

Investigating Best Approaches for Activity Classification in a Fully Instrumented Smarthome Environment

A. Mihoub, S. Zidi, and L. Laouamer

Abstract—Analyzing human behavior in smart environments is an important research area dealing with a multitude of issues related to ubiquitous computing, machine learning and ambient assisted living. With recent advancements in sensing technologies, it is henceforth possible to build computational models that select relevant sensors, and apply statistical models for accurate detection of residents' activities in smarthomes. To this end, we choose to work with the "Orange4home" dataset which represents one of the latest dataset in this research field. The main contribution of this paper is (1) to perform accurate detection of resident activities (extracted from the "Orange4Home" dataset) by proposing relevant preprocessing and machine learning approaches, and (2) enhance previous classification results already published on the same dataset. Thus, our methodology in this paper is to explore the whole process from data preprocessing to classification metrics. Indeed, a carefully designed and original preprocessing algorithm is proposed in order to properly prepare data to the training phase. Then, to perform relevant exploration of the feature space, many strategies for features selection and reduction (based on Univariate feature selection and Principal Component Analysis) were proposed. For the activities classification task, many well-chosen discriminative models (SVMs, Decision Trees, Random Forests) were explored. Our main results outperform previously published results on the same dataset. Moreover, comparing all proposed classifiers, Random Forests outperform other classifiers and shows that the optimal accuracy rate (95%) was obtained thanks to a smart choice of a limited number of sensors rather the use of the full feature space (i.e. data from all installed sensors). Based on our results, many recommendations (for building optimal smarthomes and activity classification models) were emphasized in the end of this paper.

Index Terms—Activity classification, data preprocessing and feature selection, discriminative models, smarthome.

I. INTRODUCTION

Modeling human activities in ambient intelligent environments is becoming an attractive research field and there is a growing awareness about its technological, scientific and socioeconomic challenges [1]. Actually, human behavior modeling and analysis can be developed for a myriad of applications such as robotics [2], [3], skills assessments [4]–[6], sports analytics [7], [8], learning analytics [9], human-human interaction analysis [10], [11], affective computing [12], healthcare [13]–[15], ambient

assisted living [16]–[18], surveillance and entertainment in smarthomes [19].

In particular, smart home environments, equipped with multiple sensors, have given the opportunity to model Activities of Daily Living (ADLs) with an accurate sensing and a deep understanding of the human behavior. The concept of ADLs [20], [21] is largely used in health care, describing activities and daily routines, such as eating, cooking, sleeping, bathing, etc. In fact, ADLs monitoring in smart homes represents a great instrument to track functional status of residents and assess their ability to live independently [20]. Thus, one major goal of these augmented houses is to assist disabled and elderly people in order to enhance their well-being as well as their independent and decent living [16], [22]. In addition to persons with special needs, analyzing and modeling behaviors in smart homes, comes with a multitude of benefits for all the occupants. Indeed, it helps to improve house security, reduce energy costs, ameliorate entertainment and enhance comfort within the house [19].

Our challenge in this paper is to build computational models that select relevant sensors and apply statistical models to detect accurately ADLs in a smart home environment. The end goal of this research is to build user-aware homes able to automatically respond to human needs, behavior and activities. To reach this goal, we choose to work with a new dataset specially designed for the smart home research field. This dataset called "Orange4home" [23] comes with many advantages comparing to existing databases (c.f. subsection III.A).

In this paper, our approach tries to mine the whole process of ADLs recognition: from applying many datamining methods for data preprocessing and selection, to the assessment of a wide range of machine learning models relying on a multitude of evaluation criteria. The proposed computational models should be able to analyze multimodal environmental data, analyze subtleties of human behavior and then detect accurately ADLs patterns of the smarthome occupants.

The rest of the paper is organized as follows: the next section reviews the state of the art of ADLs recognition models. In Section III, we introduce the used smarthome dataset and then we detail our proposed algorithm for data preprocessing. In Section IV, we expose applied strategies for data selection and reduction. Machine learning models and how they were applied are presented in Section V. Classification results and main findings are exposed in Section VI. Section VII summarizes our contributions and concludes the paper.

Manuscript received November 2, 2019; revised December 14, 2019.

The authors are with the Department of Management Information Systems, College of Business and Economics, Qassim University, P.O. Box: 6640, Buraidah: 51452, Saudi Arabia (e-mail: a.mihoub@qu.edu.sa, s.zidi@qu.edu.sa, l.laouamer@qu.edu.sa).

II. RELATED WORK

ADLs detection is a classification problem [20]. It consists of mapping sensors data to relevant activities. This recognition task can be categorized based on data modalities of the environment devices. Two types are widely explored: video-based [24] or sensors-based recognition [25]. For the second type, they can be either wearable or ambient sensors. Smartphones, smart watches, smart glasses and helmets are typical examples of wearable devices [26]. Ambient sensors are installed within the environment such as energy consumption sensors or on separate objects such as doors sensors with the aim of catching all possible human interactions [21], [23].

But before going into any classification process, a preprocessing step is required for raw data to get the final features [27]. It may consist of filtering data noise or redundancies, dealing with missing data, rescaling data values, changing data types, selecting relevant features, extracting new features [28]-[30], reducing feature space etc. [31]. Afterwards, classification step consists of mapping observations (features) into activities (classes).

ADLs classification algorithms span over a broad range of machine learning techniques [32] from classic classifiers such as Support Vectors Machines (SVMs) [33] to more sophisticated approaches such as Hidden Markov Models (HMMs) [34]. These machine learning models can be categorized into two types: discriminative models and generative models [35]. Discriminative approaches model the conditional probability of classes given data observations while generative ones model the joint probability of data observations and their corresponding classes.

SVMs and Random Forests [36] are typical examples of the first type. In particular, SVMs [37] are considered one of the most powerful discriminative algorithms that were applied to diverse classification problems including ADL recognition [38]. In fact, SVMs, are known to be particularly efficient to cope with high dimensional data spaces [20]. However, the problem with SVMs, is the fact of being frame-based classifiers i.e. they are inherently unable to model temporal dependencies. Additionally, high computational complexity required in training phase may in some cases limit their applicability. In ADL literature, SVMs have been applied in [16] to detect activities of an elderly person using wearable sensors. These sensors recorded basic signals such as acceleration, altitude and temperature. Many features were extracted and transformed from this raw data such as mean or standard deviation of accelerations in different axes. Coupled with many feature selection techniques (e.g. feature clamping [39]), SVMs achieved the best classification rate (90%) compared to other classifiers namely MLP (Multi-Layer Perceptron) and RBF (Radial Basis Function).

Random Forests (RFs) [36] is another well-known discriminative technique for activity recognition. It belongs to the Ensemble Learning methods [40] since it builds a set of Decision Trees (DTs) [41] and merge their outputs according to a voting system. This procedure preserves decision trees advantages but prevents overfitting problems and ends up

with a more accurate and stable classification result. However, one limitation of RFs is the slowness of the model in some cases especially with a high number of decision trees. Random Forests were applied in [21] in order to enhance daily living of elderly people and improve their security. Passive infrared (PIR) sensors were used to detect motion inside a smart apartment and a set of ADL were collected manually from the residents. For ADL recognition, RFs outperformed all tested models which were basically SVMs and Naïve Bayes classifiers. Note that, other discriminative models were also explored in ADL classification literature such as Conditional Random Field (CRFs) [42], [43] and Artificial Neural Networks (ANNs) [44], [45].

HMMs [46] are another type of models that are known to be part of the Probabilistic Graphical Models (PGMs) [42] and especially part of the generative ones. In spite of having less discriminative power than SVMs and other discriminative classifiers, they have the advantage of modeling efficiently sequences and temporal data due to their internal network configuration [10]. This property has made HMMs very popular in the activity recognition literature [20]. For instance, HMMs were applied in [47] for multimodal behavior modeling in a smart living room environment. Many Features were extracted from a 3D video tracker, a speech activity detector (for occupants) and a sound detector (for the environment). HMMs were designed to detect specific situations like "aperitif" or "siesta" and classification results were quite interesting. Other activity recognition studies applying HMMs can be found in [22], [48]-[50].

One Major problem with generative models (e.g. HMMs) is online classification. This type of model may be efficient for offline classification, but for real time applications, discriminative frame-based classifiers represent a better solution, especially with their discriminative power and their ability to cope with high dimensional data. For these reasons, in our work, we propose to develop, evaluate and compare several discriminative models widely used in the literature of ADLs recognition which are: SVMs, Decision Trees and Random Forests. These models were applied to one of the newest datasets in the field of smarthomes: the "Orange4Home dataset" [23]. The advantages of this dataset are listed in the next section. A main goal of this work is to enhance previous recognition rates obtained in a previous study for the same dataset and the same classification task [51]. Thus, our main contributions are as follows:

- The Enhancement of previous classification results: our methodology to reach this goal is to explore and mine the whole process of ADLs recognition task.
- The introduction of a dedicated preprocessing algorithm (cf. Fig.) for the "Orange4Home" dataset: this algorithm proposes detailed and precise steps for data preparation including data fusion and many suited strategies to deal with missing values and data normalization. Note that this algorithm can be easily generalized to other datasets disposing similar data format.
- The exploration of several strategies to get a deep mining of the feature space: these strategies are based on a combination of data selection and reduction algorithms

especially ANOVA [52] and Principal Component Analysis (PCA [53]).

- The development of a multitude of well-chosen machine learning approaches (SVMs, DTs, RFs) and their assessment according to several metrics.
- The proposition of recommendations for building smarthomes and ADLs classification models deduced from our results and findings.

In the next section, we present the used dataset and then we detail all steps of our data preprocessing algorithm.

III. DATA PREPROCESSING ALGORITHM

A. Dataset Presentation

Our proposed models are applied on a new ADLs dataset called "Orange4Home dataset" [23]. Actually, numerous datasets exist in the literature of ADLs such as Opportunity [54], Transfer Learning [55] and ARAS [56]. However, many remarks can be drawn on these datasets. For instance, the Opportunity dataset was recorded in only one room, and captured sequences were very limited in time (nearly 30 minutes). Moreover, body-worn sensors represent half of used devices which is nonrealistic for normal population. Transfer Learning dataset comes with 3 different homes recordings. Nevertheless, limited number of sensors were used which lessens the capacity to sense the complexity of human behavior. In addition, it labels only eight classes of activities, which may be too simple to model effectively human routines. ARAS dataset contains recorded activities over one full month in two separate real houses. It comes with 27 different classes of activities tagged in situ by the resident with a remarkable degree of accuracy. Unfortunately, each house is equipped with only 20 binary sensors which constrains future experiments with heterogeneous data processing algorithms.

"Orange4Home dataset" [23] due to its original experimental protocol, combine advantages of existing datasets and represents a great benchmark for behavior modeling and activity recognition. "Orange4Home" is the result of a joint cooperation between Orange Labs and INRIA¹ (France). It provides multimodal data collected from 236 heterogeneous sensors in a fully equipped two-story apartment called "Amiqua4Home"² (cf. Fig. 1, extracted from [23]). Orange4Home comes with approximately 180 hours of recording activities of daily living distributed over 4 consecutive weeks of work (i.e. 20 working days, each one from 8.00 am to 5.00 pm). A total number of 24 activity classes was accurately and manually labeled in situ by the single occupant of the smart apartment.

The exhaustive list of classes is exposed in Table I. Each class is denoted by the place in which activity occurs followed by the activity label (e.g. Living_room|Eating). Note that we have added the "Inter" activity class to original dataset. This latter activity can be seen as a transition activity from one activity to another. Therefore, for this work, the total number

of classes is rather 25 classes. For data, many types of sensors were used to sense different kinds of information such as door opening, luminosity, presence, water (hot and cold) consumption, energy consumption, electrical data, light switches, temperature, noise, weather data, etc. These sensors generate many types of data (binary, integer, real number and categorical) as shown in Table II (table extracted from [23]). "Orange4Home" realistic sensed data and accurate labeled classes make it an appropriate dataset to evaluate data preprocessing techniques, activity recognition models as well as assessing other smarthome research problems as we will see in the next section.

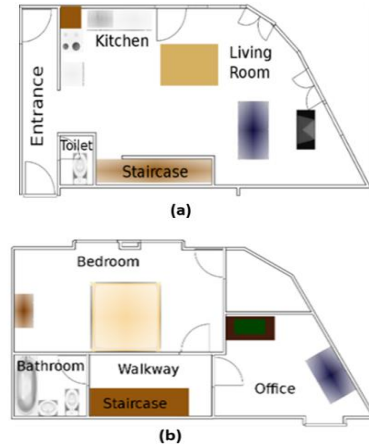


Fig. 1. Layout of Amiqua4Home apartment: (a) ground floor (b) first floor (extracted from [23]).

TABLE I: LIST OF CLASSES (EXTRACTED FROM [23]): EACH CLASS IS DENOTED BY THE PLACE IN WHICH ACTIVITY OCCURS FOLLOWED BY THE ACTIVITY LABEL

Entrance Entering	Entrance Leaving
Kitchen Preparing	Kitchen Cooking
Kitchen Washing_the_dishes	Kitchen Cleaning
Living_room Eating	Living_room Cleaning
Living_room Computing	Living_room Watching_TV
Staircase Going_up	Staircase Going_down
Office Computing	Office Watching_TV
Office Cleaning	Bedroom Dressing
Bedroom Reading	Bedroom Napping
Bedroom Cleaning	Toilet Using_the_toilet
Bathroom Showering	Bathroom Using_the_sink
Bathroom Using_the_toilet	Bathroom Cleaning

TABLE II: NUMBER OF SENSORS PER PLACE AND DATA TYPE. THESE FIGURES WERE EXTRACTED FROM [23]

Place/Type	Data				Total
	Binary	Integer	Real	Categorical	
Entrance	3	1	2	3	9
Kitchen	13	21	18	0	52
Living room	16	6	8	7	37
Toilet	3	1	1	0	5
Staircase	3	0	0	0	3
Walkway	9	0	1	0	10
Bathroom	9	6	8	3	26
Office	9	3	3	5	20
Bedroom	17	4	6	7	34
Global	1	13	20	6	40
Total	83	55	67	31	236

¹ <https://www.inria.fr/en/>

² <https://amiqua4home.inria.fr/>

B. Preprocessing Algorithm

"Orange4Home dataset" contains 236 files representing each one sensor. Each file contains two columns: timestamp and sensor value at that time. Time granularity is different from one sensor to another. Our objective is to merge all data sensors to one table with sensor labels as columns indexes and timestamp as a row index. The Final table should not contain any missing values and should be suitably normalized. To reach this goal, we propose a specific preprocessing algorithm that chains essentially three big steps:

1. Data Fusion
2. Filling Missing Values
3. Data Normalization

Our preprocessing algorithm is summarized in Fig. 2. All details about the three steps are presented in the following subsections. Furthermore, note that our proposed algorithm can be easily applied to other smarthome datasets disposing similar sensors configuration.

1) Data fusion

As aforementioned, we target to fuse all data sensors to one table with sensor labels as columns indexes and timestamp as a row index. To reach this goal, a "union strategy" is applied for fusion. It consists of using the union of all timestamps from all sensor files, even if does not exist in some sensor files. Depending of sensors timestamps, some missing values may occur in some rows (c.f. Appendix A for an example of a possible scenario).

2) Filling missing values

To tackle missing values issue, many strategies were applied depending on the type of column values and the location of missing values in the table. We remind that each column represents a sensor from the smart apartment. The first step is to apply a "forward strategy". It consists of replacing a missing value with the nearest value from past. In fact, a sensor records a new time/value only if there is a change. Hence, it is quit relevant in this case to fill gaps with the last valid observation. After this step, missing values located at the beginning of columns remain untreated because of past observations absence. Here, we distinguish two cases: (1) If the type of sensor value is numerical, missing values are simply replaced by zero, since theoretically it wasn't recording that time, we call this strategy the "zero strategy". (2) If the type of sensor value is categorical, missing values are replaced using a "backward strategy". It consists of replacing a missing value with the nearest value from the future. We choose this strategy since change is unlikely to occur within few seconds (duration of the missing values at the beginning of table) especially with the categorical sensors. For instance, if we take the wind direction sensor, it is unlikely that direction changes totally from east to west within few seconds. Hereby, all missing values cases were carefully treated (c.f. Appendix B for an example of a possible scenario).

3) Data normalization

To prepare data to machine learning models, it is recommended (even required for some models) to normalize

data to a homogeneous format. To reach this goal, the following steps are applied: (1) A min-max scaler is applied on numerical values. After this step all numeric values are ranged between 0 and 1. (2) Binarization of two-categorical variables to Boolean Variables is applied (i.e. sensors with two-categorical values were changed to 0 and 1 values). For instance, for the "Kitchen light" sensor, "OFF" is changed to 0 and "ON" to 1. (3) "Hot encoding" is applied for multiple-categories columns (more than two categories).

"Hot encoding" manipulation consists of transforming one column with n possible categories to n columns. Each new column represents one category (c.f. Appendix C for an example of a possible scenario). This way, values of new transformed column will be only 0 or 1 (that category is observed or not). Therefore, we come out with a homogenous numerical table which all values are ranged between 0 and 1.

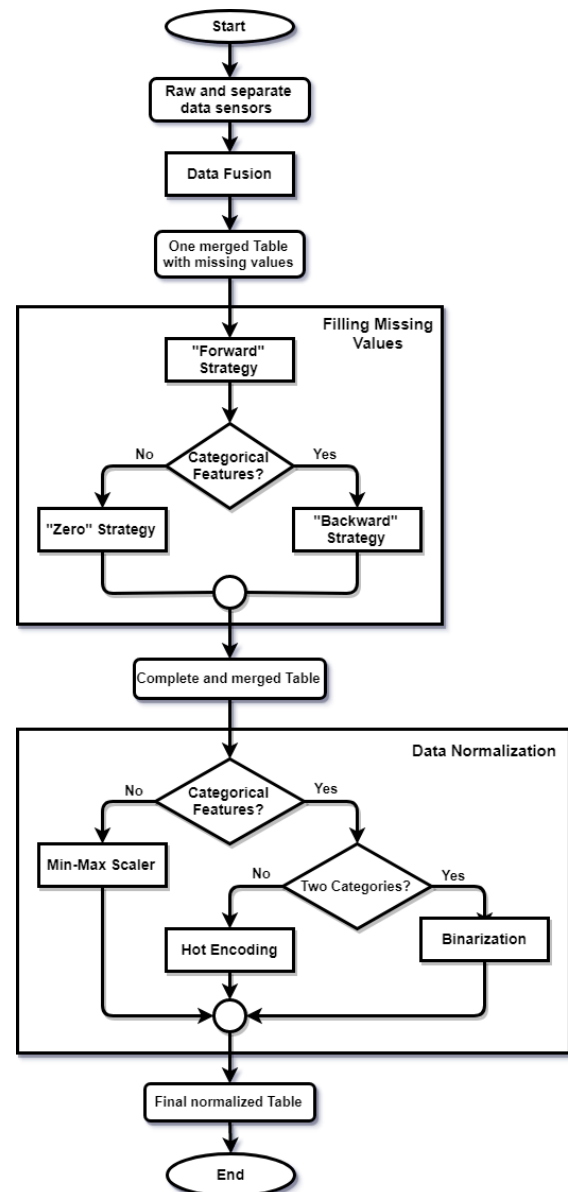


Fig. 2. Our proposed preprocessing algorithm. This algorithm is composed of three main steps: Fusion, filling missing values and normalization.

C. Output of the Preprocessing Algorithm

Initially, our dataset comes with 236 features, each feature representing one sensor. As explained before, many steps

were undertaken to preprocess data and prepare training phase, namely data fusion, dealing with missing and data normalization. The fusion step outputs one table with 236 columns (sensors) and nearly 224000 lines (timestamps). In that last step of normalization, "Hot encoding" is applied for multiple-categories features which increased features number from 236 to 265. Before applying selection and reduction, one last manipulation is applied: we deleted constant features. Constant features can be defined as features with variance equal to zero. Since they are constant all time, they do not have any discriminative effect and it better to neutralize them from the beginning. This way, 7 columns were deleted and the final dataset contained essentially 258 normalized features. Remember all preprocessing steps can be reviewed in the schema of our proposed preprocessing algorithm (cf. Fig. 2).

IV. FEATURES SELECTION AND REDUCTION

A. Applied Strategies

Once data table is ready, next issue is to select and/or extract relevant features in order to optimize the training step. Four strategies were explored:

- S1: Selection of all features i.e. the whole feature space is selected (258 features).
- S2: Selection of a set of features based on the "Univariate feature selection" method (cf. subsection IV.B for more details). Tested sets of features were 40, 50, 60, 70, 80, 100, 120 and 140.
- S3: Reduction of the feature space using PCA (Principal Components Analysis, cf. subsection IV.C for more details). Tested numbers of PCA components were 3, 5, 7, 9, 11, 13, 15, 17, 19 and 21.
- S4: Selection of a set of features based on the "Univariate feature selection" method (S2) + Reduction of the selected feature space using PCA (S3). Based on aforementioned sets for selection (S2) and reduction (S3), all possible combinations were tested. For instance, suppose 80 features are selected (an example of S2), S3 reduction strategy is applied on that feature space.

B. Univariate Feature Selection

Univariate feature selection concept consists of selecting the best features based on univariate statistical tests between each feature and the output variable. Univariate feature selection keeps only features with the highest scores according to ANOVA tests [52]. In fact, when the features are quantitative, which is our case, we can compute ANOVA F-value score between each feature and the target variable (i.e. classes labels). This way, selecting the best k features is simply selecting the k features having the highest ANOVA F-value scores.

C. Principal Components Analysis

Principal component analysis (PCA) [53] is a statistical approach for dimensionality reduction. PCA objective is to transform a set of possibly correlated features into a set of linearly uncorrelated features called principal components. Due to this transformation, the first principal component has

the largest possible variance which decreases with each succeeding component. Therefore, PCA reduces feature space to a lower space and allows to conserve maximum information and variability from original data.

V. CLASSIFICATION OF HUMAN ACTIVITIES

A. Applied Models

As explained before, discriminative frame-based classifiers represent a promising solution for our classification problem, especially with their capacity to deal with high dimensional data space and their effective discriminative power. Therefore, for our modeling, we proposed three discriminative models carefully-designed for our classification problem. These models are based respectively on SVMs, Decision Trees and Random Forests. In the next subsections, we briefly introduce the concepts behind these three models.

1) SVMs

Support vector machines (SVMs) are a powerful technique for solving classification problems. They have been also successfully applied for regression issues [57]. SVMs are based on two key principles: the maximum-margin hyperplane principle and the kernel function principle. These two principals were well-formalized for several years before they were pooled together to formally build SVMs in 1995 by Vapnik [58]. From this date, SVMs quickly gained popularity thanks to the theoretical guarantees that were provided, and the good results in many applications. The first key idea is the maximum margin. The margin represents the distance between the separation boundary and the closest data points samples. These points are called support vectors. The challenge here is to find the optimal dividing boundary (which maximizes the margin), from learning data. If the boundary is linear, this amounts to a quadratic optimization problem. When the data is linearly inseparable, the kernel function (second key idea of the SVMs) allows to transform the initial representation space of the data into a larger dimensional space, where it is likely to find a linear separator. This way, SVMs concepts help in solving efficiently nonlinear classification problems.

2) Decision trees

Decision trees are another popular method of supervised learning [41]. They have the advantage to be represented graphically in a tree like structure. Each node in a decision tree represents an input variable and each branch represents a value taken by that variable. In a classification problem, each leaf of the tree is associated with a class. To classify a new instance, simply traverse the tree, and associate it with the class it ends in. In spite of the apparent simplicity, several problems occur during the construction of the tree. There are essentially three methods widely used by the scientific community in trees construction: C4.5 algorithm [59], CART algorithm (Classification And Regression Tree) [60] and the CHAID algorithm (CHi-squared Automatic Interaction Detector) [61]. In the construction phase,

decision trees tend to form a quite complex model, excessively sticking to the data and thus causing over-fitting in some cases. In addition, it was shown that tree size increases proportionally with the size of the training set. To fix this issue, a pruning phase is applied in order to reduce and determine the optimal size of the tree. Note that there are other solutions for the over-fitting problem especially Random Forests [36].

3) Random forests

Random Forests (RFs) [36] is another well-known discriminative technique for class recognition. It belongs to the Ensemble Learning methods [40] since it builds a set of decision trees [41] and merge their outputs according to a voting system. This procedure preserves decision trees advantages but prevents over fitting problems and ends up with a more accurate and stable classification result. However, one limitation of RFs is the slowness of the model in some cases especially with a high number of decision trees. Next, we will present how these introduced classifiers were applied in our ADLs recognition problem.

B. Application Setting

We remind that tested classifiers were principally Support Vector Machines (SVMs), Decision Trees (DTs) and Random Forests (RFs). Before presenting all classifiers results, please note:

- For SVMs, one-vs-one strategy is used to deal with multiclass issue since our target variable have more than two classes (25 classes). The used kernel was Radial Basis Function (RBF) kernel [37].
- For each model, the four strategies of data selection/reduction were applied.
- For selection, tested sets of features were 40, 50, 60, 70, 80, 100, 120 and 140.
- For reduction, tested numbers of components were 3, 5, 7, 9, 11, 13, 15, 17, 19 and 21.
- For each model, regularization parameters were used to limit overfitting issues.
- Our dataset contains four weeks of recording. For all models, 4-cross validation is applied, it means in each iteration, three weeks are used for models training and one week for testing. The presented evaluation metrics are the average of all iterations of the 4-cross validation.
- Data preprocessing, selection/reduction techniques, models training/testing evaluation metrics, plots and figures were all implemented in Python using well known libraries for Data Science and Machine Learning such as Pandas, Numpy, Scikit-learn and Matplotlib [62].
- Scikit-learn library uses SVC algorithm (cf. [63] for more details) for SVMs implementation and an optimized version of the CART algorithm [60] for Decision Trees implementation. For RFs implementation, Scikit-learn library uses an enhanced version of the original Random Forests algorithm [36]. Note that in Scikit-learn, per default, all hyper-parameters were finely tuned based on literature review to probably get the best classification performance.

VI. RESULTS AND DISCUSSION

A. Comparison to Previous Work

For each model and for each selection/reduction configuration, several classification metrics were recorded. In this work we will focus especially on accuracy, recall, precision and F-measure [64]. Firstly, we start with comparing our classification rates to previous work in literature. In a previous study [51], two approaches were tested in order to classify activities of the Orange4Home dataset. Please note in that study, training/testing protocols may slightly differ from ours, thus, in our experiments, we tried our best to make the comparison as relevant as possible. The first previous approach uses all sensors in the recognition task. The best performance was obtained with an SVM classifier and an F-measure rate equal to 89.61%. In our work, this approach can be compared to the application of the S1 strategy (selection of all features, cf. S1 in subsection IV.A). Compared to our results, two of our classifiers outperform the latter result. Classification rates of our SVM and RFs classifier were respectively 91.50% and 92.82%. This performance gap, especially for the same used classifier (SVM), can be explained by the relevancy of the steps undertaken by our preprocessing algorithm.

The second approach presented in the previous study [51] relies on a multitude of place-based classifiers. One classifier was used per place in the smart apartment and a decision fusion step is needed to deduce the right class. Each one of those classifiers selects only relevant sensors for that specific place. The best performance was obtained with an MLP (Multi-Layer Perceptron) classifier and an F-measure rate of 93.05%. In our work, this approach can be compared to our selection strategies of the relevant sensors. As we will see next, in our application, Random Forests shows the best performance in "only features selection" strategy (i.e. S2 strategy) especially with a selected number of features equal to 50 features. In this configuration, recorded F-measure of RFs was 95.30% which outperforms the previous work (93.05%) [51]. This difference may be in part explained by the fact that our method selects best sensors over all sensors over all activities in the apartment not by place as used in [51]. Next we compare our three models in the different proposed selection/reduction configurations.

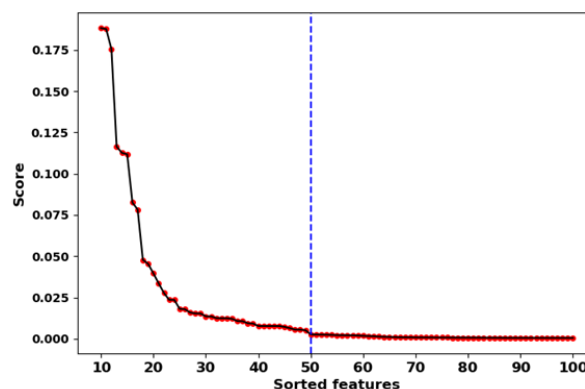


Fig. 3. Plotting features scores (F-value from ANOVA) sorted in a decremented order. This figure shows that scores begin to stagnate in a low level beyond the best 50th feature (blue dashed line).

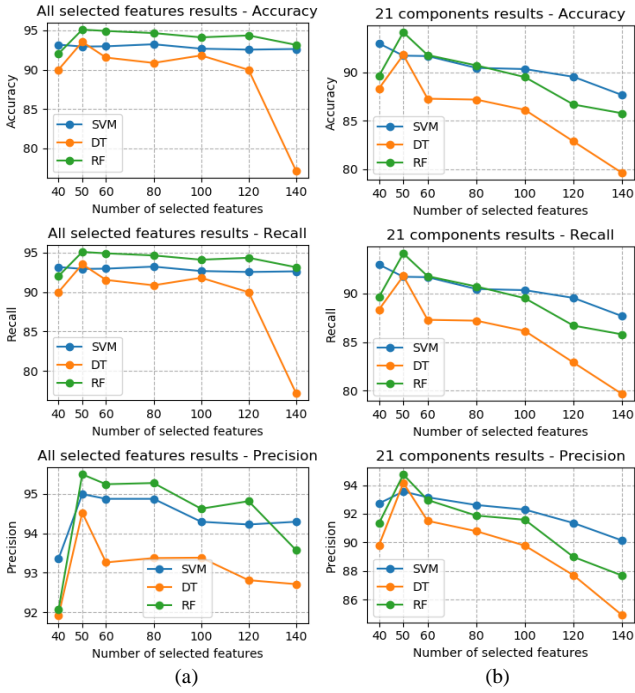


Fig. 4. (a) In this configuration, the three classifiers (SVM, DT, RF) used all selected features (40:140) without any reduction. (b) In this configuration, for each selected number, PCA is applied and 21 components were extracted.

TABLE III: ACCURACY RESULTS OF RFs AND SVMs USING DIFFERENT NUMBER OF FEATURES: 50 FEATURES, 140 FEATURES (BOTH WITHOUT REDUCTION), 140 FEATURES REDUCED TO 21 PCA COMPONENTS

Classifier	Number of Features		
	50	140	140 (reduced to 21)
RFs	95.09	93.15	85.79
SVMs	92.94	92.63	87.67

B. Comparison of Our Models and Discussion

In Fig. 4, accuracy, recall and precision results are presented for the three classifiers SVMs, DTs and RFs. In that Figure, two selection/reduction configurations are tested: in the first column, we have "only features selection" (cf. S2 in subsection IV.A) while in the second column, beside selection we apply PCA reduction (S4 strategy) to extract 21 components (21 was the biggest tested number of components). Random Forests shows the best performance in "only features selection" configuration especially with a moderate number of feature (e.g. 50, 60, 80 features). The best accuracy rate at all (95%) was given by RFs with only 50 selected features (c.f. Table III). We remind that best features were selected according to their F-value scores from ANOVA tests. Fig. 3 explores and plots all features scores sorted in a decremented order. This figure shows that scores begin to stagnate in a low level beyond the 50th feature which explains in part the best accuracy rate (95% given by RFs) using only 50 selected features. Moreover, the RFs high performances confirm many previous studies claiming that RFs are usually better than other classifiers especially in a multi-class problem [21], [65], [66]. However, when we increase the number of selected features (e.g.140 features), SVMs obtain similar results to RFs and even better for precision. This can be explained by SVM efficiency in high feature spaces [20]. Moreover, when we apply PCA reduction on a large feature space (e.g. 120, 140, c.f. Table III), SVMs clearly outperform RFs which loses its superiority even with

some lower selected features (e.g. 60, 80). This can be explained by a better capacity of SVMs to handle artificial extracted components comparing to RFs.

In Fig. 5, recall and precision results are presented for the two classifiers RFs and SVMs. This figure confirms that best results are obtained with the "only features selection" strategy i.e. without any reduction (cf. the blue curve). From this finding, we think that in building ADLs classification models for similar smarthomes and recognition tasks, it is highly recommended to work with the original preprocessed values of sensors (or at least extracted features from single sensors) than the extracted reduced artificial components. In this recommended specific configuration ("only features selection") and as seen in Fig. 4, we can see that 50 features are sufficient to get the best performance. This particular result is more obvious in Fig. 6 in which the orange curve - representing 50 selected features - is clearly above all other curves.

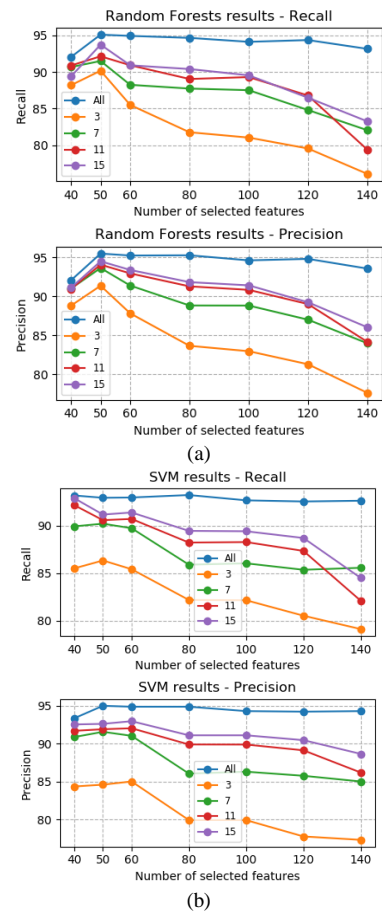


Fig. 5. RFs results (a) and SVMs results (b) for many selection/reduction strategies. Each color/curve represents the number of PCA components (for instance red color represents 11 components, note that 'all' means no reduction). X axis represents the number of selected features.

Indeed, this result shows that there is no need to have a fully instrumented smarthouse to have an effective classification of human activity. In fact, multiplying features does not add any discriminative power and may disturb classifiers. Thus, it is better to focus on a smart choice of sensors types and locations rather than multiplying sensing devices. For instance, in our smarthome dataset, if we explore the first best selected features/sensors (best F-value scores from ANOVA) we find principally light sensors (e.g.

"bathroom_light", "bedroom_light", "office_light") and energy consumption sensors (e.g. "livingroom_tv_plug_consumption", "office_tv_plug_consumption"). As a recommendation, for a similar setting to our smarthome and similar human activities to our resident activities, it is then quite judicious to explore this strategy of reducing full-instrumentation and optimizing sensors setting. In addition, besides having potentially high classification rates, this strategy presents many other advantages: it reduces smarthomes installation costs, cut down operational charges, limits technical problems, reduces data storage issues and mostly enhances resident's privacy since less devices are used to detect their behaviors. Therefore, this strategy can contribute significantly on solving well-known problems of smarthomes relating to costs and privacy [16], [19], [20].

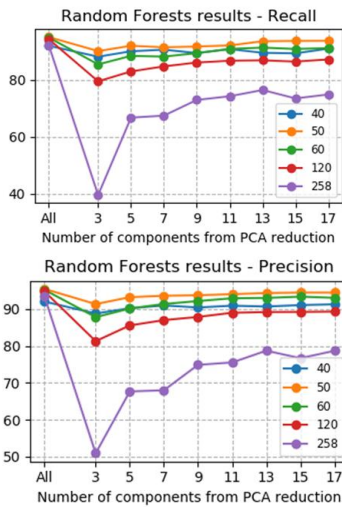


Fig. 6. RFs results (top: recall / bottom: precision) for many selection/reduction strategies. Each color/curve represents the number of selected feature (for instance orange color represents 50 selected features). X-axis represents the number of components from PCA reduction.

VII. CONCLUSION

In this research work, we have tried to propose the best process for automatic activities detection in a smarthome environment. To this end, we have used a new dataset called "Orange4home" specially designed for the smart home research field. We have managed to obtain consistent results by proposing efficient ADL classification process. First, a dedicated algorithm for data preprocessing was finely designed and applied. Second, we have built computational algorithm that selects the most relevant sensors. For this goal, four strategies of selection and reduction were explored. Third, for activities classification, three discriminative models (SVMs, Decision Trees and Random Forests) were tested and compared. As a result, we have reached attractive ADL recognition rates that outperform previous results in literature. Particularly, for our models, Random Forest outperforms other classifiers with an accuracy rate beyond 95%. Moreover, through the proposed features selection strategies, we have succeeded to determine the most relevant sensors, pointing that full instrumentation of houses does not necessarily give optimal results. This finding can be a first step toward solving some major smarthomes issues related to efficiency, costs and privacy. As a conclusion, this research work consists of exploring a multitude of strategies in order

to build smart ADL recognition models for the "Orange4home" dataset. It can be considered as a generic benchmark that may be used in future by other researchers in this field. The perspectives of this work revolve around three axes. First, we intent to enhance our preprocessing algorithm in order to consider other than ambient sensor types, such as vision sensors or wearables sensors. Second, since PCA results were not quite relevant, we plan to apply other promising techniques such as auto-encoders which ensure nonlinear transformations and thus, may overcome PCA limitations. The last axe concerns the application of deep learning classifiers which perform better automatic features learning, and then compare it to our proposed models in this paper.

APPENDIX

timestamp	office_luminosity	timestamp	bedroom_CO2	timestamp	global_wind_direction_ext
1/30/2017 8:48:44 AM	52	1/30/2017 8:48:44 AM	496.96	1/30/2017 8:53:44 AM	ENE
1/30/2017 8:51:27 AM	84	1/30/2017 8:53:44 AM	501.76		

timestamp	office_luminosity	bedroom_CO2	global_wind_direction_ext
1/30/2017 8:48:44 AM	nan	496.96	nan
1/30/2017 8:48:49 AM	52	nan	nan
1/30/2017 8:51:27 AM	84	nan	nan
1/30/2017 8:53:44 AM	nan	501.76	ENE

Appendix A: An example of a data fusion step: here three sensors files were merged in one table using the "union" strategy (keeping all observations). Notice that some missing values appear since timestamps differ from one sensor to another.

timestamp	office_luminosity	bedroom_CO2	global_wind_direction_ext
1/30/2017 8:48:44 AM	nan	496.96	nan
1/30/2017 8:48:49 AM	52	nan	nan
1/30/2017 8:51:27 AM	84	nan	nan
1/30/2017 8:53:44 AM	nan	501.76	ENE

Filling missing values

timestamp	office_luminosity	bedroom_CO2	global_wind_direction_ext
1/30/2017 8:48:44 AM	0	496.96	ENE
1/30/2017 8:48:49 AM	52	496.96	ENE
1/30/2017 8:51:27 AM	84	496.96	ENE
1/30/2017 8:53:44 AM	84	501.76	ENE

Appendix B: An example of applied strategies to cope with missing values. We remind that three strategies were used (cf. the "Filling Missing Values" subsection for more explanations): the "forward strategy" is outlined in green, the "zero strategy" in red and the "backward strategy" in blue.

timestamp	global_wind_direction_ext
1/30/2017 9:24:19 AM	ENE
1/30/2017 12:24:19 PM	SSE
1/30/2017 4:24:19 PM	S

Hot encoding

timestamp	global_wind_direction_ext_ENE	global_wind_direction_ext_SSE	global_wind_direction_ext_S
1/30/2017 9:24:19 AM	1	0	0
1/30/2017 12:24:19 PM	0	1	0
1/30/2017 4:24:19 PM	0	0	1

Appendix C: An example of "Hot encoding": we split the categorical column "global_wind_dir_ext" to three columns since it has three categories (ENE, SSE and S). The new table contains only binary values.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

A. Mihoub conducted the research. A. Mihoub and S. Zidi developed the models and analyzed results. L. Laouamer contributed on the state of art and discussions. All authors contributed on paper redaction. All authors had approved the final version.

ACKNOWLEDGMENT

The authors gratefully acknowledge Qassim University, represented by the Deanship of Scientific Research, on the materiel support for this research under the number cbe-2018-1-14-S2327.

REFERENCES

- [1] A. Gustavsson *et al.*, "Differences in resource use and costs of dementia care between European countries: Baseline data from the ICTUS study," *J Nutr Health Aging*, vol. 14, no. 8, pp. 648–654, Oct. 2010.
- [2] L. Piyathilaka and S. Kodagoda, "Human activity recognition for domestic robots," in *Proc. Field and Service Robotics: Results of the 9th International Conference*, 2015, pp. 395–408.
- [3] H. Admoni and B. Scassellati, "Data-driven model of nonverbal behavior for socially assistive human-robot interactions," in *Proc. the 16th International Conference on Multimodal Interaction*, New York, NY, USA, 2014, pp. 196–199.
- [4] A. Mihoub and G. Lefebvre, "Wearables and social signal processing for smarter public presentations," *ACM Trans. Interact. Intell. Syst.*, vol. 9, no. 2–3, pp. 1–24, Mar. 2019.
- [5] T. Gan, J. Li, Y. Wong, and M. S. Kankanhalli, "A multi-sensor framework for personal presentation analytics," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 15, no. 2, pp. 1–21, Jun. 2019.
- [6] A. Mihoub and G. Lefebvre, "Social intelligence modeling using wearable devices," in *Proc. the 22Nd International Conference on Intelligent User Interfaces*, New York, NY, USA, 2017, pp. 331–341.
- [7] D. Ravì C. Wong, B. Lo, and G.-Z. Yang, "A Deep learning approach to on-node sensor data analytics for mobile or wearable devices," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, no. 1, pp. 56–64, Jan. 2017.
- [8] B. Barshan and M. C. Yuksek, "Recognizing daily and sports activities in two open source machine learning environments using body-worn sensor units," *The Computer Journal*, vol. 57, no. 11, pp. 1649–1667, Nov. 2014.
- [9] A. Jalal and M. Mahmood, "Students' behavior mining in e-learning environment using cognitive processes with information technologies," *Educ Inf Technol*, vol. 24, no. 5, pp. 2797–2821, Sep. 2019.
- [10] A. Mihoub, G. Bailly, C. Wolf, and F. Elisei, "Graphical models for social behavior modeling in face-to face interaction," *Pattern Recognition Letters*, vol. 74, pp. 82–89, Apr. 2016.
- [11] A. Mihoub, G. Bailly, C. Wolf, and F. Elisei, "Learning multimodal behavioral models for face-to-face social interaction," *J Multimodal User Interfaces*, pp. 1–16, Jul. 2015.
- [12] Z. Zhu, H. F. Satizábal, U. Blanke, A. Perez-Uribe, and G. Tröster, "Naturalistic recognition of activities and mood using wearable electronics," *IEEE Transactions on Affective Computing*, vol. 7, no. 3, pp. 272–285, Jul. 2016.
- [13] A. Yassine, S. Singh, and A. Alamri, "Mining human activity patterns from smart home big data for health care applications," *IEEE Access*, vol. 5, pp. 13131–13141, 2017.
- [14] A. Jalal, M. A. K. Quaid, and M. A. Sidduqi, "A triaxial acceleration-based human motion detection for ambient smart home system," in *Proc. 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCASP)*, 2019, pp. 353–358.
- [15] A. Jalal, Md. Z. Uddin, J. T. Kim, and T.-S. Kim, "Daily human activity recognition using depth silhouettes and \mathcal{R} Transformation for smart home," *Toward Useful Services for Elderly and People with Disabilities*, Berlin, Heidelberg, 2011, pp. 25–32.
- [16] S. Chernbumroong, S. Cang, A. Atkins, and H. Yu, "Elderly activities recognition and classification for applications in assisted living," *Expert Systems with Applications*, vol. 40, no. 5, pp. 1662–1674, 2013.
- [17] A. Jalal, M. A. K. Quaid, and K. Kim, "A wrist worn acceleration based human motion analysis and classification for ambient smart home system," *J. Electr. Eng. Technol.*, vol. 14, no. 4, pp. 1733–1739, Jul. 2019.
- [18] A. Jalal and S. Kamal, "Real-time life logging via a depth silhouette-based human activity recognition system for smart home services," in *Proc. 2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2014, pp. 74–80.
- [19] C. Wilson, T. Hargreaves, and R. Hauxwell-Baldwin, "Smart homes and their users: A systematic analysis and key challenges," *Personal and Ubiquitous Computing*, vol. 19, no. 2, pp. 463–476, Feb. 2015.
- [20] C. Debes, A. Merentitis, S. Sukhanov, M. Niessen, N. Frangiadakis, and A. Bauer, "Monitoring activities of daily living in smart homes: Understanding human behavior," *IEEE Signal Processing Magazine*, vol. 33, no. 2, pp. 81–94, Mar. 2016.
- [21] T. Nef *et al.*, "Evaluation of three state-of-the-art classifiers for recognition of activities of daily living from smart home ambient data," *Sensors*, vol. 15, no. 5, pp. 11725–11740, May 2015.
- [22] A. Benmansour, A. Bouchachia, and M. Feham, "Multioccupant activity recognition in pervasive smart home environments," *ACM Computing Surveys*, vol. 48, no. 3, pp. 1–36, Dec. 2015.
- [23] J. Cumin, G. Lefebvre, F. Ramparany, and J. L. Crowley, "A dataset of routine daily activities in an instrumented home," in *Proc. International Conference on Ubiquitous Computing and Ambient Intelligence*, 2017, pp. 413–425.
- [24] A. Jalal, S. Kamal, and D. Kim, "A depth video-based human detection and activity recognition using multi-features and embedded hidden Markov models for health care monitoring systems," *IJIMAI*, vol. 4, pp. 54–62, 2017.
- [25] R. Chavarriaga *et al.*, "The opportunity challenge: A benchmark database for on-body sensor-based activity recognition," *Pattern Recognition Letters*, vol. 34, no. 15, pp. 2033–2042, Nov. 2013.
- [26] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1192–1209, Third 2013.
- [27] U. A. B. U. A. Bakar, H. Ghayvat, S. F. Hasanm, and S. C. Mukhopadhyay, "Activity and anomaly detection in smart home: A survey," *Next Generation Sensors and Systems*, vol. 16, pp. 191–220, 2016.
- [28] D. Singh and C. K. Mohan, "Graph formulation of video activities for abnormal activity recognition," *Pattern Recognition*, vol. 65, pp. 265–272, May 2017.
- [29] A. Jalal, S. Kamal, and C. A. Azurdia-Meza, "Depth maps-based human segmentation and action recognition using full-body plus body color cues via recognizer engine," *J. Electr. Eng. Technol.*, vol. 14, no. 1, pp. 455–461, Jan. 2019.
- [30] K. Kim, A. Jalal, and M. Mahmood, "Vision-based human activity recognition system using depth silhouettes: A smart home system for monitoring the residents," *J. Electr. Eng. Technol.*, vol. 14, no. 6, pp. 2567–2573, Nov. 2019.
- [31] S. Garc á, J. Luengo, and F. Herrera, *Data Preprocessing in Data Mining*, vol. 72, Cham: Springer International Publishing, 2015.
- [32] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, Jul. 2015.
- [33] S. Suthaharan, "Support vector machine," in *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*, S. Suthaharan, Ed. Boston, MA: Springer US, 2016, pp. 207–235.
- [34] L. Nguyen, "Tutorial on hidden Markov model," *Applied and Computational Mathematics*, vol. 6, no. 4, p. 16, Jun. 2016.
- [35] F. J. Ord óez, P. de Toledo, and A. Sanchis, "Activity recognition using hybrid generative/discriminative models on home environments using binary sensors," *Sensors (Basel)*, vol. 13, no. 5, pp. 5460–5477, Apr. 2013.
- [36] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct. 2001.
- [37] C. J. C. Burges, "A Tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, Jun. 1998.
- [38] A. Jalal, M. A. K. Quaid, and A. S. Hasan, "Wearable sensor-based human behavior understanding and recognition in daily life for smart environments," in *Proc. 2018 International Conference on Frontiers of Information Technology (FIT)*, 2018, pp. 105–110.
- [39] W. Wang, P. Jones, and D. Partridge, "Assessing the impact of input features in a feedforward neural network," *NCA*, vol. 9, no. 2, pp. 101–112, Jul. 2000.
- [40] R. Polikar, "Ensemble learning," *Scholarpedia*, vol. 4, no. 1, p. 2776, 2009.
- [41] S. B. Kotsiantis, "Decision trees: A recent overview," *Artif Intell Rev*, vol. 39, no. 4, pp. 261–283, Jun. 2011.
- [42] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*, The MIT Press, 2009.
- [43] Y. Song, Z. Wen, C.-Y. Lin, and R. Davis, "One-class Conditional Random Fields for Sequential Anomaly Detection," in *Proc. the Twenty-Third International Joint Conference on Artificial Intelligence*, Beijing, China, 2013, pp. 1685–1691.
- [44] A. K. Jain, J. Mao, and K. M. Mohiuddin, "Artificial neural networks: A tutorial," *Computer*, vol. 29, no. 3, pp. 31–44, Mar. 1996.
- [45] H. D. Mehr, H. Polat, and A. Cetin, "Resident activity recognition in smart homes by using artificial neural networks," in *Proc. 2016 4th International Istanbul Smart Grid Congress and Fair (ICSG)*, 2016, pp. 1–5.

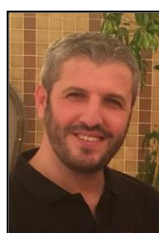
- [46] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," in *Proc. the IEEE*, 1989, pp. 257–286.
- [47] O. Brdiczka, M. Langet, J. Maisonnasse, and J. L. Crowley, "Detecting human behavior models from multimodal observation in a smart home," *IEEE Transactions on Automation Science and Engineering*, vol. 6, no. 4, pp. 588–597, Oct. 2009.
- [48] X. Li, "HMM based action recognition using oriented histograms of optical flow field," *Electronics Letters*, vol. 43, no. 10, pp. 560–561, May 2007.
- [49] A. Jalal, S. Kamal, and D. Kim, "A depth video sensor-based life-logging human activity recognition system for elderly care in smart indoor environments," *Sensors*, vol. 14, no. 7, pp. 11735–11759, Jul. 2014.
- [50] A. Jalal, Y.-H. Kim, Y.-J. Kim, S. Kamal, and D. Kim, "Robust human activity recognition from depth video using spatiotemporal multi-fused features," *Pattern Recognition*, vol. 61, pp. 295–308, Jan. 2017.
- [51] J. Cumin, G. Lefebvre, F. Ramparany, and J. L. Crowley, "Inferring availability for communication in smart homes using context," in *Proc. IEEE International Conference on Pervasive Computing and Communications*, Athens, Greece, 2018, pp. 1–6.
- [52] R. G. Miller Jr., *Beyond ANOVA: Basics of Applied Statistics*, Chapman and Hall/CRC, 1997.
- [53] I. T. Jolliffe, *Principal Component Analysis*, Springer Science & Business Media, 2006.
- [54] D. Roggen *et al.*, "Collecting complex activity datasets in highly rich networked sensor environments," in *Proc. 2010 Seventh International Conference on Networked Sensing Systems (INSS)*, 2010, pp. 233–240.
- [55] T. L. M. Van Kasteren, G. Englebienne, and B. J. Kröse, "Transferring knowledge of activity recognition across sensor networks," in *Proc. International Conference on Pervasive Computing*, 2010, pp. 283–300.
- [56] H. Alemdar, H. Ertan, O. D. Incel, and C. Ersoy, "ARAS human activity datasets in multiple homes with multiple residents," in *Proc. the 7th International Conference on Pervasive Computing Technologies for Healthcare*, ICST, Brussels, Belgium, Belgium, 2013, pp. 232–235.
- [57] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, Aug. 2004.
- [58] V. N. Vapnik, *The Nature of Statistical Learning Theory*, New York, NY, USA: Springer-Verlag New York, Inc., 1995.
- [59] J. R. Quinlan, *C4.5: Programs for Machine Learning*, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.
- [60] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and Regression Trees*, 1st ed. New York, N.Y.: Chapman and Hall/CRC, 1984.
- [61] G. V. Kass, "An exploratory technique for investigating large quantities of categorical data," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 29, no. 2, pp. 119–127, Jan. 1980.
- [62] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, p. 2825–2830, Oct. 2011.
- [63] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, Apr. 2011.
- [64] D. M. Powers, *Evaluation: From Precision, Recall and F-measure to ROC, Informedness, Markedness and Correlation*, Dec. 2011.
- [65] M. Fernandez-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?" *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3133–3181, 2014.
- [66] J. Maroco, D. Silva, A. Rodrigues, M. Guerreiro, I. Santana, and A. de Mendonça, "Data mining methods in the prediction of Dementia: A real-data comparison of the accuracy, sensitivity and specificity of linear discriminant analysis, logistic regression, neural networks, support vector machines, classification trees and random forests," *BMC Research Notes*, vol. 4, no. 1, p. 299, Aug. 2011.



Alaeddine Mihoub received his PhD degree in computer science from Grenoble-Alpes University, France in 2015). Before, he received his MSc degree in computer science from Paris Descartes University, France in 2012 and his engineering diploma in statistics and data analysis from ESSAIT, Tunisia in 2011. Actually, Dr. Alaeddine Mihoub is an assistant professor of computer science in the College of Business and Economics (CBE, MIS Department) at the Qassim University (Saudi Arabia). Previously, he was a research and data scientist at Maskott, France and earlier, a postdoctoral researcher at Orange Labs, Meylan, France. He is the co-author of many innovative patents as well as several publications in well-ranked international journals and conferences. His research interests include social signal processing, statistical and probabilistic models for interactive systems and more largely artificial intelligence, machine learning and data science.



Salah Zidi received his electrical engineering degree in 2003 from the University of Sfax and after one year he received his master's degree in computer engineering and embedded systems. He achieved his PhD in the University of Lille in France in 2007. His PhD research work was concentrated on the optimization and using machine learning for decision support systems, where he published multiple papers in prestigious international journals and conferences. Between 2007 and 2014, he worked as an instructor and researcher in the University of Science and Technologies of Lille, and as a consultant in innovation in some TIC enterprises. Currently, Dr. Zidi is an assistant professor in MIS department in the College of Business and Economics in Qassim University since 2014 and associate professor in the University of Gabes in Tunisia starting from 2018. He received his HDR degree in 2017 from the University of Lille I. It had dealt with the application of machine learning techniques for complex systems. His research interest covers optimization as a general topic of his PhD thesis. After his PhD, he is interested in artificial intelligence, machine learning, feature extraction and data analysis for automation systems and complex systems. Furthermore, he co-supervised many students in PhD and master between 2007 and 2018.



Lamri Laouamer is an associate professor at the Department of Management Information Systems, College of Business and Economics, Qassim University, KSA. He is also an associate researcher in Laboratoire des Sciences et Techniques de l'Information, de la Communication et de la Connaissance (Lab-STICC), University of Bretagne Occidentale, Brest, France. He received his Ph.D. in 2012 in computer science, field of information security from the University of Bretagne Occidentale, France. He got the M.Sc. in 2006 in computer science and applied mathematics from the University of Quebec at Trois Rivieres in Canada. He also got the B.Sc. in 1999 in computer science from the University of Setif, Algeria. His research interests include multimedia watermarking, cryptology, information security. Dr. Lamri Laouamer served as an associate editor of the journal of Telecommunication system, Springer and an associate editor of the Journal of Innovation in Digital Ecosystems by Elsevier.