Towards Deep Learning-based Occupancy Detection Via WiFi Sensing in Unconstrained Environments

Cristian Turetta*, Geri Skenderi*, Luigi Capogrosso*, Florenc Demrozi†,
Philipp H. Kindt‡, Alejandro Masrur‡, Franco Fummi*, Marco Cristani*, Graziano Pravadelli*
*Department of Computer Science, University of Verona, Italy, name.surname@univr.it
†Department of Electrical Engineering and Computer Science, University of Stavanger, Norway, name.surname@uis.no
‡Faculty of Computer Science, TU Chemnitz, Germany, name.surname@informatik.tu-chemnitz.de

***Abstract*— In the context of smart buildings and smart cities, the design of low-cost and privacy-aware solutions for recognizing the presence of humans and their activities is becoming of great interest. Existing solutions exploiting wearables and video-based systems have several drawbacks, such as high cost, low usability, poor portability, and privacy-related issues. Consequently, more ubiquitous and accessible solutions, such as WiFi sensing, became the focus of attention. However, at the current state-of-the-art, WiFi sensing is subject to low accuracy and poor generalization, primarily affected by environmental factors, such as humidity and temperature variations, and furniture position changes. Such issues are partially solved at the cost of complex data preprocessing pipelines. In this paper, we present a highly accurate, resource-efficient deep learning-based occupancy detection solution, which is resilient to variations in humidity and temperature. The approach is tested on an extensive benchmark, where people are free to move and the furniture layout does change. In addition, based on a consolidated algorithm of explainable AI, we quantify the importance of the WiFi signal w.r.t. humidity and temperature for the proposed approach. Notably, humidity and temperature can indeed be predicted based on WiFi signals; this promotes the expressivity of the WiFi signal and at the same time the need for a non-linear model to properly deal with it.**

***Index Terms*—WiFi Sensing, Channel State Information, Deep Learning**

I. Introduction

In the context of smart buildings and smart cities, the design of remote monitoring and control systems is becoming of paramount interest, with applications in several fields like, for example, healthcare, ambient assisted living, and Industry 4.0 scenarios [1]. Leveraging dedicated machine learning (ML) techniques, these systems allow observing the status of the environment (*e.g.*, presence and number of occupants) [2], [3], as well as recognizing the human activities performed in it (*e.g.*, walking, standing, sitting) [4]–[6]. This kind of knowledge has practical uses in several contexts. For example, it allows to automatically turn on/off a lighting and air-conditioning system, if a room is occupied or found to be empty, thus optimizing the energy consumption and the building's security [1]. Considering safety aspects, it is exploited in industrial plants to detect and possibly prevent workplace accidents [7].

Current systems utilize wearables, smartphones, dedicated sensors (*e.g.*, passive infrared sensor sensors), ambient sensors (*e.g.*, temperature, humidity, brightness, pressure, CO_2 level), or video cameras to implement the mentioned recognition scenarios [8], [9]. However, these approaches demand special hardware to be installed/worn and, in some cases, have high computational requirements. In addition, video-based systems suffer from field-of-view and privacy-related issues, since it is impossible to install video cameras in every room (*e.g.*, toilets), or to ensure the absence of blind spots [10].

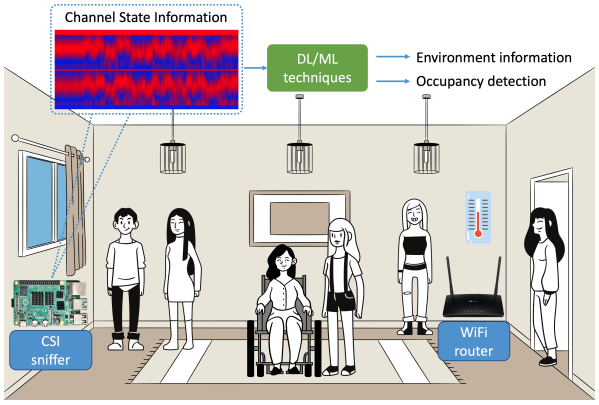

Fig. 1. WiFi sensing systems overview.

WiFi sensing: Beyond the approaches listed above, techniques based on WiFi sensing have become a prominent solution, thanks to WiFi's ubiquity, both indoor and outdoor, and the relatively low deployment costs of such commodity off-the-shell WiFi routers [11], [12]. As shown in Figure 1, WiFi sensing exploits the Channel State Information (CSI) to gain an insight of how WiFi radio signals propagate from a transmitter (*e.g.*, a WiFi router) to a receiver (*e.g.*, a CSI sniffer). By analyzing the variations of the propagation pattern through ML techniques, different works have proposed to recognize the environment status, the number of its occupants, their identity, or the activities they perform [11]–[13].

WiFi sensing limitations: Even though WiFi sensing may seem a panacea for many sensing scenarios, its practical application is not straightforward for real-world cases. In fact, WiFi signal propagation is strongly affected by environmental conditions like *i)* changes in the furniture's layout, *ii)* people's presence and their activities, and *iii)* humidity and temperature [14]. These aspects limit the applications of WiFi-based monitoring systems, and in the current state-of-the-art, this is solved by reducing the possibility of environmental variations.

This research work has been partially supported by the project "Dipartimenti di Eccellenza" 2018-2022 funded by the Italian Ministry of Education, Universities and Research (MIUR), and by the Gruppo Nazionale per il Calcolo Scientifico (GNCS/INDAM).

978-3-9819263-7-8/DATE23/© 2023 EDAA

In particular, existing methodologies [14]–[16] present experimental setups where, *e.g.*:

- the distance between transmitter and receiver is ≤ 2 m;
- the participant performing the activity to be recognized is located exactly between the transmitter and the receiver;
- expensive multi-antenna hardware or software-defined radios are used;
- ad hoc, costly technologies, such as ultrawideband and millimeter wave are used, instead of WiFi;
- computationally-demanding pre-processing pipelines are applied to the data to enhance the performance of the classification/regression algorithms.

Our contributions: This paper is intended to go beyond the limitations mentioned above, for the specific scenario of occupancy detection. It presents a deep learning network capable to detect the presence of people in an indoor environment, independently from their positions, the activities they perform, or the humidity level and temperature. The network is lightweight, permitting the deployment on resource-constrained devices (*e.g.*, Nucleo-L432KC) for real-time utilization. Experiments have been carried out on a large dataset from a completely unconstrained environment, which constitutes a harsh and demanding benchmark for WiFi sensing. Notably, an overall accuracy of 97% is achieved. As a second contribution, a consolidated algorithm of explainable AI for deep networks, Grad-CAM [17], is applied to our approach to explain the reasons for such efficacy. Surprisingly, we found that, given WiFi signals, humidity and temperature variations are of low importance for occupancy detection. Finally, encouraged by a preliminary correlation analysis, we show that the same network is also capable of estimating humidity and temperature from WiFi signals, thus offering an interesting complementary application, which is the third contribution of this paper.

Paper organization: The remaining of the paper has the following structure: Section II provides the background on CSI data and ML performance metrics; Section III gives an overview of the existing literature; Section IV presents our methodology; and Section V reports and discusses the obtained results. Finally, Section VI draws some conclusions and illustrates possible future perspectives.

II. Background

A. Channel State Information

Consider a WiFi sender and a receiver. The sender emits a signal $X(t)$ at time t, while the receiver gets the signal $Y(t)$ characterized by the following equation:

$$Y(t) = H(t) \cdot X(t) + n(t) \tag{1}$$

where $H(t)$ is the CSI vector and $n(t)$ stands for the noise. The dimension, d_H, of $H(t)$ depends on the bandwidth of the communication channel according to the formula $d_H = 3.2 \cdot bandwidth$ [14]. The IEEE 802.11ac protocol supports channels with a bandwidth ranging from 20 MHz up to 160 MHz. For example, if we transmit the signal $X(t_i)$ at time t_i over a 20MHz channel, we obtain a CSI vector $H(t_i)$ of dimension $d_H = 64$. Each element in $H(t)$ is a complex number that describes how WiFi signals propagate from the sender to the receiver on a given channel and its associated subcarriers. The real part describes the amplitude, while the imaginary part is the phase. In this paper, we use only the information contained in the CSI amplitude.

B. Performance measurements metrics

In order to evaluate the forecasting accuracy of the model to estimate humidity and temperature from CSI we utilize the Mean Absolute Error (MAE) and the Mean Absolute Percentage Error (MAPE) functions. In particular, if $\hat{y_i}$ is the predicted value of the i-th sample, and y_i is the corresponding true value, we can define the mentioned scores as follows.

On the one hand, MAE computes a risk metric corresponding to the expected value of the absolute error loss or $L1$-norm loss. MAE estimated over N samples is defined as follows:

$$\mathit{MAE}(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} |y - \hat{y}| \tag{2}$$

On the other hand, MAPE is an evaluation metric for regression problems, which is sensitive to relative errors. It is, for example, not affected by a global scaling of the target variable. MAPE estimated over N samples is defined as follows:

$$\mathit{MAPE}(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} \frac{|y - \hat{y}|}{max(\epsilon, |y_i|)} \tag{3}$$

where ϵ is an arbitrarily small yet strictly positive number to avoid undefined results when y is zero.

III. Related Work

Existing techniques for CSI-based sensing are usually designed under strict constraints, or they exploit costly and ad hoc hardware due to the limitations mentioned in Section I. In particular, existing CSI-based occupancy detection solutions and pipelines analyzing the impact of humidity and temperature variations on the accuracy of WiFi sensing methods are still unsatisfactory.

Occupancy detection has been widely discussed in the literature [14]–[16]. The most promising existing solutions [2], [3], [13], [18] perform well (achieving an accuracy over 95%) in constraint environments, but they are unable to generalize over time. This is related to the training/testing approaches defined by the authors. In particular, such models are trained and tested on temporally close data, thus presenting a high similarity between training and testing samples. However, if such models had been trained and tested over temporally distant data (*e.g.*, training data from one day and test data from another day), their accuracy would drop significantly.

Moreover, to the best of our knowledge, there is no existing article that studies the effect of temperature and humidity variations over CSI-based occupancy detection methods. However, few attempts have shown that CSI can be used to estimate the humidity level [19], or the presence of an indoor fire [20]. Despite this, in [21], authors show that radiation with wavelengths of millimeters leads to a higher accuracy than WiFi in predicting humidity (*i.e.*, 98% vs. 89%).

TABLE I
FORMAT OF THE COLLECTED DATA.

Timestamp	CSI Amplitude			Environment		Occupancy
	$a0$	...	$a63$	Temperature	Humidity	status
15:38:45.550	0.027	...	1	21.97	43	1
15:38:45.600	0.027	...	1	21.82	43	1
15:38:45.650	0.027	...	1	21.82	42	1
...	...	...	...	...	...	...
15:38:45.700	0.027	...	1	23.97	36	0

All the previous approaches present the severe limitations highlighted in Section I, thus being impracticable in unconstrained environments. For example, in [19] authors collected data from an empty environment without the effect of human presence. In [21], authors made use of costly devices that need to be first installed. Finally, in [20], the distance between the receiver and transmitter was set to only 40 cm without the influence of human presence either.

IV. OUR SYSTEM

This section presents the device setup employed to collect CSI and environmental data in an unconstrained environment, and the machine learning model to deal with the occupancy detection task, equipped with a strategy for the model interpretability.

A. Data collection setup

Our data collection setup is shown in Figure 2. A single large office is taken into account: it is $12 \times 6 \times 3$ meters large, with three (2×1.8 meters) windows and one entrance door. Internal walls are made of plasterboard (12 centimeters thick), and external walls are made of reinforced concrete (55 centimeters thick). The occupants were instructed to carry out their office activities (*e.g.*, walking, standing, sitting, going out, and getting in) without any constraints.

Four devices are used for data collection: two Raspberry Pi (RP1 and RP2 in Figure 2), one Access Point (AP), and one Nordic Thingy 52. The AP exposes a WiFi network (over the 2.4 GHz band) observed by the RP1. Every Raspberry Pi device is patched using the Nexmon framework [22] and extracts CSI amplitude data at 20 Hz. The AP and RP1 are placed 2 meters apart at a height of 140 cm, and occupants cannot move between them. The RP2 device communicates over Bluetooth with a Nordic Thingy device, which serves as a ground-truth sensor for the environmental information (*i.e.*, humidity, and temperature). Both Raspberry's are connected to a Dell Inspiron 7559 laptop that synchronizes them and stores the CSI amplitude together with humidity and temperature data. An external observer manually annotated the presence of humans[1] based on recorded video data. A semiautomatic annotation tool simplified the process considerably by avoiding the need to explicitly annotate every single timestamp. An overview of the final dataset format is shown in Table I. At each timestamp, we have the CSI amplitude of the 64 subcarriers (*i.e.*, $a0$ to $a63$), temperature ($^{\circ}C$), humidity ($\%$), and the occupancy status.

[1]Label 0 if the environment is empty, and label 1 if there is at least one person in the environment.

B. The proposed deep network

Given a subcarrier x of a CSI signal, we refer to the time series of CSI amplitudes as $S(x,t)$, where t refers to the t-th time of the day and $x \in \{0,\ldots,63\}$. The time series of humidity h and temperature e are defined as $S(h,t)$ and $S(e,t)$, where $h,e \in R$ represent the values of humidity and temperature at time t.

The ultimate goal of our approach is to exploit the expressiveness of the CSI signal for the occupancy detection task. At the same time, we want a technique that is interpretable, which means that its decisions can be motivated in simple terms, even by a non-expert. Finally, we are interested in presenting a lightweight system, paving the way for real-time applications.

For these reasons, we are proposing here a memory and computation-efficient deep learning-based solution. In detail, we implement a lightweight multilayer perceptron (MLP) comprised of four layers, each except the last activated by the rectified linear activation unit (ReLU) function. Specifically, the first layer has 8.320 neurons, the second 33.024, the third 32.846, and the latest 129 neurons, for a total of 77.881 trainable parameters, with a forward/backward pass size of 0.01 MB. These size parameters have been chosen considering similar problems we encountered previously, with special care in keeping the number of parameters bounded. The MLP network is trained via adaptive mini-batch gradient descent, with a weight decay strategy [23], in order to prevent the network from overfitting the training data as well as the exploding gradient problem.

Since we cast an occupancy binary problem, given the input feature set $F = S(x,t) \cup S(e,t) \cup S(h,t)$ at timestep t, the model has to predict a binary label $p_t \in \{0,1\}$ corresponding to an empty and non-empty (*i.e.*, occupied) office, respectively. The discriminant function for such a task can be learned by minimizing binary cross-entropy (BCE), defined as follows:

$$BCE(y,p) = -\frac{1}{T}\sum_{t=1}^{T} y_t \cdot \log(p_t) + (1-y_t)\cdot \log(1-p_t) \quad (4)$$

Here, y_t is the target value (either 0 or 1). The prediction p_t can be any value between zero and one, indicating the confidence in that sample being positive. Predictions are often normalized to match a probability via the 1-bounded logistic function, often known as the sigmoid function.

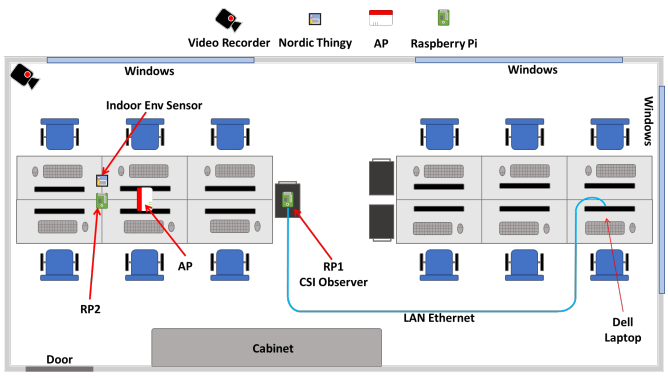

Fig. 2. Overview of the data collection environment.

This model is capable of learning complex, non-linear relationships [24], it comes with powerful interpretability tools, and is very fast, as we will see later. Importantly, the model can be modified to be fed with a subset of signals, for example, the CSI signal $S(x,t)$ only. This will allow us to make comparative experiments, and compare the relative importance of the different features.

Interpretability of the model: As an additional contribution of our paper, we propose the use of an XAI (Explainable AI) technique named Grad-CAM [17], which relies on the gradients of the loss function to compute importance weights associated to the input features (*i.e.*, CSI subcarriers, humidity, or temperature). This allows us to understand what input features the network pays the most attention to when taking a decision for a specific class.

Intuitively, this method is a post-hoc attention mechanism, since it works on the already-trained neural network. In particular, the Grad-CAM approach has been proved to pass the "sanity check" for saliency-based interpretability approaches [25], which means that it is *guaranteed* to express the relationships between input features and outputs present in the data on the specific deep learning model that has been adopted, in our case the MLP. Grad-CAM essentially computes the importance weights as the average of the gradients flowing down from a class label throughout the different layers until the input features.

To this end, we start by computing the hidden importance coefficient of Grad-CAM α_k^c as follows:

$$\alpha_k^c = \frac{1}{N} \sum_d \frac{\partial y^c}{\partial A_d^{(k)}} \tag{5}$$

where $A_d^{(k)}$ is the hidden feature map at layer k (*i.e.*, the output of the neurons of the layer k), d represents the number of hidden neurons at the layer k and c the class. Specifically, the weight α_k^c represents the partial linearization of the deep network downstream from A and captures the value of the feature map of the k-th layer for a target class c. Finally, we perform a weighted sum between the value just calculated and the feature maps $A_d^{(k)}$, followed by a ReLU to reset the negative values of the gradient to zero:

$$L_{GradCAM}^c = ReLU(\sum_k \alpha_k^c A_d^{(k)}) \tag{6}$$

The last operation is adopted for numerical stability.

Dimensionality and speed of the model: Our network is designed using the PyTorch Lightning framework, resulting in a model size of 15.18 KiB, with a RAM occupancy of 23.04 KiB being easily deployable over a resource-constraint device such as Nucleo-L432KC. Once trained, the model is capable of an inference time of 10.781 ms for each sample with the full amount of input features (CSI, temperature, humidity).

V. Experimental results

The Section is organized as follows: in Section V-A we report the profiling of the data collected with our setup discussed in Section IV-A; in Section V-B we report the results of the occupancy detection problem and show how our system performs better against three comparative approaches; in Section IV-B we report the interpretability study and, finally, in Section V-D, we show that CSI not only is capable of detecting the presence of people in a room but also helps to assess humidity and temperature, demonstrating the richness of the CSI signal and the goodness of the proposed model.

TABLE II
Simultaneous subject's presence distr. in terms of data samples.

Occupancy	Empty = 0	Occupied = 1			
Occupants	Zero	One	Two	Three	Four
# Samples	3389840	986180	569480	332440	84400
(%)	(63.2%)	(18.4%)	(10.6%)	(6.2%)	(1.6%)
5362340	3389840 (63.2%)	1972500 (36.8%)			

A. Data profiling

From January 04, 2022, at 15:08:40 hs to January 07, 2022, at 17:38:40 hs, for a total of 74 hours (*i.e.*, 268117 seconds with 32174040×66 samples) we collected a 10 GB dataset whose format is shown in Table I. As features, we focus here on timestamp, CSI amplitude (*i.e.*, 64 subcarriers), humidity, and temperature measurements. As for occupancy detection, six subjects (two women and four men) entered and used the office. They were made aware of the data monitoring system and told to accomplish their office activities as usual. Notably, the subjects worked freely in the room, moving chairs, raising/lowering curtains, and moving without a predefined pattern. Table II presents the distribution of (simultaneous) subjects' presence in the environment. Overall, 63.2% of the dataset represents the empty environment; the remaining 36.8% represents the environment with at least one subject in it.

The collected data undergoes a time series analysis composed of the following steps. Initially, we control for null values or duplicates present at the same t. Afterward, we analyze the data distribution of $S(x,t)$, $S(h,t)$, and $S(e,t)$ both visually and numerically. As a second step, we test the time series for stationarity using a statistical unit root test, namely the Augmented Dickey–Fuller (ADF) test [26]. The result shows that all the time series treated in this problem are stationary, so the correlation analysis can be performed on the raw data, without any other preprocessing (*e.g.*, detrendization) [27]. To do this, we rely on Pearson's ρ coefficient, defined as:

$$\rho = \frac{cov(X,Y)}{\sigma_x \sigma_y} \tag{7}$$

where, given a pair of random variables (X,Y), cov is the covariance, σ_x is the standard deviation of X, and σ_y is the standard deviation of Y. Notably, we report that temperature and humidity have a positive correlation of 0.45. The temperature has a correlation of 0.44 with respect to the binary occupancy variable, while humidity correlates at 0.35. As for the subcarriers, they are mostly correlated with neighboring subcarriers and the mid-to-high band carriers (*i.e.*, $a15$ to $a28$ and $a48$ to $a64$) are somewhat correlated with temperature and humidity ($\sim$ 0.20 to 0.30). An obvious point is that the time as a feature is strongly correlated (*i.e.*, 0.77) with the

TABLE III
START TIME, END TIME, NUMBER OF SAMPLES, MIN/MAX TEMPERATURE AND HUMIDITY FOR THE TRAINING (0) AND TESTING (1 TO 5) FOLDS.

Fold	Start	End	Empty	Occupied	T	H
0	04/01 15:08	06/01 19:16	2348151	1405500	18.72/40.09	16/49
1	06/01 19:16	06/01 23:44	321742	0	20.36/23.90	20/45
2	06/01 23:44	07/01 04:12	321742	0	18.86/21.80	25/42
3	07/01 04:12	07/01 08:41	321742	0	18.68/20.80	25/43
4	07/01 08:41	07/01 13:09	56223	265519	18.38/22.10	22/43
5	07/01 13:09	07/01 19:16	0	321741	20.19/31.60	20/38

environmental data. This is because temperature and humidity strictly depend on the heating system and on human presence. The former, since the office presents a heating system that activates and deactivates automatically, and the latter, since humans modify the environment (*e.g.*, modify the heating system setup, open/close windows or doors, and there is also the effect of body temperature and breathing).

B. Occupancy detection task

To evaluate the performance of our model, the dataset is divided into train and test sets, taking particular care in the division of the test set. In temporal order, the train set represents 70% of the collected data, and the test set the remaining 30%. The test set is further divided into five folds, representing different scenarios over time. This allows us to evaluate the model for general performance and test generalization abilities on unseen days separately. This is to stress the fact that when evaluating over different folds, the train set never changes, and the models are never re-trained. Table III presents the train/test split of the collected data.

Our model is trained for 10 epochs with a learning rate of $5e^{-3}$ on an NVIDIA GeForce RTX 3090 GPU. In addition to our model, we used two well-known ML models, Logistic Regression and Random Forest (RF), implemented using the Scikit-learn framework as a baseline to compare with the MLP model results. The results for the occupancy detection task are reported in Table IV. The tested models are trained on three different subsets of the collected dataset: *i)* only CSI data, *ii)* only environment (Env) data (*i.e.*, humidity and temperature), and *iii)* CSI and environment (C+E) data. In particular, the Logistic Regressor is a linear classifier whose results demonstrate that it is not easy to describe the intricate relationships of data in a linear manner.

Instead, the RF is a non-linear ensemble model based on decision trees, famous for its ability to resist overfitting, which achieves excellent performance. The proposed MLP model also achieves remarkable performance. These results show how the non-linear models (RF and MLP) are capable of properly using the CSI data, thereby reaching higher classification scores.

TABLE IV
OCCUPANCY DETECTION ACCURACY (IN %) OVER THE 5 TESTING FOLDS COMPARING THREE DIFFERENT ML MODELS.

	Logistic Regressor			Random Forest			MLP		
Fold	CSI	Env	C+E	CSI	Env	C+E	CSI	Env	C+E
1	68	99	76	99	100	99	100	99	92
2	71	100	72	100	100	100	100	100	99
3	77	100	86	99	100	100	100	100	100
4	94	18	86	88	75	88	83	54	65
5	96	31	91	100	100	100	100	99	99
Avg.	**81**	**70**	**82**	**97**	**95**	**97**	**97**	**90**	**91**

However, we specify that the use of the MLP model is preferable (compared with RF) because:

- RF is computationally and space-intensive (*i.e.*, does not allow real-time operation and deployment on embedded boards);
- an MLP model can be trained continuously. There is no need to use the whole dataset again but only new data, which can also arrive in real-time, thus doing online training;
- an MLP can be extended easily by someone who wants to work on top of this problem.

In turn, this shows that using only environmental data is insufficient to optimally solve occupancy detection, while the CSI data contains sufficient information to agglomerate the possible impact of temperature and humidity. In addition, we specify that, if we used only time as a feature for our analysis, the performance in terms of accuracy does not present good results (*i.e.*, 89.3%) compared with those of the MPL in Table IV.

C. Utilizing Grad-CAM to explain model outputs

The plot in Figure 3 depicts the importance of each feature (*i.e.*, CSI subcarriers in yellow background, environment in red) given to the MLP classifier for the occupancy prediction as explained in IV-B.

Interestingly, we found that temperature and humidity have no importance (values close to 0, if not negative in the plot) while the highest importance values locate between low frequencies (subcarriers $a9$ to $a17$) and high frequencies (subcarriers $a57$ to $a60$), confirming that paying attention to humidity and temperature does not help the task while paying attention to CSI does.

D. Humidity and temperature prediction

Inspired by the previous results, we perform multiple regression analyses to see if we can estimate humidity and temperature starting from CSI data. To do this, we fit a least-squares solution, both using linear regression (ordinary least squares) and non-Linear regression (minimization of a squared error objective), implemented with our neural network model.

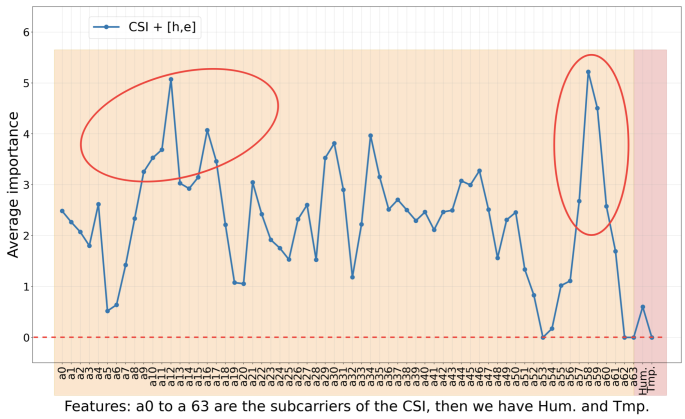

Fig. 3. Results of the XAI method, depicting the importance over all the features using CSI, humidity (h), and temperature (e).

TABLE V
MAE/MAPE RESULTS OF LINEAR AND NEURAL NETWORK REGRESSION MODELS ON THE HUMIDITY (H) AND TEMPERATURE (T) PREDICTION.

	Linear Regressor		Neural Network	
Fold	MAE (T/H)	MAPE (T/H)	MAE (T/H)	MAPE (T/H)
1	2.72/2.47	12.65/7.11	1.04/3.74	4.18/11.26
2	1.87/1.65	9.24/4.86	0.56/7.30	2.82/21.98
3	3.57/2.84	18.17/8.25	0.73/6.08	3.72/18.55
4	6.04/6.92	29.38/20.51	3.88/3.44	18.59/10.46
5	8.08/7.51	35.94/25.89	3.81/2.55	16.94/9.54
Avg.	**4.46/4.28**	**21.08/13.32**	**2.39/4.62**	**9.25/14.35**

The goal of this analysis is to show that sufficient information regarding the environmental factors is contained in the feature set of CSI signals, which would therefore allow us to gain a good understanding of the environment given only the amplitude values of CSI. We confirm such a result in Table V.

The fact that MLP (which is a non-linear method) performs definitely better than the linear regressor shows that the variation of temperature and humidity inside the room is mostly reflected by CSI data in a non-linear fashion.

This result allows also understand why the non-linear classification models do not perform better when using the CSI + Env data: the latter represents a redundant feature and redundancy is widely known to negatively affect the performance of any classifier [28].

VI. CONCLUSIONS

In this paper, we presented a highly accurate (*i.e.*, 97%), resource-efficient deep learning-based solution (*i.e.*, model size of 15.18 KiB, and inference time of 10.781 ms) for the device-free occupancy detection task in an unconstrained indoor environment. Our analysis shows that CSI amplitude data over time is highly predictive for occupancy detection. Furthermore, our solution uses a non-linear model capable of describing the agglomerate impact of temperature and humidity incorporated in the CSI signal. This aspect was further verified through a multiple regression analysis where humidity and temperature are estimated from CSI data, which confirms that sufficient information regarding the environmental factors is contained in the CSI signals. Finally, based on a consolidated algorithm of explainable AI, we qualitatively showed the validity of our claims.

For future work, we intend to design an ML model that simultaneously performs occupancy detection and activity recognition, with a particular emphasis on finding those activities which can be reliably detected.

REFERENCES

[1] B. N. Silva, M. Khan, and K. Han, "Towards sustainable smart cities: A review of trends, architectures, components, and open challenges in smart cities," *Sustainable Cities and Society*, vol. 38, pp. 697–713, 2018.

[2] S. Khoche, G. Sasirekha, J. Bapat, and D. Das, "Near real-time occupancy detection for smart building emergency management: a prototype," in *2020 IEEE international symposium on smart electronic systems (iSES)(Formerly iNiS)*. IEEE, 2020, pp. 115–120.

[3] C. Tang, W. Li, S. Vishwakarma, K. Chetty *et al.*, "Occupancy detection and people counting using wifi passive radar," in *2020 IEEE Radar Conference (RadarConf20)*. IEEE, 2020, pp. 1–6.

[4] L. Capogrosso, G. Skenderi, F. Girella, F. Fummi *et al.*, "Toward smart doors: A position paper," *arXiv preprint arXiv:2209.11770*, 2022.

[5] B. Qolomany, A. Al-Fuqaha, A. Gupta, D. Benhaddou *et al.*, "Leveraging machine learning and big data for smart buildings: A comprehensive survey," *IEEE Access*, vol. 7, pp. 90 316–90 356, 2019.

[6] G. Skenderi, A. Bozzini, L. Capogrosso, E. C. Agrillo *et al.*, "Dohmo: Embedded computer vision in co-housing scenarios," in *2021 Forum on specification & Design Languages (FDL)*. IEEE, 2021, pp. 01–08.

[7] M. Boldo, N. Bombieri, S. Centomo, M. De Marchi *et al.*, "Integrating wearable and camera based monitoring in the digital twin for safety assessment in the industry 4.0 era," in *International Symposium on Leveraging Applications of Formal Methods*. Springer, 2022.

[8] A. Verma, S. Prakash, V. Srivastava, A. Kumar *et al.*, "Sensing, controlling, and iot infrastructure in smart building: A review," *IEEE Sensors Journal*, vol. 19, no. 20, pp. 9036–9046, 2019.

[9] F. Demrozi, N. Serlonghi, C. Turetta, C. Pravadelli *et al.*, "Exploiting bluetooth low energy smart tags for virtual coaching," in *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*. IEEE, 2021, pp. 470–475.

[10] D. R. Beddiar, B. Nini, M. Sabokrou, and A. Hadid, "Vision-based human activity recognition: a survey," *Multimedia Tools and Applications*, vol. 79, no. 41, pp. 30 509–30 555, 2020.

[11] H. Zou, Y. Zhou, J. Yang, W. Gu *et al.*, "Freedetector: Device-free occupancy detection with commodity wifi," in *2017 IEEE International Conference on Sensing, Communication and Networking (SECON Workshops)*. IEEE, 2017, pp. 1–5.

[12] H. Zou, Y. Zhou, J. Yang, and C. J. Spanos, "Device-free occupancy detection and crowd counting in smart buildings with wifi-enabled iot," *Energy and Buildings*, vol. 174, pp. 309–322, 2018.

[13] F. Demrozi, F. Chiarani, and G. Pravadelli, "A low-cost ble-based distance estimation, occupancy detection and counting system," in *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2021, pp. 1430–1433.

[14] Z. Wang, B. Guo, Z. Yu, and X. Zhou, "Wi-fi csi-based behavior recognition: From signals and actions to activities," *IEEE Communications Magazine*, vol. 56, no. 5, pp. 109–115, 2018.

[15] X. Wang, C. Yang, and S. Mao, "On csi-based vital sign monitoring using commodity wifi," *ACM Transactions on Computing for Healthcare*, vol. 1, no. 3, pp. 1–27, 2020.

[16] Z. Chen, L. Zhang, C. Jiang, Z. Cao *et al.*, "Wifi csi based passive human activity recognition using attention based blstm," *IEEE Transactions on Mobile Computing*, vol. 18, no. 11, pp. 2714–2724, 2018.

[17] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam *et al.*, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[18] J. Yang, H. Zou, H. Jiang, and L. Xie, "Device-free occupant activity sensing using wifi-enabled iot devices for smart homes," *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 3991–4002, 2018.

[19] X. Zhang, R. Ruby, J. Long, L. Wang *et al.*, "Wihumidity: A novel csi-based humidity measurement system," in *International Conference on Smart Computing and Communication*. Springer, 2016, pp. 537–547.

[20] J. Li, A. Sharma, D. Mishra, and A. Seneviratne, "Fire detection using commodity wifi devices," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 1–6.

[21] Q. Dai, Y. Huang, L. Wang, R. Ruby *et al.*, "mm-humidity: Fine-grained humidity sensing with millimeter wave signals," in *2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, 2018, pp. 204–211.

[22] M. Schulz, D. Wegemer, and M. Hollick. (2017) Nexmon: The c-based firmware patching framework. [Online]. Available: https://nexmon.org

[23] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=Bkg6RiCqY7

[24] S. Billings, H. Jamaluddin, and S. Chen, "Properties of neural networks with applications to modelling non-linear dynamical systems," *International Journal of Control*, vol. 55, no. 1, pp. 193–224, 1992.

[25] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow *et al.*, "Sanity checks for saliency maps," *Advances in neural information processing systems*, vol. 31, 2018.

[26] Y.-W. Cheung and K. S. Lai, "Lag order and critical values of the augmented dickey–fuller test," *Journal of Business & Economic Statistics*, vol. 13, no. 3, pp. 277–280, 1995.

[27] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*. OTexts, 2018.

[28] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification and scene analysis*. Wiley New York, 1973, vol. 3.