Appliance and State Recognition using Hidden Markov Models

Antonio Ridi[*†], Christophe Gisler[*†], Jean Hennebert[*†]
[*]University of Applied Sciences Western Switzerland,
College of Engineering and Architecture of Fribourg, IcoSys Institute
{antonio.ridi, christophe.gisler, jean.hennebert}@hefr.ch
[†]University of Fribourg, Department of Informatics, Fribourg, Switzerland
{antonio.ridi, christophe.gisler, jean.hennebert}@unifr.ch

***Abstract*—We asset about the analysis of electrical appliance consumption signatures for the identification task. We apply Hidden Markov Models to appliance signatures for the identification of their category and of the most probable sequence of states. The electrical signatures are measured at low frequency (10^{-1} Hz) and are sourced from a specific database. We follow two predefined protocols for providing comparable results. Recovering information on the actual appliance state permits to potentially adopt energy saving measures, as switching off stand-by appliances or, generally speaking, changing their state. Moreover, in most of the cases appliance states are related to user activities: the user interaction usually involves a transition of the appliance state. Information about the state transition could be useful in Smart Home / Building Systems to reduce energy consumption and increase human comfort. We report the results of the classification tasks in terms of confusion matrices and accuracy rates. Finally, we present our application for a real-time data visualization and the recognition of the appliance category with its actual state.**

Keywords*—*Intrusive Load Monitoring (ILM); Appliance Identification; Appliance State Recognition

I. Introduction

The appliance energy consumption represents a considerable source of consumption for the industrial and residential sector. The quantity of energy used by appliances is difficult to be exactly estimated since energy meters are usually placed at the panel level. They measure the energy consumption of the whole location, including Heating, Ventilation and Air Conditioning system and lighting. As a consequence, householders might be in trouble in explaining their electricity bill. Energetic labels give rough information about appliance electrical consumption without providing the exact amount of wasted energy. Moreover, the energy consumption of several appliances depends on user utilization and can widely vary from user to user.

Information about appliances can be retrieved through two approaches: Intrusive Load Monitoring (ILM) and Non-Intrusive Load Monitoring (NILM). In the first case numerous sensors are typically distributed at plug level or directly on appliances. Several signatures are contemporaneously acquired enabling a high degree of system granularity. In the second case only one smart meter is used, typically at panel level. All electrical signatures are aggregated in one signal and *disaggregation* techniques permit to recover part of the original signatures.

The appliance waste of energy is mainly due to stand-by and inappropriate use of appliances. In the latter case we include the inattention to electricity use, i.e. behaviors such as forgetting to turn off energy-consuming appliances. The knowledge of the state in which the appliances are (their *actual state*) could be useful for adopting potentially energy saving measures, as switching off stand-by appliances or changing their state to another less energy-consuming.

Information about users activities can be usually retrieved from the appliance state transition: the user interaction usually involves a transition of the appliance state. The appliance can be separated into three categories [13]:

- *Usage dependent* appliances (e.g. Microwave). In this case the appliance transitions provide straight information about the user activity.
- *Fixed operation* appliances (e.g. Coffee Machine). These appliances usually require the user to start the sequence of fixed operations. For instance, the Coffee Machine needs to be switched on and a button needs to be pressed for the coffee supply.
- *Thermostatically controlled* appliances (e.g. Fridge). The users can interact with them as well. For instance, if we want to interact with a Fridge we need to open the door: the small but detectable lamp inside the Fridge will switch on automatically.

State-based modelling techniques, like Gaussian Mixture Models and Hidden Markov Models, are particularly suitable for this task, given the state-base nature of electrical signatures. Up to now, state based modelling has been applied to aggregated signals coming from NILM systems. In this paper we asset about the utilization of state-based modelling on appliance signatures, equivalent to ILM system. In particular, we focus on the state recognition and we evaluate the capability of our machine learning algorithm to recover information about the actual state of the tested appliances.

In Section II we provide details about related works by using state-based modelling and appliance state recognition in the domain. In Section III we present the database used for the analysis, the pre-processing operation and the feature selection. In Section IV we resume the theory of Hidden Markov Models (HMMs) which is the state based modelling algorithm used in this paper. In Section V we present and discuss the results. In Section VI we present our application for the real-time appliance identification and state recognition. Finally, we conclude the paper in Section VII.

II. Related works

State-based modelling is a well-known approach in the context of device recognition. As previously said, this approach is widely used in NILM systems which is not the case for ILM systems. This is probably due to the fact that for ILM the device recognition is easier and it does not require the algorithm complexity necessary for NILM.

Durand et al. [2] present the analysis of the electrical consumption based on Hidden Markov Chains. In particular, they estimate the influence of factors such as month, day, hours, type of contract and maximal power on the log-consumption. In their approach the hidden states are domestic activities. The electrical consumption is estimated from the restored states and a contingency table relating the hidden states with the electrical appliance consumption. In their work they use 100 houses monitored for an entire year.

Zia et al. [14] propose an approach for recognition of single appliances from the whole home energy consumption. They build an HMM for each individual appliance and they tune the number of states and their topology depending on appliance characteristics. All the models are merged together to describe the entire energy consumption. They recover the sequence of states using the Viterbi algorithm and derive the operational mode of single appliances. They test Fridge, Dishwasher, Microwave, Coffee machine, Computer and Printer categories.

Other works propose more complex variants of HMMs. Factorial HMM (FHMM) [1][6][7][10][15] seems quite popular in NILM context. FHMM allows a single observation to be related with the hidden variables of multiple independent Markov Chains. Hidden Semi-Markov Model (HSMM) are also used [6]. In HSMM the probability of changing the hidden state depends on the amount of time elapsed starting from the entry into the current state.

Information about the actual state of appliances and their transitions could potentially be retrieved by using state based-modelling. However, other machine learning techniques have been successfully used for this task. In NILM systems the state recognition based on state transition graph [4] or based on integer programming [5] has been examined. In NILM systems some authors analyzed the steady-state, occurring when the values of consecutive samples are lower than a certain threshold. The difference between two steady-states defines a steady-state signature. This approach allows to identify appliances turned on / off, as reported in [3].

In ILM systems Saitoh et al. [11] present a current based sensor able to perform appliance and state recognition. They collect a total of 28 appliance categories commonly present in home or office environment. Three appliances for each category are present (for a total of 84 appliances). They define for every instance a number of states variable between one to four (except the off state), yielding to a total of 162 states. In their analysis they use three typical features of the current: the peak value (I_{peak}), the average value (I_{avg}) and the root mean square value (I_{rms}). As machine learning techniques they use k-Nearest Neighbor and Support Vector Machine. The best results they obtain are 90.3% for the category recognition, 86.4% for the recognition of specific appliances and 84.3% for the state recognition. In the last case they use the leave-one-out technique because of the few number of samples per states.

III. Data, Pre-processing and Feature Selection

A. Data

We used the ACS-F2 database as source of data [9]. The database contains 225 appliances spread into 15 categories: Mobile phone (via charger), Coffee machine, Computer station (including Monitor), Fridge and Freezer, Hi-Fi system (CD players), Lamp (CFL), Laptop (via chargers), Microwave oven, Printer, Television (LCD or LED), Fan, Shaver, Monitor, Lamp (incandescent) and Kettle. Every appliance is recorded into two separated instances of one hour (called *sessions*). The electrical signatures are acquired at the low sampling frequency of 10^{-1} Hz. This database has two main peculiarities:

1) All the appliances belonging to the same category differ in their brands and / or models. If this would not be the case, a model representing a specific category could be biased by the data repetition. Moreover, the generated models are reasonably generic to be used on instances coming from appliances not present in the database.
2) Two evaluation protocols have been presented with the database. This is useful for comparing the result of the classification task. In the first protocol, called *Intersession*, the first hours of recording are used for training the models, while the other instances are tested. In the second protocol, called *Unseen appliance*, a 15-cross fold validation is performed and in every fold both sessions of one appliance per category are present.

More details on protocol procedures can be found in [9].

B. Pre-processing

In the pre-processing phase we computed some basic operations on data. The available features are real power (W), reactive power (var), network frequency (Hz), RMS current (A), RMS voltage (V) and phase of voltage relative to current (ϕ).

ϕ is described by an angle between 0 and 360 degrees. Because of the cyclic nature of the phase, some appliances can have consecutive samples in time that shift from 0 to 360 degrees and vice versa. For avoiding this jump in the phase value we computed the sinus of ϕ in radiant. The problem of bijection is excluded by the positivity of the cosine of the phase in radiant of all the samples. For sake of simplicity hereunder we will use the term "phase angle" for indicating the sinus of ϕ in radiant.

We repeated the computation of dynamic coefficients, being their usefulness demonstrated in similar tasks [8]. The *velocity* and *acceleration* parameters, also called respectively *delta* and *delta-delta* coefficients, were computed. The *velocity* coefficients are calculated as follows:

$$\Delta o_n = \sum_{w=-W}^{W} w \times o_{n-w} \tag{1}$$

where W determines the length of the analysis window, that we set equal to 50 seconds. We computed the *acceleration* coefficients as:

$$\Delta\Delta o_n = \Delta o_{n+1} - \Delta o_{n-1} \tag{2}$$

The *velocity* and *acceleration* coefficients were added to the feature space, reaching a total of 18 features.
We used the z-normalization for normalizing the features:

$$x_{ki} = \frac{o_{ki} - \mu_k}{\sigma_k} \tag{3}$$

where μ_k and σ_k are respectively the mean and variance computed for every k feature using all the signatures in the training set. After this operation the mean is equal to zero and the variance is equal to one.

C. Feature Selection

In spite of the known correlation existing among the measurements provided by the database (as per the laws of electricity), we wanted to evaluate, from a machine learning perspective, the information brought by each measurement and coefficient in the context of the identification task.
Generally speaking, the features used for classification task have to be *relevant* and *not redundant*. A feature is considered *relevant* if it is highly correlated to the classes and *redundant* if it is highly correlated to other features [12]. This information can be provided through the analysis of the correlation between variables. Two approaches are mainly used: the *linear correlation* and the correlation from the *information theory*. When the variables are not linearly correlated the first approach is scarcely effective. In this work we used the second approach: we computed the *entropy* of the features [12].
The entropy (H) of a feature X can be computed as follows:

$$H(X) = -\sum_i P(x_i) log_2(P(x_i)) \tag{4}$$

where $P(x_i)$ are the prior probabilities of X. The entropy of the feature X when observing another feature Y is defined as:

$$H(X|Y) = -\sum_j P(y_j) \sum_i P(x_i|(y_j)) log_2(P(x_i|(y_j))) \tag{5}$$

where $P(x_i|(y_j))$ are the posterior probabilities of X given the values of Y. Using the previous two equations, we can compute the information gain (IG) as:

$$IG(X|Y) = H(X) - H(X|Y) \tag{6}$$

The *symmetrical uncertainty* (SU) was chosen for compensating the IG bias and for normalizing the values:

$$SU(X,Y) = 2\left[\frac{IG(X|Y)}{H(X) + H(Y)}\right] \tag{7}$$

In Figure 1 the SU is depicted by using the gray scale. For finding the "not relevant" features we checked the attributes for which SU is lower than a prefixed threshold. In our case, the network frequency and its dynamic coefficients (lines 2, 8, 14 in Figure 1) have been categorized as not relevant. As expected, the small deviations on the network frequency are substantially independent by the appliance category.
As second step we determined the redundant features. We used the information available from the theory of electricity instead of the SU matrix for sake of simplicity. The phase angle, reactive power and active power supply redundant information: the third one can be computed from the other two. The phase angle has the lowest SU value among the three, therefore

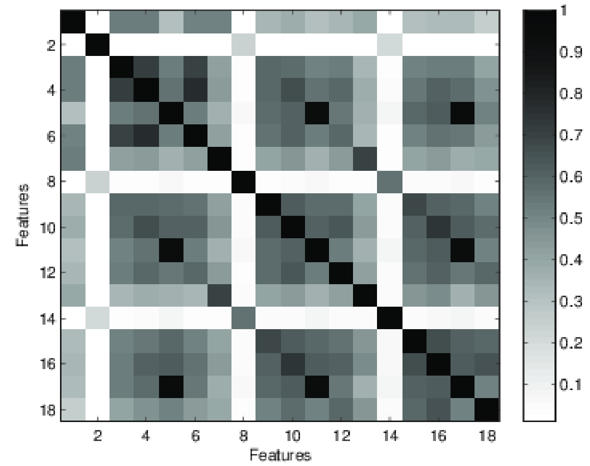

Fig. 1. Symmetrical uncertainty in gray scale. From the position 1 to 6 there are the original features (in the order: phase angle, network frequency, reactive power, real power, RMS voltage, RMS current), from 7 to 12 the velocity coefficients and from 13 to 18 the acceleration coefficients.

this feature and its dynamic coefficients were selected as redundants. Following the same procedure we analyzed the RMS voltage, the RMS current and the active power. We selected the RMS voltage and its dynamic coefficients as redundants because it had the lowest SU value among the three features.
As next step we evaluated the impact of the feature space reduction through the evaluation of the classification results. We considered four feature sets:

1) Set I: all features were included.
2) Set II: we removed the network frequency and its dynamic coefficients.
3) Set III: we removed the network frequency, the RMS voltage and their dynamic coefficients.
4) Set IV: we removed the network frequency, the RMS voltage, the phase angle and their dynamic coefficients.

This procedure has been repeated for the two protocols by using as machine learning algorithm the Gaussian Mixture Model (GMMs). In Figure 2 we reported the variation of the accuracy rate with increasing number of Gaussians. The continuous lines are used for the *Intersession* protocol, while the dashed lines for the *Unseen appliance* protocol. The Set I is in blue, Set II in red, Set III in green and Set IV in black. In our implementation the initialization of the points is randomly attributed and the Expectation Maximization algorithm could converge to different final parameters. For this reason, for the *Intersession* protocol we computed the models 10 times, randomly changing the initial conditions, and averaging the results. We skipped this step for *Unseen appliance* protocol given that a 15-cross fold validation is performed by default. Observing the results we came to the following conclusions:

- *Intersession* protocol. We observed a stable improvement of the performances when removing the network frequency (more than 4%). In other cases non significative differences have been ascertained. The

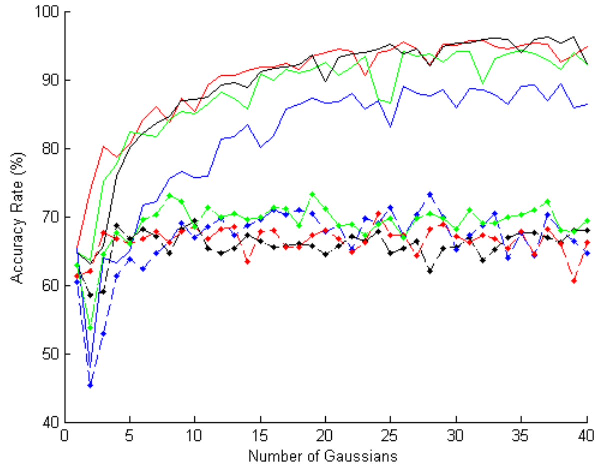

Fig. 2. Accuracy rate trends when varying the number of Gaussians. The continuous lines are for the *Intersession* protocol, while the dashed lines for *Unseen appliance* protocol. The Set I is in blue, Set II in red, Set III in green and Set IV in black.

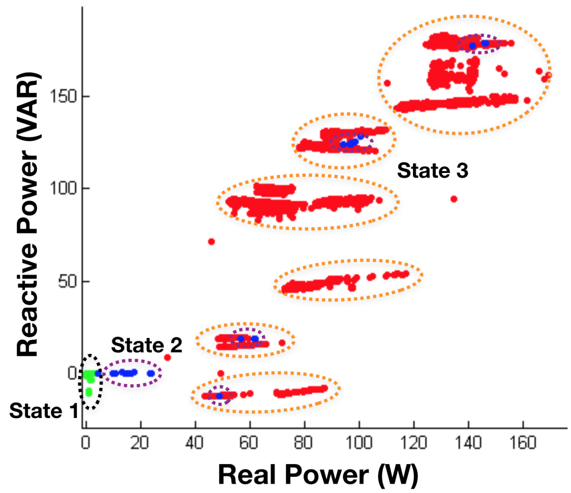

Fig. 3. All the samples contained in the Fridge category of the ACS-F2 database. The non-compression phase is in green, the opening of the door in blue and the compression phase in red.

simplest feature set (Set IV) is taken into account for further analysis.

- *Unseen appliance* protocol. We observed a complex situation. The accuracy rate variations can hardly be distinguished and compared because of the large variability intrinsic in this protocol. We decided to maintain all the features (Set I) given that no meaningful conclusions could be established.

IV. Classification Method

Given the natural state-based nature of the electric appliance signatures, we are interested in the use of state-based modelling for the appliance identification. We used Hidden Markov Models (HMMs) as baseline state-based machine learning algorithm widely used in the scientific community.
During the learning phase we need to determine the structure of the models and learning its parameters. The structure of a Markov model is described by the set of states $S = S_1, S_2, ..., S_N$, where N is number of states, and the topology of their connections. The structure can be created by using information on the nature of the model or can be computed with complex strategies, mainly in the information theory field. In this work we chose models with two states for almost all the categories (on / off states). For the Fridge category we used three states (compression / non-compression phase / door opening). The topologies usually belong to one of the following categories:

1) Ergodic. Every state of the model can be reached in a single step from every other state.
2) Forward. At each step we can remain in the same state or reach the one with the incremental index. This topology is widely used in speech recognition.

Different topologies could be created depending on specific application. In this work we used the ergodic topology.
Once the structure of the model is defined, it is possible to learn its parameters. Transition between states (*transition probabilities*) and the conditional distributions of the observed variables for each state (*emission probabilities*) are two typical learnable parameters. A model λ can be completely defined as $\lambda(A, B, \pi)$, where:

- A is the set of state transition probabilities. Often it is arranged as a $N \times N$ matrix (called *transition matrix*):
$$A = a_{ij} = P[q_{t+1} = S_j | q_t = S_i] \quad (8)$$
where $1 \leq i, j \leq N$ and q_t is the state at time t.
- B is the probability distribution in each of the states.
$$B = b_j(O_t) = P[O_t | q_t = S_i] \quad (9)$$
where $1 \leq j \leq N$ and $b_j(O_t)$ is a function of the observations.
- π is the initial state probabilities of being in state q at time $t = 0$:
$$\pi = \pi_i = P[q_1 = S_i] \quad (10)$$
where $1 \leq i \leq N$.

When facing the classification task, HMMs rely on learning one model per class as other generative algorithms. In our implementation each state of models corresponds to a real state of the appliance. For instance, in Figure 3 we reported all the samples contained in the Fridge category of the ACS-F2 database. Different colors have been used for representing different states: the non-compression phase is in green, the opening of the door in blue and the compression phase in red. Depending on the data distribution, the necessary number of Gaussians per state could vary.
Given a sequence of observations $O = O_1, O_2, ..., O_T$ and a model $\lambda(A, B, \pi)$ described by the set of states $S = S_1, S_2, ..., S_N$ we can decode the observation sequence and recover the most probable sequence of states. The path of states $q_1, q_2, ..., q_T$ (called *alignment*) and its probability of outcoming (called *likelihood*) are recovered by using the Viterbi algorithm. During the test phase we can use these information for the classification task. The instances in the test

set are checked against the models, computing an *alignment* and *likelihood* score for every model.

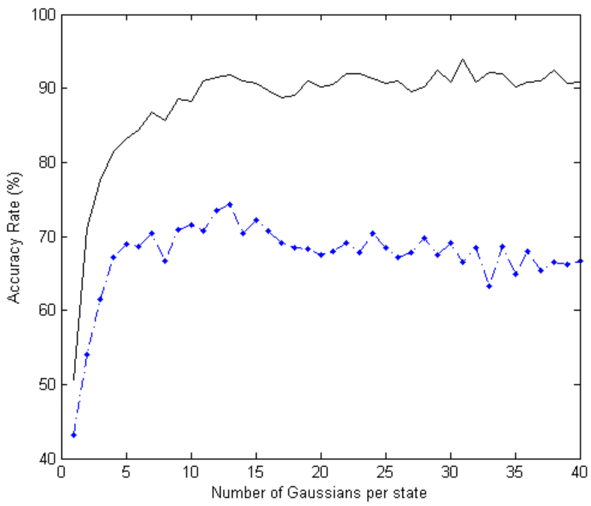

Fig. 4. Accuracy rate trends when varying the number of Gaussians per state. The continuous line is for the *intersession* protocol (in black the Set IV), while the dashed lines for *unseen appliance* protocol (in blue the Set I).

Using the *likelihood* score and the *a priori* probability we can compute the *a posteriori* probability, according to the Bayes rule. In the ACS-F2 database the appliances are uniformly distributed, therefore we can make the hypothesis of equal priors. The *alignment* can be compared with the original labels for obtaining the accuracy rate for state recognition. The *alignment* depends on the most probable model chosen by the appliance recognition task: as a consequence, we could compare an *alignment* of a wrong model with the original labels. For this reason we computed two accuracy rates for the state recognition:

- The real accuracy rate (AR_R), when considering the *alignment* of the chosen models.
- The oracle accuracy rate, (AR_{Or}), when considering the *alignment* of the correct models.

Given that AR_R is subject to misclassification errors of the appliance recognition, its value is expected to be lower than AR_{Or}.

V. Results and Discussion

We performed the tests by using the conclusions coming from the feature selection. For the *Intersession* protocol we used the smallest feature set (Set IV), while for the *Unseen appliance* protocol we maintained all the features (Set I). We used HMMs as machine learning algorithm for the reasons explained in the previous Sections. For both protocols we computed the accuracy rate as a function of the number of Gaussians per states (I). In fact, in our application we have the same number of Gaussians among states.

For the *Intersession* protocol we repeated the experiment 10 times and we averaged the results, as done with GMMs during the feature selection. In Figure 4 we presented the mean value of the accuracy rates (continuous blue line). The performances increased significantly until 15 mixtures where the improvements start saturating. In our best case we obtained an accuracy rate of 93.91% when $I = 31$. In Table I we reported the confusion matrix. No improvements were shown compared to GMMs algorithm. However, in our best single case (before the average operation) we attained an accuracy rate of 97.78%. Single categories had very good accuracy rates with the exception of the Fridge that was misclassified with Oven.

For the *Unseen appliance* protocol we also calculated and reported the accuracy rates with varying number of Gaussians per state. In Figure 4 we reported the trend of the accuracy rate using the Set I (dashed black line). As best case we attained an accuracy rate of 74.2% when $I = 13$. As expected, fewer mixtures were necessary because they do not specialize on appliance brands and models. In Table II we reported the confusion matrix. We noticed an improvement of almost 1% compared to the best case when using GMMs. Single categories had good accuracy rates with the exception of Shaver that was misclassified with other low consumption categories. We also noticed that Kettle attained the 100% of accuracy rate.

TABLE I. Confusion matrix using the intersession protocol when $I = 31$.

	Hi-Fi	Television	Mobile P.	Coffee M.	Computer	Fridge	Lamp Inc.	Laptop	Oven	Printer	Fan	Kettle	Lamp CFL	Monitor	Shaver
Hi-Fi	.95	0	0	0	0	0	0	0	0	0	0	0	0	0	.05
Television	0	.99	0	0	0	0	0	0	0	0	0	0	0	.01	0
Mobile P.	0	0	.9	0	0	0	0	0	0	0	0	.07	0	0	.03
Coffee M.	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Computer	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Fridge	0	.01	0	.01	0	.6	0	.03	.31	.05	.01	0	0	0	0
Lamp Inc.	0	0	0	0	0	0	.99	0	0	0	.01	0	0	0	0
Laptop	0	0	0	0	0	0	0	.95	0	.01	.04	0	0	0	0
Oven	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Printer	0	0	0	0	0	0	0	0	0	.93	0	0	0	0	.07
Fan	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Kettle	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Lamp CFL	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Monitor	0	0	0	0	0	0	0	.02	0	.07	0	0	0	.91	0
Shaver	0	0	.07	0	0	0	0	0	0	0	0	.07	0	0	.86

TABLE II. Confusion matrix using the Unseen appliance protocol when $I = 13$

	Hi-Fi	Television	Mobile P.	Coffee M.	Computer	Fridge	Lamp Inc.	Laptop	Oven	Printer	Fan	Kettle	Lamp CFL	Monitor	Shaver
Hi-Fi	.53	0	.07	0	0	.07	0	.03	0	.03	.17	0	.03	0	.07
Television	0	.57	0	0	.17	.03	0	.13	0	0	.03	0	0	.07	0
Mobile P.	0	0	.8	0	0	0	.03	0	0	0	0	0	.03	0	.13
Coffee M.	0	0	0	.83	0	0	0	0	.1	.03	0	.03	0	0	0
Computer	0	.2	0	0	.77	0	0	.03	0	0	0	0	0	0	0
Fridge	0	.1	0	.03	.03	.77	0	0	0	0	.07	0	0	0	0
Lamp Inc.	0	0	0	.03	0	.03	.87	0	0	0	.07	0	0	0	0
Laptop	0	0	0	0	0	0	0	.73	0	.03	.07	0	.07	.1	0
Oven	0	0	0	.03	0	.07	0	0	.9	0	0	0	0	0	0
Printer	0	0	0	.03	.03	.07	0	.03	.03	.5	.07	0	0	.07	.17
Fan	0	.07	0	0	0	0	0	0	0	0	.93	0	0	0	0
Kettle	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Lamp CFL	0	0	0	0	0	0	0	0	0	0	.17	0	.77	0	.07
Monitor	0	.03	0	0	0	0	0	.13	0	0	0	0	0	.83	0
Shaver	.13	0	.23	0	0	0	.03	0	0	0	.07	.07	.13	0	.33

TABLE III. THE AVERAGES OF AR_R AND AR_{Or} FOR THE APPLIANCE STATE RECOGNITION USING BOTH PROTOCOLS.

	Intersession		Unseen Appliance	
	AR_R (%)	AR_{Or} (%)	AR_R (%)	AR_{Or} (%)
Hi-Fi	81.9	76.8	69.7	70.2
Television	99.4	99.4	95.9	97.3
Mobile P.	66.6	64.8	67.5	70
Coffee M.	99.5	99.5	95.4	96.9
Computer	99.9	99.9	99.7	99.8
Fridge	84.8	84.7	87.8	91.2
Lamp Inc.	99.8	99.8	98.6	99
Laptop	97.3	97.9	96.4	96.5
Oven	99.8	99.8	99	98.7
Printer	91.1	91.2	60.3	71
Fan	99.9	99.9	99.7	97.7
Kettle	99.9	99.9	99.9	99
Lamp CFL	99.9	99.9	99.9	97.4
Monitor	93.5	99.9	99.5	99.8
Shaver	89.2	90.1	86.9	84.4
average	93.5	93.6	90.4	91.4

In Table III we reported the averages of AR_R and AR_{Or} per classes when performing the state recognition for both protocols. As expected, the average AR_{Or} is greater than the average AR_R, even if the difference is small for the *Intersession* protocol. The states of some categories, like Mobile Phone and Hi-Fi, seem harder to be correctly identified, providing poor results in all the cases. Others categories, like Computer, Lamp, Fan and Kettle, provided in all cases very good performances for state recognition.

VI. REAL-TIME APPLICATION FOR THE APPLIANCE AND STATE RECOGNITION

In the previous Section we have shown the system performances when identifying the appliance category and the states of appliances. The work has been realized by using the data of the ACS-F2 database. In this Section we want to show the feasibility of a system working in real-time by using the parameters previously learnt by the HMMs algorithm.
We realized a system able to provide in real-time the same features of those contained in the ACS-F2 database. The data is provided with the same sampling frequency: every 10 seconds we receive new data. The samples are pre-processed as previously described. The samples are progressively stored for being used for the recognition task. We use the emission and transition matrices of all the models for the best case ($I = 31$) of the *Intersession* protocol. In Figure 5 the interface of our application is presented. Four parts have been implemented:

1) Device connection. The application establishes the connection and gives a positive or negative feedback to the user depending on the connection result.
2) Data communication. When the device is connected, the communication can be started, stopped or reset by the user. The feedback on the total number of samples received is provided to the user.
3) Data visualization. The original features are shown in real-time: a popup menu allows to select the desired features. After 50 seconds from the start of the communication (the size of the analysis window) the dynamic coefficients can be computed and visualised also in real-time.
4) Appliance and state identification. After the computation of the dynamic coefficients, this last layer can be activated. We used the Viterbi algorithm to determine the *alignment* and *likelihood* of all stored samples. Two classifications are effectuated in real-time, the identification of the appliance category and the actual state of the estimated appliance. The actual state corresponds to the last state of the *alignment*. The results of the classifications are shown in the bottom part of the interface.

Fig. 5. Interface of the real-time application for the appliance and state recognition.

This real-time task provides the first classification results after less then one minute from the beginning of the data communication. For obtaining similar accuracy rates of those reported in the previous Section we should use the same time duration of the instances, namely one hour. Using a shorter time duration the results are expected to drop. A longer time duration leads to better results. Preliminary results indicate an accuracy rate of about 80% after ten minutes of recording.

VII. CONCLUSION AND FUTURE WORKS

In this paper we reported the appliance identification task and the recognition of appliance states. We applied HMMs to electrical signatures contained into the ACS-F2 database [9]. We followed two pre-defined protocols: the *Intersession* and the *Unseen appliance* protocol. The first protocol tests instances coming from appliances already seen by classifiers, which is not the case of the second protocol. The electrical signatures are acquired using a low sampling frequency (10^{-1} Hz). We performed some pre-processing operation, like computation of the dynamic coefficients and z-normalisation. We performed also feature ranking and selection in order to remove the features having not-relevant or redundant information. For the first protocol we were able to reduce the feature space by half, while for the second protocol the feature space was maintained intact.
We applied HMMs algorithm to appliance electrical signatures

following the protocols guidelines. In our tests we incremented progressively the number of Gaussians (I) contained in every state and we reported the trend of the accuracy rate when incrementing I. For the first protocol we obtained the best results of 93.91% having $I = 31$. For the second protocol we obtained the best results of 74.2% when $I = 13$. The results are similar to those obtained when using GMMs.
The main advantage of this approach consists in the computation of the state sequence. Using the Viterbi alignment we recovered the most probable sequence of states (called *alignment*) that we compared with the data labelling. In the standard case the *alignment* is generated by the model recognised by the classifier and therefore it could be wrongly selected. Otherwise we can inject *a priori* the knowledge of the correct model. Depending on the case, we obtained two different results in terms of accuracy rate, respectively called AR_R and AR_{Or}. For the first protocol we obtained $AR_R = 93.5\%$ and $AR_{Or} = 93.6\%$ when $I = 31$. For the second protocol we obtained $AR_R = 90.4\%$ and $AR_{Or} = 91.4\%$ when $I = 13$. The appliance state recognition provided good accuracy rates even for the *unseen appliance* protocol that appears more challenging in the recognition of the appliance category.
We also realized a real-time application for the appliance and state recognition. This application has two main functions: data visualization and data classification. The data upcoming after the establishment of the connection can be visualized real-time after less than one minute, given the possibility to the user to select the feature and the dynamic components of his interest. The appliance and state recognition can also be performed providing the two classification results in real-time. We used the emission and transition matrices of all the models for the best case ($I = 31$) of the *Intersession* protocol. After less than one minute the classification results were available but given the small amount of time were not trustworthy. In our future works we plan to evaluate the system performances in classification task by using small data durations.
Information about the user activities can be extracted from the state transitions: we should be able to know when the user interacted with the appliance. We plan to use finer models with more states per category and eventually inject this information in systems used for the activity recognition in Smart Home / Buildings. Sensible improvement could be provided by merging the data used in this paper with other data coming from different datasets.

VIII. Acknowledgments

This work is supported by the grant Smart Living GreenMod from the Hasler Foundation in Switzerland and by the HES-SO.

References

[1] S. N. Akshay Uttama Nambi, T. G. Papaioannou, D. Chakraborty, and K. Aberer. Sustainable energy consumption monitoring in residential settings. *2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 1–6, Apr. 2013.

[2] J. Durand, L. Bozzi, G. Celeux, and C. Derquenne. Analyse de courbes de consommation electrique par chaines de Markov cachees. 2003.

[3] M. Figueiredo, A. De Almeida, and B. Ribeiro. Home electrical signal disaggregation for non-intrusive load monitoring (NILM) systems. *Neurocomputing*, 96:66–73, 2012.

[4] G. Hart. Nonintrusive Appliance Load Monitoring. *Proceedings of the IEEE*, 80(12):1870–1891, 1992.

[5] S. Inagaki, T. Egami, T. Suzuki, H. Nakamura, and K. Ito. Nonintrusive appliance load monitoring based on integer programming. *PIEEJ Transactions on Power and Energy*, 128(11):1386–1392, 2008.

[6] H. S. Kim. *Unsupervised disaggregation of low frequency power measurement*. PhD thesis, University of Illinois at Urbana-Champaign, 2012.

[7] J. Kolter and M. Johnson. REDD : A Public Data Set for Energy Disaggregation Research. In *Proceedings of the ACM Workshop on Data Mining Applications in Sustainability (SustKDD)*, 2011.

[8] A. Ridi, C. Gisler, and J. Hennebert. Automatic identification of electrical appliances using smart plugs. In *Proceedings of the 8th Internation Workshop on Systems, Signal Processing and their Applications (WOSSPA)*, pages 301–305, 2013.

[9] A. Ridi, C. Gisler, and J. Hennebert. ACS-F2 - A New Database of Appliance Consumption Analysis. In *Proceedings of the International Conference on Soft Computing and Pattern Recognition (SocPar 2014)*, 2014.

[10] L. Robert and K. Liszewski. Methods of electrical appliances identification in systems monitoring electrical energy consumption. In *Proceedings of the 7th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS)*, pages 10–14, 2013.

[11] T. Saitoh, Y. Aota, T. Osaki, R. Konishi, and K. Sugahara. Current Sensor based Non-intrusive Appliance Recognition for Intelligent Outlet. In *Proceedings of the 23rd International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, pages 349–352, 2008.

[12] L. Yu and H. Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proceedings of 20th International Conference on Machine Learning (ICML)*, 2003.

[13] F. Zaidi Adeel Abbas and P. Palensky. Load Recognition for Automated Demand Response in Microgrids. In *Proceedings of the 36th IEEE Annual Conference on Industrial Electronics Society (IECON)*, pages 2442–2447, 2010.

[14] T. Zia, D. Bruckner, and A. Zaidi. A Hidden Markov Model Based Procedure for Identifying Household Electric Loads. In *IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society*, pages 3218–3223, 2011.

[15] A. Zoha, A. Gluhak, M. Nati, and M. A. Imran. Low-power appliance monitoring using Factorial Hidden Markov Models. *2013 IEEE Eighth International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pages 527–532, Apr. 2013.