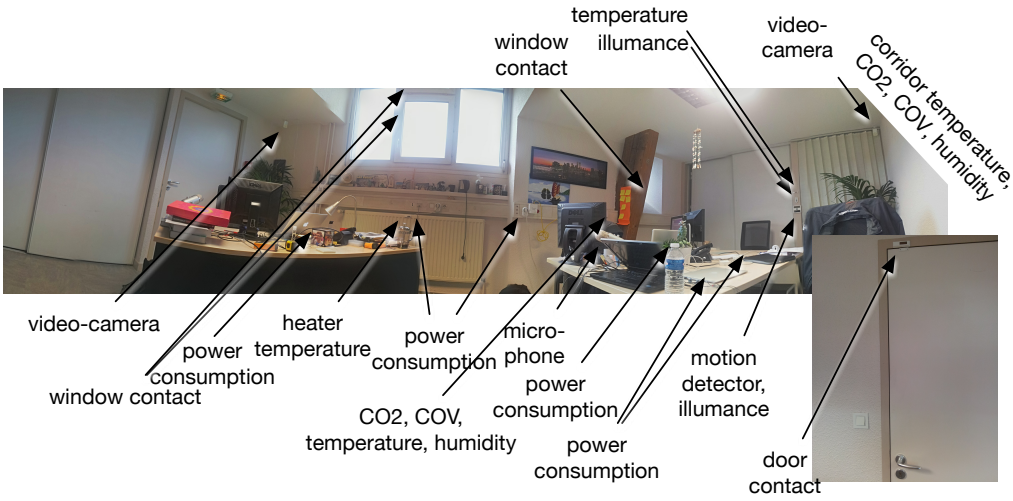

Figure 1: Sensor test bed at Grenoble INP

The test bed (figure 1) is an office in Grenoble Institute of Technology, which accommodates a professor and 3 PhD students. The office frequently houses visitors with a lot of meetings and presentations all throughout the week. The set-up for the sensor network includes:

- 2 video cameras for recording real occupancy numbers and activities.
- An ambiance sensing network, which measures luminance, temperature, relative humidity (RH), motions, CO_2 concentration, power consumption,

door and window positions, acoustic pressure from microphone (described as 'microphone' in the rest of the article). Data are sent thanks to ENOCEAN protocol on significant value change event.

- A centralized database with a web application for continuously retrieving data from different sources.

To perform the task of finding the number of occupants, a link needs to be established between the office environment and the number of people in it. The office environment can be represented as a set of state variables, $V_t = [v_1, v_2, \ldots, v_n]_t$. This set of state variables Vmust be indicative of occupancy at any instance of time t . A state variable can be termed as a feature, and therefore the set of features as a feature vector. Similarly, the n-dimensional space that contains all possible values of such a feature vector is the feature space. The underlying approach for the experiments is to formulate the classification problem as a map from a feature vector into some feature space that comprises several classes of occupancy. Therefore, the success of such an approach depends heavily on how good (those which provide maximum separability between classes) the selected features are. In this case, features are attributes from multiple sensors accumulated over a time interval. The choice of interval duration is highly context-dependent, and has to be done according to the granularity required. However, some features do not allow this duration to be arbitrarily small. As an example, it has been observed that CO_2 levels do not rise immediately, and one of the factors affecting this delay is the ventilation of the space being observed. The results presented in this paper are based on an interval of T_s = 30 minutes (which has been referred to here as 1 quantum).

Before computing any feature for the training data, some basic data pre-processing had to be carried out: application of an *outliers removal algorithm* and *interpolation* for non-existent data. The interpolation part is necessary to fill in the missing values from sensor data. This is frequent in devices which are event-triggered i.e., no data points are reported if there is no change in the feature being reported. Thus, the previous data point had to be duplicated to fill the blanks.

3.1. Data preprocessing

Despite the use of reliable sensors, some single data point spikes have been observed in the recordings, which are attributed to random faults in the sensors. The faults can easily be visually identified in a continuous time-series, but in order to identify and remove them statistically, it is necessary to understand what makes a data point an outlier. Subsequent removal is almost trivial. Contextual outliers are defined as data points which, when compared with its preceding and ensuing

data points, seem highly improbable. These points were identified as those that simultaneously answer the equations $\forall k$:

$$|\Delta x_k| > m(\Delta x) + \lambda\sigma(\Delta x)$$
$$|\Delta x_{k+1}| > m(\Delta x) + \lambda\sigma(\Delta x)$$
$$\Delta x_{k+1}.\Delta x_k < 0$$

with:

x_k	the value of the feature at time quantum k
Δx	the sequence of differences between two consecutive data: $(x_{k+1} - x_k, \forall k)$ with $\Delta x_k = x_k - x_{k-1}$
$m(.)$	the average value
$\sigma(.)$	the standard deviation
λ	a configurable parameter, typically $\lambda = 5$

All the data points that satisfy the above equation were removed.

3.2. Data interpolation

Let $\mathcal{T}_k = \{t_i : t_i \in [kT_s, (k+1)T_s]\}$ be the time samples related to time quantum k. Two kinds of sensors exist:

counters such as PIR motion detectors which send impulses that have to be computed for each time quantum

states which send a value any time the measured state such as the temperature, the CO_2 concentration, or a door or window contact changes. The way of averaging state variables is given in figure 2.

The following measurements directly corresponding to a basic feature were taken into consideration:

- averaged CO_2 concentration in this office and in the corridor (outdoor CO_2 concentration is assumed to be 395ppm)
- averaged temperature in this office, the corridor and outdoor
- averaged opening for door and windows. This feature is extracted for the door and window contact sensors. 0 means the door/window was always closed and 1 always opened. A value in $(0, 1)$ denotes the time ratio during which it was opened during the related time quantum.
- motion counter. The PIR sensor in use is a binary sensor that reports a value of 1 whenever it senses some motion.
- time slot generated from calendar: NIGHT, PRELUNCH, LUNCH, POSTLUNCH. It corresponds to the respective time intervals [20-8),[8-12),[12-14),[14-20).

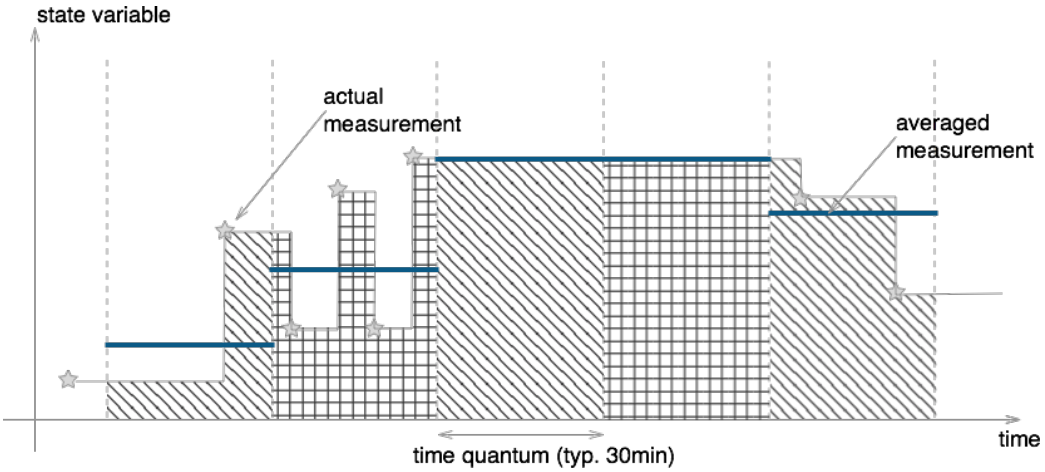

Figure 2: State averaging principle

- type of day generated from calendar: one among working, absence

Additional compound features are also considered:

- averaged opening for door and windows. This feature is extracted for the door and window contact sensors. 0 means the door/window was always closed and 1 always opened. A value in $(0, 1)$ denotes the time ratio during which it was opened during the related time quantum.
- difference between outdoor and indoor averaged temperatures
- CO_2 trend defined as $\Delta CO_{2K} = CO_{2k} - CO_{2k-1}$

4. Calculation of information gains

From the large set of features discussed above, some of them may not be worth considering in order to achieve our target of occupancy classification. These features are the ones which, when added to the classification algorithm, make no difference to the overall output. In other words they are not useful enough for our purpose. As an example, absolute temperature readings would be useless, as it is not representative of occupancy at all. One quantitative measurement of the usefulness of a feature is *information gain*. Before detailing what an *information gain* is, it is imperative to discuss the concept of *entropy*. Entropy is an attribute of a random variable that categorizes its disorder. The higher the entropy, the higher the disorder associated with the variable i.e. the less it can be predicted.

A supervised learning approach has been used. Hence, for preparing training data, occupancy count was manually annotated using the video feed from two

cameras strategically positioned in the office. The weighted average of the number of people visiting the office was recorded for each 30 minutes of the day. To complement this data, a keyboard connected to a *Raspberry PI* module was set up for use by the visitors for updating person-count in the office as they entered/left the office. Data recorded in this way is much easier to feed to a script, hence allowing full automation of the training part; provided visitors and office workers care to follow procedures, it would be possible to consider the data obtained as reliable.

The *decision tree* classification technique has been selected as our prediction model because it provides human-readable results which can be analyzed and easily adapted by building managers. The decision tree algorithm selects a class by descending a tree of decision nodes where each internal node represents a comparison of a single feature value with a learned threshold. The leaf nodes represent the selected classes for the given features. The target of the decision tree algorithm is to select features that are more useful for classification. As information gain approaches zero, the difference between the initial disorder (entropy) of the target variable and the entropy of the variable after adding the observation from the test feature x becomes negligible. Hence, this particular feature will not be very helpful during the decision-making process.

A decision tree algorithm provides quite a few advantages. As per (Quinlan, 1986), the features with higher information gains are situated much higher up the tree, therefore making the process of feature selection intrinsic to the classifier. Since the path to the leaf may consist of many internal nodes, each of which may check different feature values, such paths exploit the correlation among the different features. The decision tree approach offers the advantage of generating rules that the path towards the leaf node is quite informative and it clearly points out direct causes for the selection of a particular class. Unlike methods that use decision boundaries (SVMs, regression techniques), decision tree analyzes are independent of the scale of the input data, so no or little conditioning of the data is necessary.

Using this raw training data, the features previously mentioned were extracted. A vector of features and target $\langle f_1, f_2, f_3, \ldots, f_N; y \rangle$ was generated for each time quantum, where f_i stands for the i^{th} feature and y, for the level of occupancy. Here, the level corresponds to an interval in the partition $\mathcal{L} = \langle I_0, \ldots, I_{l-1} \rangle$ where l in the number of occupancy levels.

5. Selection of best features

This section introduces the most relevant features that have been identified.

5.1. Occupancy from power consumption

Power consumption sensors are easy to deploy in most households and offices. In order to investigate the possibility of using power consumption data in occupancy recognition, 4 sensors have been connected to inhabitant laptops in the office studied. By analyzing the power consumption data, a threshold was determined to discriminate between cases of computer standby and cases in which someone was working on a laptop, which increased the power consumption and lead to wrong occupancy estimations (Kleiminger et al., 2013).

$$\pi_{i,k}(\text{threshold}) = \begin{cases} 0 & \text{if } power_{i,k} < \text{threshold} \\ 1 & \text{otherwise} \end{cases}$$

where $power_i$ stands for the i^{th} laptop averaged power consumption during time quantum ki and threshold $= 15W$ typically. The level of occupancy is then estimated with $\hat{l}_k^{\pi}(\text{threshold}) = \sum_i \pi_{i,k}(\text{threshold})$. It has to be noticed that although this estimator is one of the most relevant feature, it is not reliable in the presence of visitors.

5.2. Occupancy from CO_2 physical model

An alternative approach for occupancy estimation can be done by using physical CO_2 model. According to ASHRAE (1985), the model given by (1) represents the relationship between carbon dioxide generation, volumetric flow rate of fresh air entering the office, volumetric air flow rate going out of the office, and occupancy (Aglan, 2003). The proposed approach relies on the data for tuning air flows coming from CO_2 concentration sensors, door contact, window contact, and occupancy labels extracted from video cameras, and from constant parameters associated with the office.

$$V\frac{dC_{in}(t)}{dt} = -\left(Q_{out}(t) + Q_{cor}(t)\right)C_{in}(t) + Q_{out}(t)C_{out} + Q_{cor}(t)C_{cor}(t) + n(t)S \tag{1}$$

parameter	initial value	adjusted value
S	7ppm. m^3/s	19.6ppm. m^3/s
C_{out}	395ppm	420ppm
Q^0_{out}	0.004m^3/s	0.076m^3/s
Q^0_{cor}	0.004m^3/s	0m^3/s
Q_D	0.04m^3/s	0.1m^3/s
Q_W	0.04m^3/s	0.09m^3/s

Table 1: Adjusted parameter values for physical CO_2 model

It yields the following estimator:

$$n_k = \frac{C_{in,k+1} - \alpha_k C_{in,k}}{S\beta_k} - \frac{Q_0^{out} C_{out} + (D_k Q_D + Q_0^{cor}) C_{cor,k}}{S} \tag{2}$$

$$\alpha_k = e^{-\frac{\left(W_k Q_W + Q_0^{out} + D_k Q_D + Q_0^{out} + Q_0^{cor}\right) T_s}{V}} \text{ and } \beta_k = \frac{1 - \alpha_k}{W_k Q_W + D_k Q_D + Q_0^{out} + Q_0^{cor}}$$

where:

- time quantum T_s=1800 seconds
- indoor CO_2 concentration: $C_{in}(t)$
- corridor CO_2 concentration: $C_{cor}(t)$
- average opening of the door during a time quantum k: $D_k \in [0, 1]$
- average opening of the window during a time quantum k: $W_k \in [0, 1]$
- CO_2 production for 1 average person: S
- number of persons: n_k
- air flow exchange with corridor: $Q_{cor,k} = D_k Q_D + Q_0^{cor}$ where Q_0^{cor} stands for leaked air flow with corridor
- air flow exchange with outside: $Q_{out,k} = W_k Q_W + Q_0^{out}$

The first step is to find the best parameter values for invariant parameters S, C_{out}, Q^0_{out}, Q^0_{cor}, Q_W and Q_D using an iterative nonlinear optimization approach, taking into account the positions of the door and the window as shown in table 1. An objective function is determined in order to minimize the difference between actual and measured numbers of occupants within the room. Optimization covers a long time span but it can be imagined that less frequent observations could be sufficient.

The next step is to use these adjusted parameters for calculating the number of occupants over a time quantum lasting 30 minutes. Occupancy estimation is obtained from equation (2). Finally, the last step is to use this estimation of occupants as one feature in the classification model.

5.3. Occupancy from acoustic sensor

Acoustic features are a very important part of occupancy classification when other non-intrusive sensors offer poor class separation. A single omni-directional microphone can be used as an important tool, when it comes to occupancy classification. Omni-directional microphones are those which can pick up sound from virtually any direction. They are considerably cheaper than equivalent multiple unidirectional microphones, and have proven valuable in places where tracking/listening to multiple sources is required (Chen et al., 2012; Nuria et al., 2004) like during meetings or discussions. In this paper, the recording signal from an office generally consisted of background environmental noise with a few human voices, some doors opening, and tapping events. From the recording signal the RMS amplitude feature was defined, which is the root mean square (or average) of the amplitude of a sound. However, it is related to the volume of the sound: $V_{RMS} = \sqrt{\frac{\sum_n^{i=1}({S_i}^2)}{n}}$, where n is the number of samples taken and S_i the ${i}^{th}$ sample. High and low RMS value will give an indication of the level of occupants within the office; this relationship is easy to visualize in figure 6, which represents both the RMS amplitude in dB for 4 days, and the actual occupancy profile with respect to time (quantum time is 30 minutes).

5.4. Deciding the number of occupancy levels

In this section, a method for choosing the number of levels (L) of occupancy for classification purposes will be discussed. This number is not fixed and can be changed in accordance with the required average error (average distance between the actual occupancy numbers and the mid points of estimated levels). To determine the number of levels and related non overlapping ranges of occupancy, training data are partitioned into L clusters with $2 \leq L \leq N$, where N is the maximum possible number of occupants. At $L = 2$, the problem amounts to classifying the presence and absence of people. Table 2 shows the different discretizations considered.

5.5. Analysis of the most relevant features

To calculate the information gain, it is necessary to discretize features which contain values that are continuous in nature. A typical discretization function splits a large continuous range into several sub-ranges. However, such a function relies upon:

- sorting the values within the feature to be discretized.

Number of levels	Discretizations
L=2	$\{[=0], [>0]\}$
L=3	$\{[=0], [>0, \leq 3], [>3]\}$
L=4	$\{[=0], [>0, \leq 2], [>2, \leq 4], [>4]\}$
L=5	$\{[=0], [>0, \leq 1], [>1, \leq 2.2], [>2.2, \leq 3.2], [>3.2]\}$
L=6	$\{[=0], [>0, \leq 1], [>1, \leq 2], [>2 \leq 3], [>3, \leq 4], [>4]\}$

Table 2: Levels of occupancy considered with ranges

- determining a splitting cut-point, in accordance with given criteria (maximum and minimum values for each feature).

The first condition requires the elements in the ranges to be comparable. The considered features (CO_2 concentration, motion fluctuations, number of people) are continuous ranges of real numbers, hence can be sorted. For the second, we calculated information gain by distributing our continuous features, respectively into 5 and 8 (intuitively chosen) ranges, to get a better insight into the relationship between the number of ranges and the usefulness of the feature.

CO_2 concentration

disc. 1	$\{[390, 450]$, $[450, 690]$, $[690, 900]$, $[900, 1300]$, $[>1300]\}$
disc. 2	$\{[390, 420]$, $[420, 500]$, $[500, 600]$, $[600, 700]$, $[700, 800]$,$[800, 900]$, $[900, 1300]$, $[>1300]\}$

motion counter

disc. 1	$\{[0, 2]$, $[2, 4]$, $[4, 6]$,$[6, 9]$,$[>9]\}$
disc. 2	$\{[0, 2)$, $[2, 3)$, $[3, 4)$, $[4, 5)$,$[5, 7)$, $[7, 9)$,$[9, 11)$,$[>11]\}$

occupancy levels from physics

disc. 1	$\{[0, 0.5]$, $[0.5, 1.5]$, $[1.5, 3]$, $[3, 4.5]$, $[>4.5]\}$
disc. 2	$\{[0, 0.5]$, $[0.5, 1]$, $[1, 1.5]$,$[1.5, 2]$, $[2, 3]$, $[3, 4.5]$, $[>4.5]\}$

Using information gain values (see Table 3), the most relevant features to estimate occupancy could be determined. The table presents the information gains for the considered discretizations. Selected classifiers would have to determine an optimal discretization. It is done implicitly by the C4.5 and random forest classification algorithms which are presented in the next section.

Finally, after removing less important features, the following main features were considered:

1. motion counter
2. acoustic pressure(microphone)
3. occupancy from power

Feature	IG_1	IG_2
acoustic pressure	0.68	0.78
motion counter	0.62	0.75
occupancy from physical model	0.55	0.56
occupancy from power	0.5	0.55
CO_2 concentration	0.5	0.53
CO_2 trend	0.452	0.49
door opening	0.41	0.41
window opening	0.341	0.341
indoor temperature	0.08	0.093
indoor/outdoor temperature difference	0.07	0.082
day type	0	0

Table 3: Information gain for discretization 1 and 2

Let's now use these features alongside C45 and random forest classifiers to get estimators combining some of features.

6. Getting rule-based estimators

With decision trees. data are split at each node forming a tree according to decisions, which corresponds to nested if-then-else rules. In the if part of such a rule, the decision is made according to a feature of the recorded data. These rules cover both binary ($f_i \in \{0, 1\}$) and continuous data. For example, figure 3 shows part of the final tree structure with root nodes and a few leaf nodes.
In this example, the root is the feature test X_2 i.e., acoustic pressure average value from microphone. It is the most meaningful feature in terms of information gain, and the sub trees unravel in accordance to the rules:

if $X_i \leq \ threshold$ **then**
  left child node
else
  right child node
end if

These rules were repeated in order to build all sub-trees and determine the classification levels, depending on the most informative feature in each sub tree i.e.,:

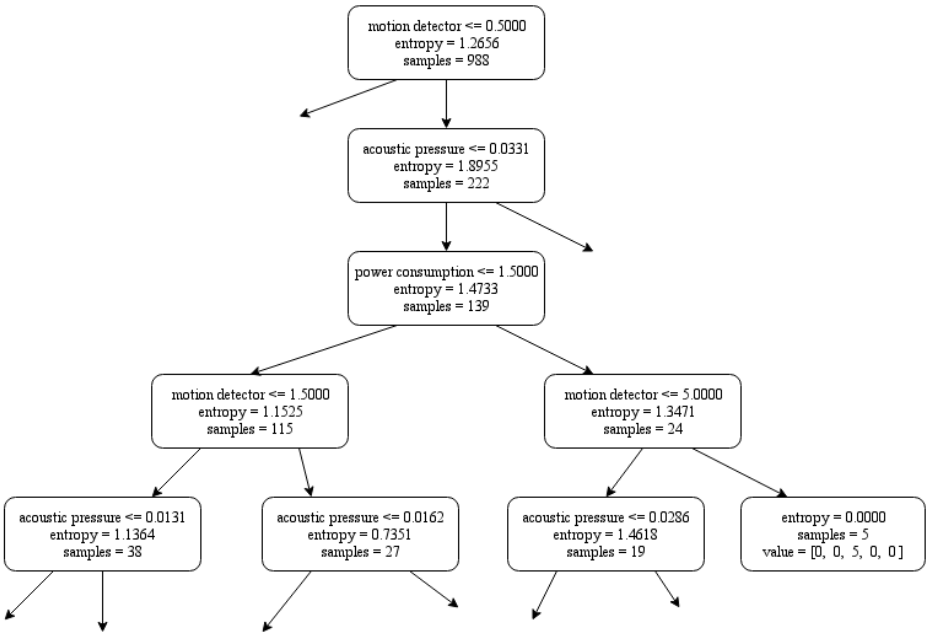

Figure 3: Part of final Decision Tree structure

- motion detector $\wedge$ acoustic pressure $\wedge$ occupancy from power consumption $\wedge$ motion detector $\rightarrow$ classification level$_3$.

For binary data the threshold was 0.5, while for continuous data defining the threshold involved sorting out the data in ascending order then calculating the mid point between successive points (f_i,f_{i+1}), and choosing a precise value for the threshold to be applied in the decision tree (Mitchell, 2007).

6.1. Random forest

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. In other words, random forest is a learning method for classification and regression that operates in constructing a lot of decision trees during training time, then choosing the appropriate classification among all the trees within. Using random forest allows testing of several classifiers, each tree is then constructed using a different bootstrap sample among the original data. About one third of the cases are left out of the bootstrap sample and not used in the construction of the actual tree. In addition, random forest will provide more validation to classification results and avoids the risk of over-fitting training data

. However, over fitting happens when the learning algorithm continues to develop hypotheses, that reduces training set error at the cost of an increased test set error.

6.2. Limiting tree depth

Decision trees provide human-readable classification (if-then) rules that could easily be read to check their realism but could also easily be adapted to another room environment. Indeed, generating a decision tree requires the labeling of a data set with actual occupancies: it raises privacy issues because of the video; also the labeling is time consuming. Reading the parameters of another classifier is more complicated than reading decision tree rules e.g. the hyper parameters in support of a vector machine. Readability of classifier parameters open the gate to its generalization onto different environments.

The maximum tree depth is considered to be a limiting factor to stop further splitting of nodes when the specified tree depth has been reached during the building of the initial decision tree.

The depth of a tree varies according to the size and nature of the sample set.

For example if the depth of the tree is set to '1', a tree with a single node is generated. Otherwise, the most complicated case builds a complete tree, where every path tests every feature. Limiting the depth avoids data over-fitting phenomena by rejecting non significant features. Assume n_s samples and n_f features, at each level (i), the remaining$(n_f - i)$ features for each sample at the level(i) should be examined to calculate the information gain. However, a learned tree is seldom complete (number of leaves is smaller or equal to n_s). In practice, complexity is proportional to both the number of features (n_f) and the number of training samples (n_s).

Furthermore, a trade-off appears between the impurities and the depth of the leaves of a tree. The nodes are expanded until every feature has already been included along this path through the tree, or all the training samples associated with this leaf node reached the same target value (i.e. their entropy is zero), or there are no remaining features for further partitioning. In general, a deep tree with many leaves is usually highly accurate when using the training data but less with the validation data. In addition, finding a shorter decision tree is preferred: it is indeed easier to understand and more reliable than longer trees, and also easier to implement and use.

Using labels is essential with supervised learning. These labels were obtained from video camera footage as discussed before. However, this occupancy detection system gave rise to concerns regarding the occupants' privacy. Another reference estimator could have been used instead: for instance the 'motion counter' or

	Average error decision tree	average error random forest	support
class 1	0.02	0.02	422
class 2	0.39	0.0.35	87
class 3	1	0.91	41
class 4	1.5	1.4	17
class 5	2	2	7
avg/total	0.24	0.23	547

Table 4: Decision tree classification results considering all features

the 'occupancy from power' that proved reliable in many situations. Then, should a problematic situation occur, it could be more deeply investigated.

7. Resulting occupancy estimators

The C4.5 decision tree and random forest algorithms (Quinlan, 2014) have been used to perform recognition tests using aggregated features and labels extracted from video cameras. Five occupancy levels were defined to generate decision trees because of the maximum number of occupants in the office.
Training data covered 22 days from 02-November-2015 to 23-November-2015, whilst the validation data was collected over 12 days from 04-May-2015 to 15-May-2015. Figure 4 shows the results obtained from the learned decision tree and the random forest considering all the features as input to the detection model, where we plotted both actual and estimated occupancy profiles as a graph of the number of occupants with relation to time (quantum time was 30 minutes). The accuracy achieved from decision tree and random forest was 81% (number of correctly estimated points divided by the total number of points), and the average error from decision tree was 0.24 persons, while it was 0.23 from random forest (average distance between actual points and estimated points). Table 4 represents the average error values for each class of estimation. Average error is more interesting than accuracy in the validation of occupancy estimation. Indeed, average error allows us to distinguish each change in the estimated values while accuracy only considers the correctly estimated points.
Figure 5 represents the results obtained from the decision tree and random forest considering only the main features. It lead to an improvement in the estimation of occupancy with an accuracy of 88% and an average error of 0.19 occupant on average, while random forest accuracy was 86% and average error was 0.18.

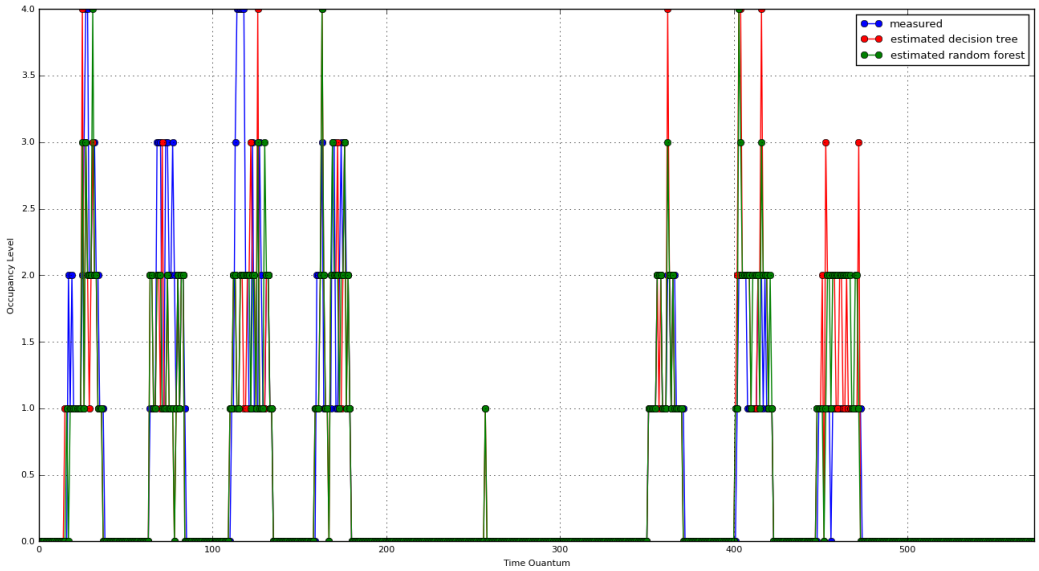

Figure 4: Occupancy estimation from DT using all features

	Average error decision tree	Average error random forest	support
class 1	0.018	0.015	422
class 2	0.09	0.5	87
class 3	0.7	0.4	41
class 4	1	1	17
class 5	1.85	1.9	7
avg/total	0.19	0.18	547

Table 5: Decision Tree classification results after selecting main features

Additionally, the results indicate that motion detector, microphone, and power consumption have the largest correlation with the number of occupants.
A significant improvement in occupancy estimation for all levels could be observed by comparing the results obtained from the decision tree and random forest classifiers using all the features (table 4) with the one using only the main features (table 5). Acoustic pressure (figure 6) was identified as one of the most important feature for occupancy classification according to the final decision tree classification, which ranked the features in an ascending order according to the information gain for each feature, (figure 7). Acoustic pressure improved the estimation in occupancy at high levels.
A deeper analysis has been done regarding the accuracy and the average error

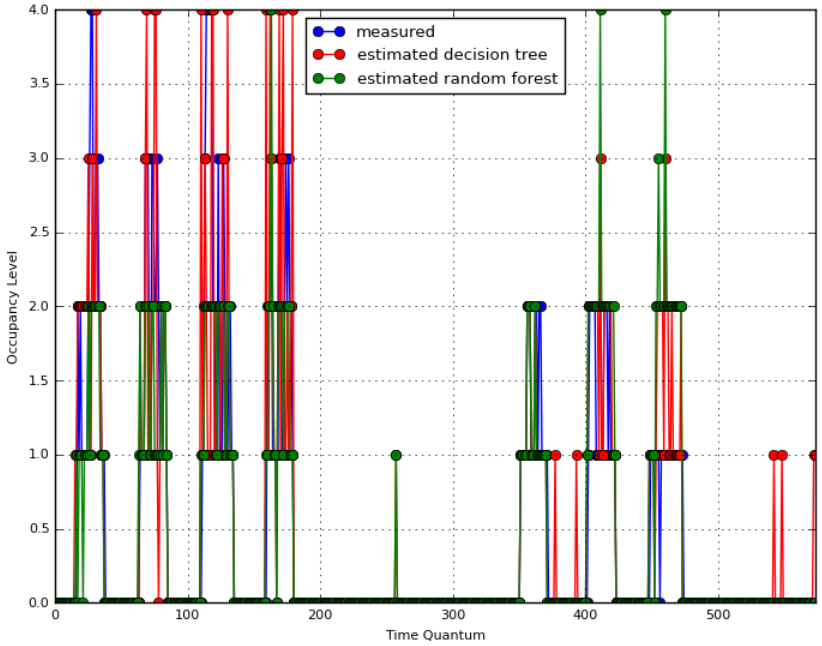

Figure 5: Occupancy estimation from DT using main features

for each day during the testing period ; see table 6. It's obvious the accuracy has a strong relation with the type of the day, in other words, it's very related to the actual level of occupancy in the studied area. Video camera has been used to investigate what happened during the days with poor results. The highest accuracy is achieved during the non-working days (08/05/15, 09/05/15, 14/05/15, 15/05/15) with almost 100% while the lowest accuracy (70% to 75%) is met during the days of high level of occupancy with more than four persons (04/05/15, 05/05/15, 06/05/15, 07/05/15, 13/05/15). For days where occupancy does not exceed 2 persons (09/05/15, 10/05/15, 11/05/15), an accuracy of 96% is obtained. Figure 8 shows average errors associated with each level when applying decision tree and random forest procedures. Accordingly, five levels of occupancy was the best option for the occupancy classification.

Figure 9 represents the results of average errors on occupancy estimation when reducing the decision tree and random forest depths from maximum depth (i.e., 12 to 1).

Figure 9 shows the best depth for decision trees is equal to 5, but it is important to notice that a depth equal to 3 is a good compromise for simplicity. A depth 4 is really interesting according to the random forest results. For the sake of readability

Day	Accuracy DT	Accuracy RF	Average DT	Average RF
04/05/15	76%	76%	0.32	0.3
05/05/15	78%	76%	0.4	0.4
06/05/15	70%	70%	0.48	0.46
07/05/15	76%	74%	0.3	0.28
08/05/15	100%	100%	0	0
09/05/15	97%	97%	0.01	0.01
10/05/15	93%	94%	0.08	0.07
11/05/15	92%	93%	0.19	0.16
12/05/15	81%	78%	0.27	0.23
13/05/15	76%	78%	0.3	0.4
14/05/15	100%	100%	0	0
15/05/15	100%	100%	0	0

Table 6: Accuracy of each day for testing data

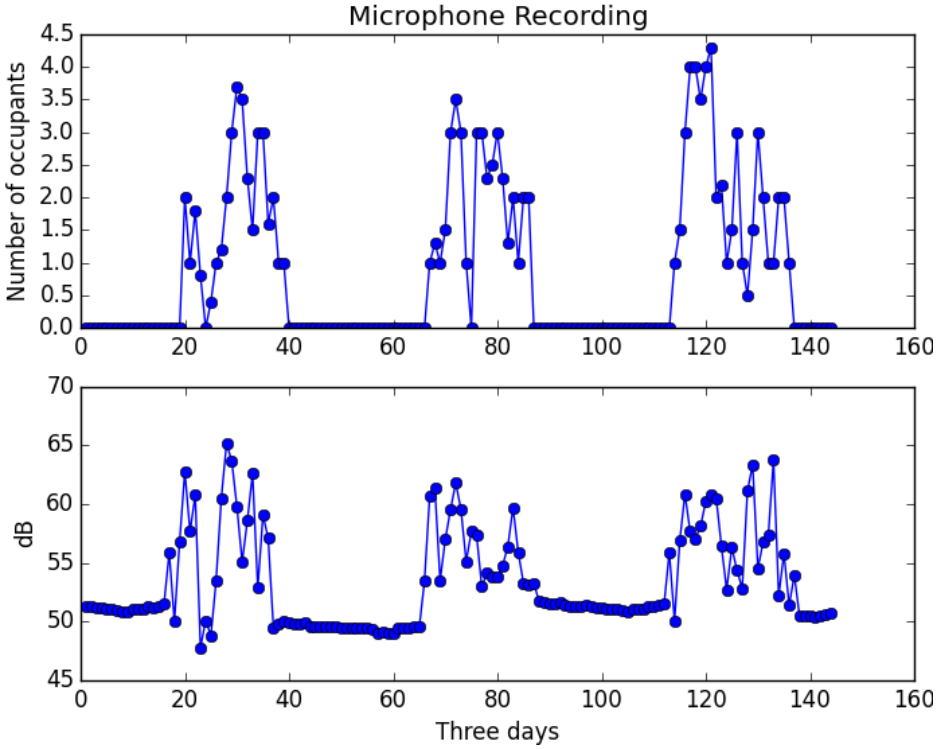

Figure 6: Microphone for three days

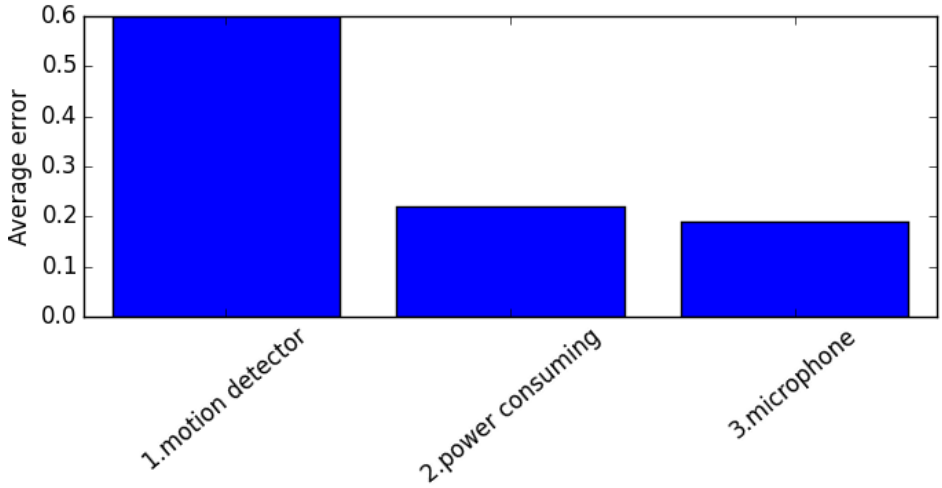

Figure 7: Ranking of features

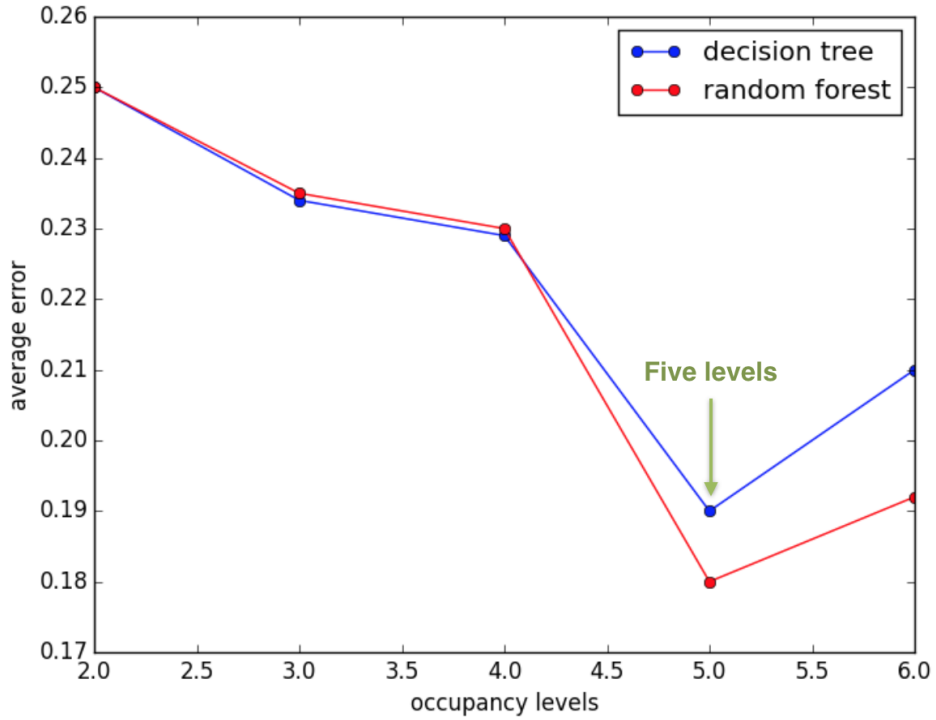

Figure 8: Optimal number of levels for estimation

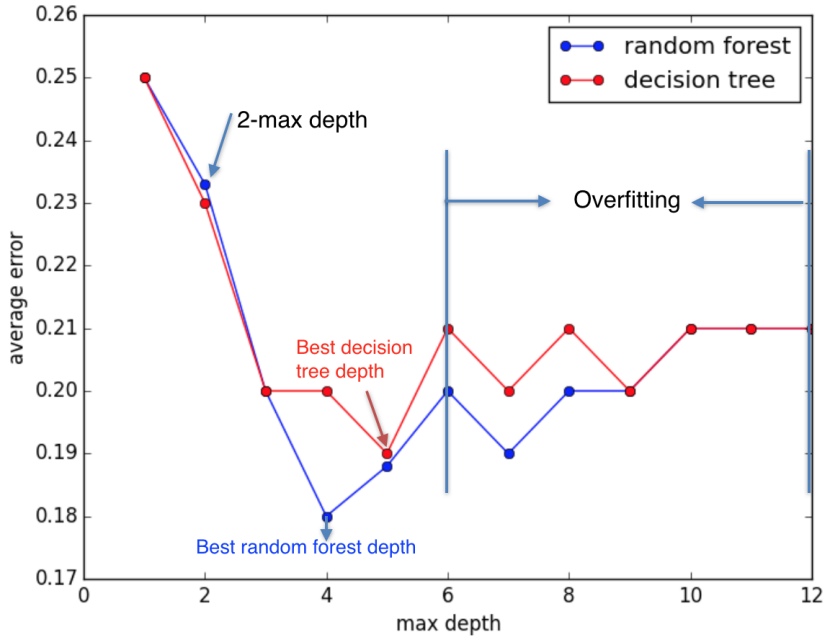

Figure 9: Maximum depth

and because it yields good results still, a depth limited to 2 has been chosen for the next analyses of occupancy estimation.
The tree is easily readable as shown in figures 10 and 11. Note that (if-then) rules from the tree structure could now be extracted easily and be applied to another context and the problem of overfitting is solved.

if motion detector is low and power consumption is low **then**
$\approx$ 0 person
else if motion detector is low and and power consumption is high **then**
$\approx$ 1 person
else if motion detector is high and microphone is low **then**
$\approx$ 2 persons
else if motion detector is high and microphone is high **then**
$\approx$ 3 persons
end if

To validate these rules, figure 12 shows the estimation of occupancy levels for 20 half working hours. The average errors achieved was 0.39 for 20 half hours during working hours.
For more validation and generalizing the occupancy estimation process, more data

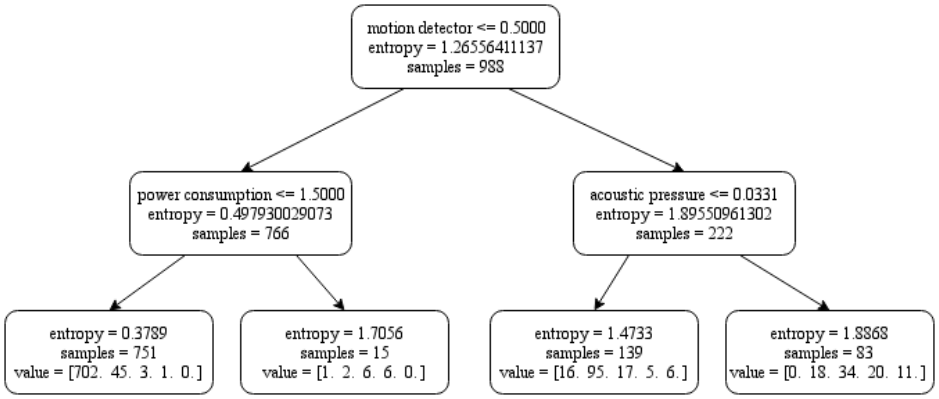

Figure 10: Decision tree structure for max depth=2

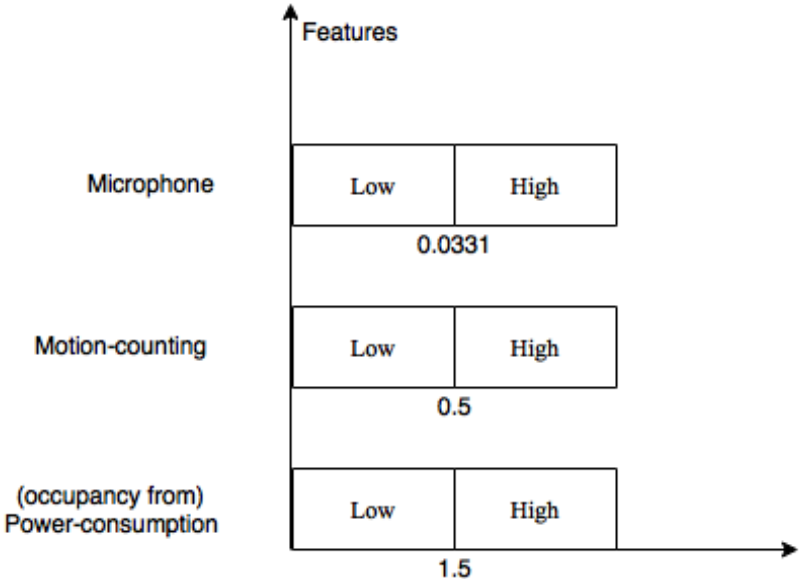

Figure 11: Decision tree rules

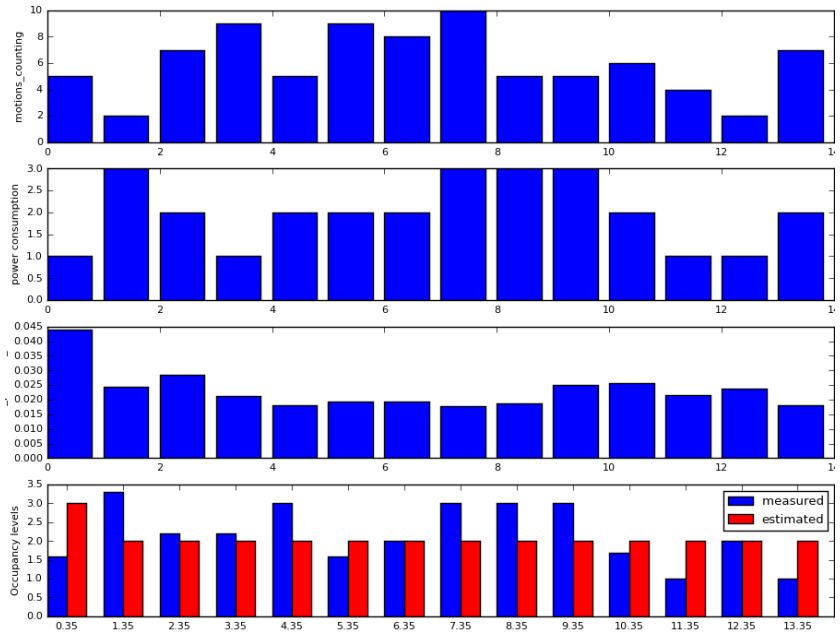

Figure 12: Results for 20 time quanta for occupancy estimation

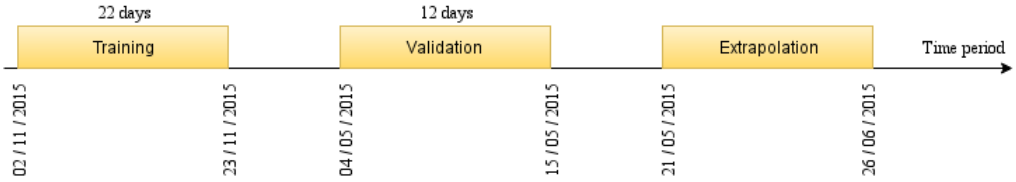

Figure 13: Data sets used for the occupancy estimation

were collected and were divided into three periods (see figure 13):

1. training period for 22 days
2. validation period for 12 days
3. extrapolation period for 37 days

During the estimation process, almost 275000 data points were collected as indicated in table 7.
Figure 14 shows two different strategies for the estimation of occupancy taking into account the limitation to depth=2:

Type of sensor	Datapoints
Power consumption (4 laptops)	92100
Motion detector	3874
microphone	1937
Labels	1728

Table 7: Raw sensor data collected over 11 days

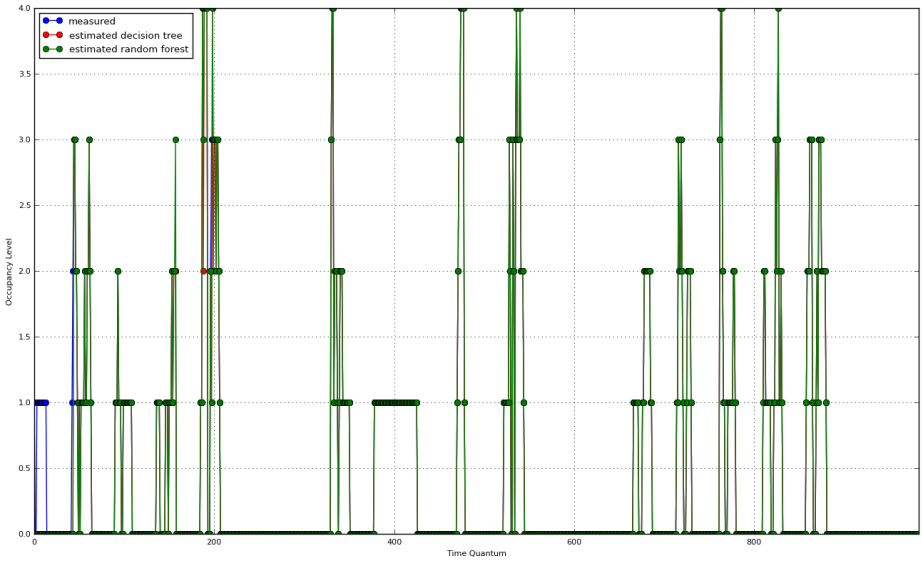

Figure 14: Estimation occupancy for training data

- First strategy: applying the decision tree estimation process for the training period, where achieved average error was 0.09 for the maximum depth and 0.2 for depth=2, according this results 0.09 is the lowest average error that can be achieved in any another period for estimation process, then the validation period where achieved average error was 0.19.

- Second strategy: applying the rules, which were extracted and obtained from the decision tree structure, for the extrapolation period.

Finally, the average number of occupants can be estimated for long periods of time, up to 2 months. Figure 15 shows the average number of occupants for each day of the week. The above results indicate that using decision tree rules gives quite a good estimate of the occupancy.

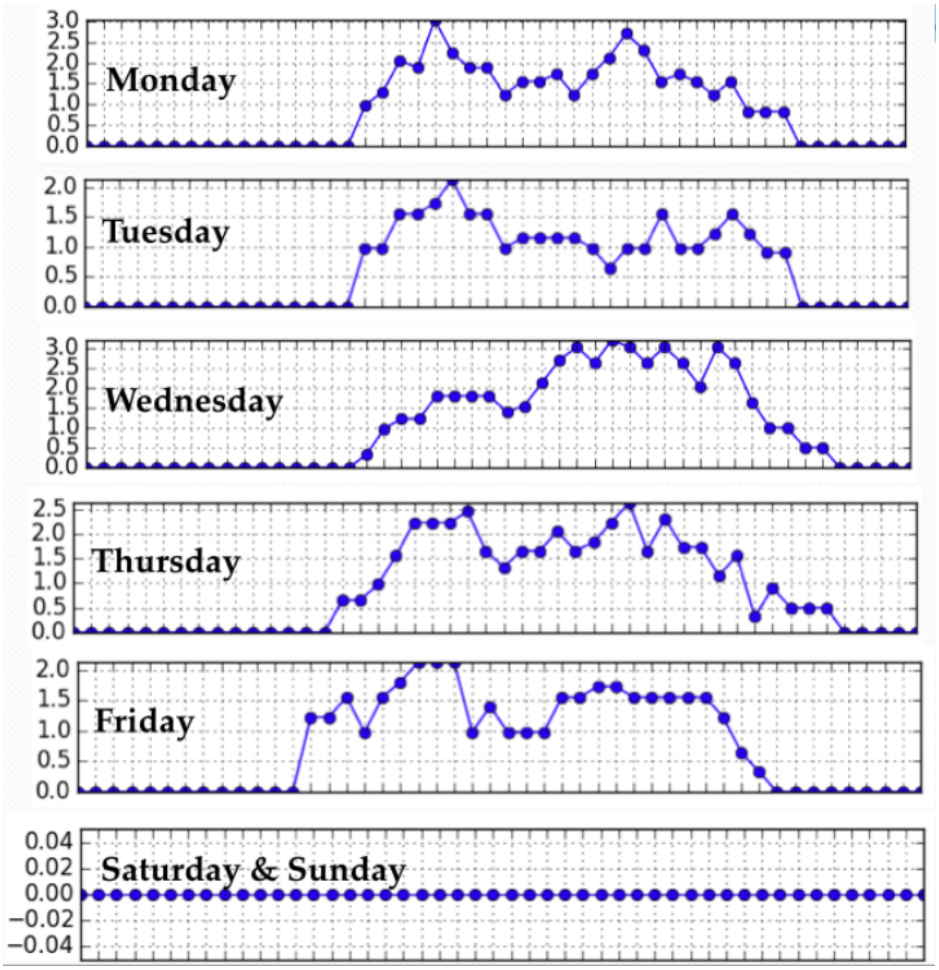

Figure 15: Periods for occupancy estimation

type of sensor	Price (€)
power consumption	4*88.75€
motion detector	119€
microphone	150 €

Table 8: Price of the sensors

type of sensor	estimation average error
motion detector	0.26
power consumption	0.265
microphone	0.48

Table 9: Price of the sensors

7.1. Economical analysis

To complete the study, let us show how to select classifiers from an economical point of view i.e. the average price of the requested sensors. Costs are presented in table 8 according to (Eltako, 2014).

Since decision tree algorithms have different estimation results for different input features, it is possible to decide the most relevant features to retain according to two factors: sensor costs and average error in occupancy.

With three main features, the number of feature combinations is seven combinations, according to $C = \sum_{i=1}^{i=n} \frac{n!}{r_i!(n-r_i)!}$, where n is the maximum number of features (i.e., 3), and r_i the number of features in the combination i. However, these combinations give the opportunity to determine the best compromises.

The 7 combinations allow a comparison between all the different possible arrays of sensors including each sensor alone. It is obvious from figure 16 that the best average error is achieved by using the 3 main sensors i.e. point (4) with an average error of 0.19. Point (0) refers to the use of motion detector sensor alone with an average error of 0.26. Table 9 shows different scenarios of occupancy estimation using each single sensor, it is obvious that using all the three main features makes the estimation process more accurate.

The points in figure 16 shows the 7 different combinations of features. The red ones are those which represent Pareto optimal i.e. the best compromise in terms of cost and average error; they are detailed in table 10.

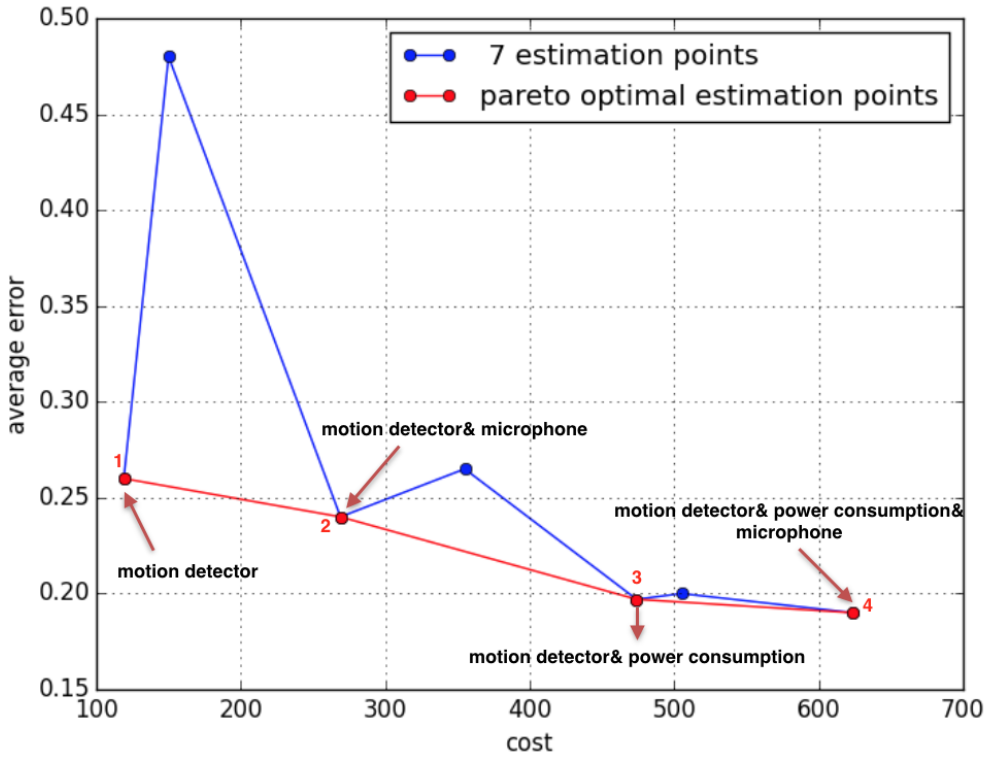

Figure 16: Occupancy estimation with different input features

8. Conclusion

A supervised learning approach has been proposed in this paper to estimate the number of occupants in an office setup. It results in a virtual sensor that relies on other sensors but with a superior performance. The proposed process makes it possible to determine the valuable sensors using the concept of information gain. In the proposed work, motion fluctuation counters using PIR sensors, power consumption sensors, a CO_2 physical model, a microphone as well as door opening contacts are found to be the most interesting sources of information. Decision trees have been obtained using C4.5 classification algorithms. Occupancy estimation using these trees gave a superior performance with an average estimation error of 0.19 occupants over a twelve days' test period. Supervised learning has been carried out using two video cameras but this approach was limited because of privacy issues. Another option has been envisaged: using discrete feedback from occupants themselves through devices like a keyboard or any other means, using

number of case	price (€)	average error	used features
1	119	0.26	motion detector
2	335	0.24	motion detector and microphone
3	407	0.196	motion detector and power consumption
4	662	0.19	motion detector, power consumption and microphone

Table 10: The best cases for occupants estimation

occupancy from other estimators such as power consumption. In addition, because decision trees are human-readable, they can be adjusted using expert knowledge and estimation rules (if-then) extracted from the decision tree structure. For instance, thresholds can be adjusted and nodes for which information is not available can be removed depending on the considered living areas. These two extensions can be combined to avoid the use of video cameras.
Estimating the number of occupants is very interesting in many areas: simulating occupant behavior at design stage, predicting the number of occupants at energy management stage, dissociate physical surroundings from building usage at diagnosis stage etc.. The proposed approach can be extended to the estimation of occupant's activities, which would be useful for developing interactive systems where suitable advice could be provided to occupants at appropriate times.

Acknowledgment

The authors acknowledge the support of the French Agence Nationale de la Recherche (ANR) under reference ANR-13-VBDU-0006 (OMEGA project).

REFERENCES

Agarwal, Y., Balaji, B., Gupta, R., Lyles, J., Wei, M., Weng, T., 2010. Occupancy-driven energy management for smart building automation. In: Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building. ACM, pp. 1–6.

Aglan, H., 2003. Predictive model for co2 generation and decay in building envelopes. JOURNAL OF APPLIED PHYSICS 93 (2).

ASHRAE, Atlanta, G., 1985. Fundamentals American Society of Heating, Refrigerating and Air-Conditioning Engineers. Fundamentals American Society of Heating, Refrig- erating and Air-Conditioning Engineers.

Chen, L., Hoey, J., Nugent, C., 2012. Sensor-based activity recognition. 790 IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS, VOL. 42, NO. 6, NOVEMBER 2012.

D'Oca, S., Honga, T., 2014. Occupancy schedules learning process through a data mining framework. elsevier, Energy and buildings.

Dong, B., Andrews, B., Lam, K. P., ynck, M. H. ., Zhang, R., Chiou, Y.-S., Benitez, D., 2010. An information technology enabled sustainability test-bed (itest) for occupancy detection through an environmental sensing network an information technology enabled sustainability test-bed (itest) for occupancy detection through an environmental sensing network. elsevier.

Ebadat, A., Bottegal, G., Varagnolo, D., Wahlberg, B., Johansson, K. H., 2013. Estimation of building occupancy levels through environmental signals deconvolution. ACCESS Linnaeus Centre, School of Electrical Engineering, KTH Royal Institute of Technology.

Eltako, 2014. The wireless building.

Erickson, V. L., Carreira-Perpiñán, M. Á., Cerpa, A. E., 2011. Observe: Occupancy-based system for efficient reduction of hvac energy. In: Information Processing in Sensor Networks (IPSN), 2011 10th International Conference on. IEEE, pp. 258–269.

Hailemariam, E., Goldstein, R., Attar, R., Khan, A., 2011. Real-time occupancy detection using decision trees with multiple sensor types. Symposium on Simulation for Architecture and Urban Design.

Honga, T., Taylor-Langea, S. C., D'Ocab, S., Yanc, D., Corgnatib, S. P., 2015. Advances in research and applications of energy-related occupant behavior in buildings. elsevier, Energy and buildings.

Kashif, A., Dugdale, J., Ploix, S., 2013. Simulating occupants' behaviour for energy waste reduction in dwellings: A multi agent methodology. Advances in Complex Systems 16, 37.

Kleiminger, W., Beckel, C., Staake, T., Santini, S., 2013. Occupancy detection from electricity consumption data. ACM, pp. 1–8.

Lam, K. P., Höynck, M., Dong, B., Andrews, B., shang Chiou, Y., Benitez, D., Choi, J., 2009. Occupancy detection through an extensive environmental sensor network in an open-plan office building. In: Proc. of Building Simulation 09, an IBPSA Conference.

Liao, C., Barooah, P., 2010. An integrated approach to occupancy modeling and estimation in commercial buildings an integrated approach to occupancy modeling and estimation in commercial buildings. American Control Conference.

Milenkovic, M., Amft, O., 2013a. An opportunistic activity-sensing approach to save energy in office buildings. In: Proceedings of the fourth international conference on Future energy systems. ACM, pp. 247–258.

Milenkovic, M., Amft, O., 2013b. Recognizing energy-related activities using sensors commonly installed in office buildings. Procedia Computer Science 19, 669–677.

Mitchell, T. M., March 2007. Machine Learning. No. 432. McGraw-Hill Science/Engineering/Math.

Nguyen, T. A., Aiello, M., 2012. Beyond indoor presence monitoring with simple sensors. In: PECCS. pp. 5–14.

Nishi, S. I. H., 2012. estimation of the number of people under controlled ventilation using a co2 concentration sensor. ECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society, 2012, p. 4834–4839.

Nuria, O., Ashutosh, G., Eric, H., 2004. Layered representations for learning and inferring office activity from multiple sensory channels.

Padmanabh, K., Malikarjuna V, A., Sen, S., Katru, S. P., Kumar, A., Vuppala, S. K., Paul, S., et al., 2009. 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. ACM, pp. 37–42.

Page, J., Robinson, D., Scartezzini, J., 2007. Stochastic simulation of occupant presence and behaviour in buildings. Proc. Tenth Int. IBPSA Conf : Building Simulation, 757–764.

Philipose, M., Fishkin, K. P., Perkowitz, M., Patterson, D. J., Fox, D., Kautz, H., Hahnel, D., 2004. Inferring activities from interactions with objects. Pervasive Computing, IEEE 3 (4), 50–57.

Quinlan, J. R., 1986. Induction of decision trees. Machine learning 1 (1), 81–106.

Quinlan, J. R., 2014. C4. 5: Programs for machine learning. Elsevier.

Risuleo, R., Molinari, M., Bottegal, G., Karl, H. H., 2015. A benchmark for data-based office modeling: challenges related to co2 dynamics. IFAC SysId 2015.

Robinson, D., Haldi, F., 2009. Interactions with window openings by office occupants. Energy and Buildings 44, 2378–2395.

Roulet, C., Fritsch, R., Scartezzini, J., Cretton, P., 1991. Stochastic model of inhabitant behavior with regard to ventilation. Technical report.

Tachikawa, T., akihiro Oda, 2008. Cooperative distributed demand control by environmental sensor network - estimating the number of people by co2 concentration. in Industrial Informatics (INDIN) 2008. 6th IEEE International Conference on, 2008, 336–341.

Tachikawa, T., Oda, A., Handa, T., Ichimura, J., Watanabe, Y., Nishi, H., 2008. Cooperative distributed demand control by environmental sensor network - estimating the number of people by co2 concentration -. IEEE international Conference on industrial Informatics.

Tomastik, R., Narayanan, S., Banaszuk, A., Meyn, S., 2010. Model Based Real Time Estimation of Building Occupancy During Emergency Egress. No. pp 215-224. Springer Berlin Heidelberg.

Zhao, J., Lasternas, B., Lam, K. P., Yun, R., Loftness, V., 2014. Occupant behavior and schedule modeling for building energy simulation through office appliance power consumption data mining. elsevier, Energy and buildings.