Load Monitoring and Appliance Recognition Using an Inexpensive, Low-Frequency, Data-to-Image, Neural Network, and Network Mobility Approach for Domestic IoT Systems

Peppino Fazio, *Member, IEEE*, Miralem Mehic, *Senior Member, IEEE*, and Miroslav Voznak, *Senior Member, IEEE*

***Abstract*—With the low integration costs and quick development cycle of all-IP-based 5G+ technologies, it is not surprising that the proliferation of IP devices for residential or industrial purposes is ubiquitous. Energy scheduling/management and automated device recognition are popular research areas in the engineering community, and much time and work have been invested in producing the systems required for smart city networks. However, most proposed approaches involve expensive and invasive equipment that produces huge volumes of data (high-frequency complexity) for analysis by supervised learning algorithms. In contrast to other studies in the literature, we propose an approach based on encoding consumption data into vehicular mobility and imaging systems to apply a simple convolutional neural network to recognize certain scenarios (devices powered *on*) in real time and based on the nonintrusive load monitoring paradigm. Our idea is based on a very cheap device and can be adapted at a very low cost for any real scenario. We have also created our own data set, taken from a real domestic environment, contrary to most existing works based on synthetic data. The results of the study's simulation demonstrate the effectiveness of this innovative and low-cost approach and its scalability in function of the number of considered appliances.**

***Index Terms*—Appliance classification, convolutional neural networks, data-to-image conversion, Internet of Things (IoT) networks, machine learning (ML).**

I. Introduction

IN THE last few years, attention to energy management and emissions reduction has grown out of greater awareness and acceptance of the critical ecological issues arising from CO_2 pollution. Energy management has become an integral part of developing ecological smart city technologies and systems that are fundamentally based on efficient, conscious, and optimized resource utilization [1], [2]. Some commercial activities, for example, those in the construction sector, have been heavily affected by this shift in development. In fact, many building projects now incorporate a considerable array of energy monitoring and management devices, purposeful architecture, and protocols [3], especially inexpensive, nonintrusive, and energy-aware implementations. Consequently, much research has been devoted to the nonintrusive load monitoring (NILM) technique, which has become increasingly popular over the last decade [4], [5], especially as powerful and sophisticated machine learning (ML) approaches have become more readily available [6]. A detailed investigation of the literature in this area reveals that most studies have obtained very high disaggregation accuracy (capability of recognizing home appliances from their aggregated power consumption), but with some significant assumptions, limitations, or hardware requirements.

1) Data sets containing a range of sampling frequencies and combinations of domestic appliances are currently scarce.
2) Expensive micro-controller unit (MCU) with high sampling frequencies are employed.
3) Dedicated ML models with high computational requirements require redesign and training each time an experiment is performed.
4) The few available data sets are based on synthetic or simulated sample generation derived from existing energy scenarios, not real appliance use.

It is also known that if energy consumption is periodically reported to consumers, they are generally motivated by cheaper bills and environmental awareness to modify their energy consumption behaviors. This has been verified in several studies, for example, [7] and [8], which observed savings of 3% to 9% in energy consumption. It is clear that if consumers have detailed information about their energy consumption (not aggregated information, but a more integrated view of how each appliance impacts the total result), they can act directly with particular appliances, and the percentage savings will be higher. This last requirement can be satisfied in two ways:

Manuscript received 2 August 2023; revised 22 October 2023; accepted 30 November 2023. Date of publication 7 December 2023; date of current version 9 April 2024. This work was supported in part by the Czech Ministry of Education, Youth and Sports under Project SP2022/5; in part by e-INFRA CZ under Project ID:90140; and in part by the European Union within the REFRESH Project-Research Excellence for Region Sustainability and High-Tech Industries of the European Just Transition Fund under Grant CZ.10.03.01/00/22 003/0000048. *(Corresponding author: Peppino Fazio.)*

Peppino Fazio is with DSMN, Ca'Foscari University of Venice, 30172 Mestre, Italy, and also with the Faculty of Electrical Engineering and Computer Science, VSB—Technical University of Ostrava, 708 00 Ostrava, Czechia (e-mail: peppino.fazio@unive.it).

Miralem Mehic is with the Department of Telecommunications, Faculty of Electrical Engineering, University of Sarajevo, 71000 Sarajevo, Bosnia and Herzegovina, and also with the VSB—Technical University of Ostrava, 708 00 Ostrava, Czechia (e-mail: miralem.mehic@ieee.org).

Miroslav Voznak is with the VSB—Technical University of Ostrava, 708 00 Ostrava, Czechia (e-mail: miroslav.voznak@vsb.cz).

Digital Object Identifier 10.1109/JIOT.2023.3340423

© 2023 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

1) with an intrusive approach (installing a dedicated meter for each appliance, with additional cabling required as a consequence) or 2) with an NILM approach (a single smart meter (SM) is sufficient to recognize the energy behaviors of every appliance). The current study, therefore, examines a typical domestic scenario in which a single, simple, and inexpensive SM is integrated into the electrical system, and a very simple convolutional neural network (CNN) is trained to recognize appliance behavior once the energy consumption behavior has been correctly encoded into images. This approach is innovative because it avoids the drawbacks of approaches proposed in related works. The strengths of the current study are summarized in the following.

1) A data set created from monitoring real devices in a domestic environment, with a maximum sampling frequency of 1 Hz for sufficiently accurate disaggregation.
2) Use of an inexpensive SM (its cost is around $40).
3) No implementation of a new CNN structure; analysis is based on the well-known proposal by LeCun [9], later extended in [10].
4) A new data set based on real scenarios, not synthetic or simulated data.
5) NILM features are not extracted directly from the sampled data but are first converted into images (produced from the results of moving nodes on a map), thus avoiding the need for a dedicated CNN.
6) The concept of background noise (BN) (BN, not the same as measurement error) is introduced for the first time in an NILM system.
7) An innovative first—energy consumption is not only tracked according to position on a plane but its movement characteristics, i.e., speed and acceleration, are also monitored (thus obtaining a perfect match with mobility analysis theory [11], despite any unnatural variations in position).

The remainder of this article is organized as follows: Section II introduces recent works on image classification, NILM, and data disaggregation; Section III describes the main proposed concept, divided into three sections. Section IV provides a detailed description of the results achievable with the proposed approach; Section V concludes this article and comments on the main advantages of the proposed approach.

II. Related Work

This section reviews the main recent publications on NILM techniques for industrial and domestic environments and studies of CNN accuracy in image recognition. To the best of our knowledge, no data-to-image encoding has ever been integrated with NILM algorithms. The main contributions in these research areas are first reviewed before the proposed approach is described.

Much attention has also been devoted to machine and deep learning (ML and DL) applications to improve their recognition, classification, and prediction capabilities of the main features of several heterogeneous processes in very different research areas [12], [13], [14]. ML/DL algorithms are very suitable and popular options for the NILM technique, hence a wide range of contributions in this research area can be reviewed.

As mentioned in the previous section, we propose a new NILM classification scheme which exploits the strength of CNNs [15] in image recognition and classification, in contrast to the majority of existing works (which implement and evaluate a dedicated ML/DL structure to predict a specific process). No new CNN layering is proposed, rather a completely innovative NILM method with CNN image classification.

A. Recent NILM Approaches for Industrial and Domestic Environments

Before the most recent contributions are reviewed, we cite the most important works in NILM research. A pioneering work introduced by Hart [16] was based on a proposal to change the methods of constructing and identifying admittance-based residential loads. Specifically, the author advanced NILM as a technique which involves several processes: 1) data acquisition [17]; 2) feature extraction [6]; 3) event detection [18]; 4) load identification [19]; and 5) energy disaggregation [20]. From among the works which discuss NILM, the research in [16] represents the current reference for Internet of Things (IoT) development in both domestic and industrial applications, especially the construction industry, even though this article is more than 30 years old. A potential limitation of the approach in Hart's work is the discussion of only ON/OFF devices (plausible with older technologies), which have an average good classification accuracy of approximately 85%.

Dowalla et al. [17] presented an application which shows the consumer, in real time, the energy consumption of each of their appliances. The main drawback of the proposed approach is that it requires two measuring stations to sample appliance behavior at frequencies from several dozen Hz to 1 MHz whereas a laboratory setup is able to monitor 24 sockets with connected appliances and provide more effective disaggregation. Tabanelli et al. [6] are critical of ML/DL approaches for their high computational and memory resource requirements. As a potential solution, they propose the application of MCUs to optimize capacity for features, computational costs and storage. Their proposal is also based on supervised learning, reaching a classification accuracy of approximately 95%. The work in [18] examines the precise extraction of transient and steady-state appliance signatures and proposes an adaptive two-stage event detection method in which dynamic adaptive detection is applied to detect events of different amplitudes. A second analysis is also performed on the geometry of events to capture the features of each appliance's behavior. The idea was tested on both private and public data sets and demonstrated high detection accuracy. Lin and Tsai [19] proposed a time-frequency approach and also examined what happens in the spectrum domain using a multiresolution S-transform-based transient feature extraction scheme. An ant colony optimization algorithm is also proposed for the identification of specific appliances. Raiker et al. [20] described a new model for generating synthetic training data, following the

TABLE I
COMPARISON OF THE MAIN ADVANTAGES (GREEN) AND DISADVANTAGES (RED) OF OUR DENARDO SCHEME AND SOME OF THE MOST RECENT AND IMPORTANT CONTRIBUTIONS IN THE LITERATURE ABOUT NILM

Ref.	Year	Max Accuracy (%)	Expensive device	Only ON/OFF	Own Model	Low Latency	Own Dataset	Back. Noise	Unconv.	Sampling	Scalability	EasM
[5]	2022	97	NO	YES	YES	YES	NO	NO	NO	YES (>4kHz)	1-8	NO
[6]	2022	95, 89.1	YES	NO	YES	YES	YES	NO	NO	1 MHz	1-MULTI	NO
[16]	1992	85	NO	YES	YES	YES	YES	NO	NO	VAR.	1-15	NO
[17]	2021	N.A. in %	YES	NO	YES	YES	YES	NO	NO	(12.5 - 250) kHz	1-20	NO
[18]	2022	93.79-98.98	NO	NO	YES	NO	YES/NO	NO	NO	(1Hz - 12kHz)	8-9-12	NO
[19]	2014	79	YES	NO	YES	NO	YES	NO	NO	YES (2kHz)	4	NO
[20]	2021	N.A. in %	YES	NO	YES	NO	YES	NO	NO	VAR.	6-9	NO
THIS	2023	95.63-100	NO	NO	NO	NO	YES	YES	YES	1Hz	42	YES

Indian standards. IoT devices are also used to feed observable states to the disaggregation model through a hidden Markov model (HMM), improving its accuracy. The proposed method consists of a platform which allows monitoring and control of the selected appliance. Nolasco et al. [5] discussed a very interesting approach to disaggregation involving CNNs and high-frequency NILM signals for the publicly available LIT-Data set [21]. The proposed CNN contains several intermediate layers and performs event-type classification, load identification, and event sample detection, with very high recognition accuracy.

From a detailed reading of the most recent works on NILM, we summarize the main research findings in this area are listed in the following.

1) The higher the sampling frequency, the higher the hardware costs and complexity of the overall NILM system and computational requirements.
2) The higher the sampling frequency, the higher the appliance recognition accuracy since it is also possible to capture transient states (the change in energy consumption between OFF and ON states).
3) Most of the existing works rely on sophisticated swarm intelligence, ML/DL, time-frequency transform algorithms.
4) Appliance recognition assists customers in substituting defective/old devices and preventing critical or building failures.
5) Interest in NILM more than ten years ago was very low, but it has started increasing recently with the advent of the IoT, Smart City, technologies (smart metering and ML/DL algorithms), and greater computational power.
6) Few data sets are publicly available, and most are synthetic.
7) If ML/DL approaches are involved, the main data structure requires a specific definition since no standard reference for the ML/DL approach is available.
8) No studies have examined the option of some appliances always being powered on or the possibility of the recognition system regarding them as BN.

B. ML CNN Approaches for Image Recognition and Classification

CNNs are widely used in NILM applications, but many of the recent published works apply dedicated CNN structures without a standard reference, and most of these CNNs require huge calculation capacities. Instead, the current study proposes converting NILM data into images and then exploiting the accuracy of a CNN and using it for classification and recognition [22]. The theory related to CNN image recognition began with LeCun's proposal [9]; Chen et al. [10] extended this theory with convolutional and pooling layers. These layered CNN types have been tested on different data sets (Cats versus Dogs [25], Cifar-10 [23], and Fer2013 [24]) and achieved very high accuracy in recognizing complex/textured features, outperforming classic methods. Pei et al. [26] examined the problem of degraded images. The significance of the power and usefulness of CNNs in handling motion blurred, noisy, or distorted input images (e.g., images containing haze, fish-eye distortion, underwater distortion, low resolution, poor focus, etc.) is again highlighted. The authors tested and compared a classic CNN and a CNN with prior filtering to remove image degradation. However, the experiment produced a drop in the CNN's image classification performance with degraded images and unsatisfactory performance from the degradation removal algorithms, which were not able to improve the results.

Xu et al. [27] investigated compressed-domain image classification in which images were processed into the desired format before encoding (the reconstruction step is bypassed). Training was then performed using a dynamic measurement rate (MR) by selecting the required MR with the aid of a sensing matrix. The authors tested their proposed method on large data sets (Cifar-10 [23] and Coil-100 [43]), obtaining satisfactory performance and resilience to noise.

Tiwari et al. [28] proposed the visual geometry group 16 (VGG16) model for classifying images into two additional categories instead of performing feature extraction or segmentation. VGG16 offers an accuracy of 99%, and images are further allocated into subcategories.

Table I shows the main pros/cons of our proposal (DENARDO) and some of the most pioneeristic and recent works in literature. Red and green values are considered negative and positive for the works, respectively. The third column (Max Accuracy %) considers the maximum disaggregation and classification accuracy; the term N.A. in % means that the accuracy described in this article is not referred to in percentage but is expressed in power/energy loss respect to the expected values. The fourth column (Expensive Device) refers to the cost of the hardware used for the testbed (expensive is higher than $100). The fifth column (Only ON/OFF) refers to the considered device models, such as only two or transient states. The sixth column (Own model) illustrates if the work

proposes a new model particularized for NILM or if it adapts existing high-performing models, adapting the input data to them. The seventh column (Low Latency) refers to the delay needed for the disaggregation result. The eighth column (Own Data set) refers to using already available online or synthetic data instead of applying the approach to real domestic or industrial environments. The ninth column (BN) considers whether the proposed work considers the set of appliances that remain always powered on. The tenth column (Unconventional) refers to the application of models that, typically, are not used in the NILM research topic. The 11th column (Sampling) refers to the sampling frequency used to collect samples, which is preferred to be high. The 12th column refers to the number of considered appliances to be recognized. In contrast, the last column (EasM, which stands for Energy as Mobility) is related to the application of the proposed model and other research fields. It is evident how the proposed idea outperforms the other approaches, except for the latency and sampling time, given that the hardware is limited to a minimum sampling period of 1 s. This is why we limited our investigation to domestic environments, where some seconds of delay can be tolerated during the disaggregation operations (if, for example, some automated actions need to be taken in the function of the powered-on appliance). The proposed NILM model and approach in the current study builds on the works reviewed above and is described in the next section.

III. Domestic NILM for Image Appliance Recognition Based on Harts' Domain (DENARDO)

Beginning with Hart's work [16] and incorporating the findings from other studies, we present a new method of identifying and recognizing appliance consumption. We start from Hart's energy consumption Domain ($HD \in \mathbb{R}^2$) and then extend and apply it to data-to-image conversion and CNN classification.

A. Initial Analytical Model, Notation, and ON/OFF Event Recognition

Let the Domestic Appliances Set be defined as $DAS = \{a_1, a_2, a_i, \ldots, a_n\}$, where $|DAS| = n$. When powered ON, all appliances in DAS consume energy continuously over time (statically or dynamically), but only the overall consumption in period T (sampling period) at sampling frequency $f = (1/T)$ can be sampled. The continuous variable t is then replaced with kT, or simply k (for shorter notation, given that T is fixed), in preparation for a discrete-time analytical/statistical approach.

Each $a_i \in DAS$ is characterized by its active power consumption $a_i^A(k)$, reactive power $a_i^R(k)$ (it can be seen as a part of complex power that corresponds to storage and retrieval of energy rather than consumption), voltage drop (or distortion, if any) $a_i^V(k)$, spectrum (or frequency) components $a_i^F(k)$, and behavior a_i^B. Therefore, in general, at a discrete time k, the overall measurable features are

$$\begin{aligned} A(k) &= b_1^k \cdot a_1^A(k) + \cdots + b_i^k \cdot a_i^A(k) + \cdots + b_n^k \cdot a_n^A(k) \\ R(k) &= b_1^k \cdot a_1^R(k) + \cdots + b_i^k \cdot a_i^R(k) + \cdots + b_n^k \cdot a_n^R(k) \\ V(k) &= b_1^k \cdot a_1^V(k) + \cdots + b_i^k \cdot a_i^V(k) + \cdots + b_n^k \cdot a_n^V(k) \\ F(k) &= b_1^k \cdot a_1^F(k) + \cdots + b_i^k \cdot a_i^F(k) + \cdots + b_n^k \cdot a_n^F(k) \quad (1) \end{aligned}$$

where b_i^k is a binary value set to 1 if $a_i \in DAS$ is powered ON at time k or set to 0 if $a_i \in DAS$ is powered OFF at time k. The concept of behavior and the variables in (1) are explained in the next section.

Defining the vectors $\vec{b(k)} = [b_1^k, \ldots, b_n^k]$, $\vec{a(k)} = [a_1^A(k), \ldots, a_n^A(k)]$, $\vec{r(k)} = [a_1^R(k), \ldots, a_n^R(k)]$, $\vec{v(k)} = [a_1^V(k), \ldots, a_n^V(k)]$, $\vec{f(k)} = [a_1^F(k), \ldots, a_n^F(k)]$, (1) become (in more compact form)

$$\begin{aligned} A(k) &= \vec{b(k)}^T \cdot \vec{a(k)}, \; R(k) = \vec{b(k)}^T \cdot \vec{r(k)} \\ V(k) &= \vec{b(k)}^T \cdot \vec{v(k)}, \; F(k) = \vec{b(k)}^T \cdot \vec{f(k)}. \quad (2) \end{aligned}$$

Let us assume that, for domestic equipment, a single SM outputs, at time k, the triplet $SM_{\text{out}}(k) = [A(k), R(k), V(k)]$, which represents the aggregate appliance consumption at time k. An explanation and evaluation of $F(k)$ is given later.

To correctly relate the separate trends of $A(k)$ and $R(k)$ in a generic measurement, we recall the definition of complex power (the representation of power by a complex number, where $j^2 = -1$ in this case) for appliance $a_i \in DAS$

$$\mathbf{S_i}(k) = A_i(k) \pm j \cdot R_i(k) \quad (3)$$

where $|\mathbf{S_i}(k)| = \sqrt{A_i^2(k) + R_i^2(k)}$ (defined as *apparent power*, measured in [VA]), and the sign depends on the load type, positive (+) for inductive load and negative (−) for capacitive load. Bold notation is used for complex numbers and phasors. Regarding the phase for $\mathbf{S_i}(k)$, we have $\underline{/\mathbf{S_i}(k)} = \arctan \pm[(R_i(k))/(A_i(k))]$. Therefore, in general, an increase in $A_i(k)$ implies a decrease in $R_i(k)$ and vice versa, for a stable value of $|\mathbf{S_i}(k)|$. From (3), we can state that A and R are joint processes, a useful concept for later. For completeness, we recall that in energy/power analysis, the following relationship is valid:

$$\mathbf{S_i}(k) = \frac{1}{2}\mathbf{V_i}(k) \cdot \mathbf{I_i}^*(k) = V_{i_{rms}}(k) I_{i_{rms}}(k) e^{j\Delta_{\theta_i}(k)} \quad (4)$$

where the root mean square (rms) value of $X(k)$ is $(X(k)/\sqrt{2})$ and $\Delta_{\theta_i}(k)$ is the phase displacement between the $\mathbf{V_i(k)}$ and $\mathbf{I_i(k)}$ phasors, equal to $\theta_i^V(k) - \theta_i^I(k)$. Finally

$$\frac{A_i(k)}{\sqrt{A_i^2(k) + R_i^2(k)}} = \cos\left[\Delta_{\theta_i}(k)\right] \quad (5)$$

describes the *power factor*, which lets us evaluate the phase displacement between voltage and current phasors according to the equation

$$\theta_i^V(k) - \theta_i^I(k) = \arccos\left[\frac{A_i(k)}{\sqrt{A_i^2(k) + R_i^2(k)}}\right]. \quad (6)$$

The values $\Delta_{\theta_i}(k)$ in (6), as with all the other defined values, can be easily de-noised with a median filter [34], [35], which is $O(W^2)$ for a mono-dimensional variable, where W is the number of observed samples. We are mainly interested in

$\Delta_{\theta_i}(k)$, because from the observation of several $A_i(k)$, $R_i(k)$, $V_i(k)$, and $\Delta_{\theta_i}(k)$ courses, the phase displacement curve has a more stable trend [than $R_i(k)$ and $V_i(k)$], with a good amplification of ON/OFF events [greater than $A_i(k)$].

Given the set of points Δ_{θ_i} and an odd value $W \geq 3$, we can thus obtain the output of the median filter at a point k ($\overline{\Delta}_{\theta_i}(k)$) applied to Δ_{θ_i} by defining

$$\Delta_{\theta_i}^{W}(k) = \left\{ \Delta_{\theta_i}\left(k - \lfloor \frac{W}{2} \rfloor\right), \ldots, \Delta_{\theta_i}(k), \ldots \right. \\ \left. \Delta_{\theta_i}\left(k + \lfloor \frac{W}{2} \rfloor\right) \right\} \quad (7)$$

and taking

$$\overline{\Delta}_{\theta_i}(k) = \begin{cases} \text{MED}\left(\Delta_{\theta_i}^{W}(k)\right), & \text{if } k > \lfloor \frac{W}{2} \rfloor \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

where MED(X) is the median operator for odd arrays, i.e., MED(X) = $X[(W+1)/2]$, with $|X| = W \geq 3$, and k is changed for covering the entire Δ_{θ_i} array (W can be also an even value, but the notations for the integer index rounding should be changed). Equations (3)–(8) remain valid also for aggregate values of powers $A(k)$, $R(k)$ and voltage $V(k)$ by simply removing the subscript i.

Fig. 1 graphs a typical 350-s trend ($T = 1$, 350 samples, $n = 3$) of power consumption. The red curve represents the sampled active power (at $k = 16$, appliance a_1 is turned OFF, then it is turned back ON at $k = 70$; at $k = 202$, appliance a_2 is turned ON, then OFF at $k = 253$; at the end, appliance a_3 is turned ON at $k = 266$, then turned OFF at $k = 297$, for a total of six events; from $k = 18$ to $k = 68$ the three appliances are turned OFF, but consumption is not zero because of background activity, explained in the next section). For a complete description, a_1 is a 75 IPS-LED Smart TV, a_2 is a 120-W incandescent lamp, and a_3 is an oven. The blue curve represents $R(k)$, the green represents $V(k)$, and the black curve with its square-like cyan approximation represents $\Delta_\theta(k)$ and its median-filtered version ($W = 10$). It must be noted that the trends of $A(k)$ and $R(k)$ are opposite during and after the ON/OFF events [confirming their joint nature, mentioned above after (3)]. When $a_i \in DAS$ is powered ON, $A(k)$ has a positive spike whereas $R(k)$ indicates a negative; when a_i is powered OFF, $A(k)$ continues to decrease while $R(k)$ increases.

Once the filtered aggregate phase displacement array $\overline{\Delta}_\theta$ has been obtained, it is possible to dynamically evaluate when an ON/OFF event occurs (the ON/OFF concept will be extended with state-change events in the next sections). Several approaches can be applied to recognize events [36], [37], [38] in both the time and frequency domains, but they are based on complex implementations and require relatively expensive hardware and metering systems. More simply, the proposal here is based first on event recognition and then on appliance classification. Given the two W-sized arrays $\overline{\Delta}_\theta(k)$ and $\overline{\Delta}_\theta(k+1)$, we can therefore evaluate whether the following inequality is satisfied:

$$\Delta(k) = \frac{1}{W} \cdot \left| \sum_{q=1}^{W} \overline{\Delta}_\theta(k, q) - \sum_{q=1}^{W} \overline{\Delta}_\theta(k+1, q) \right| > \theta_{Thr} \quad (9)$$

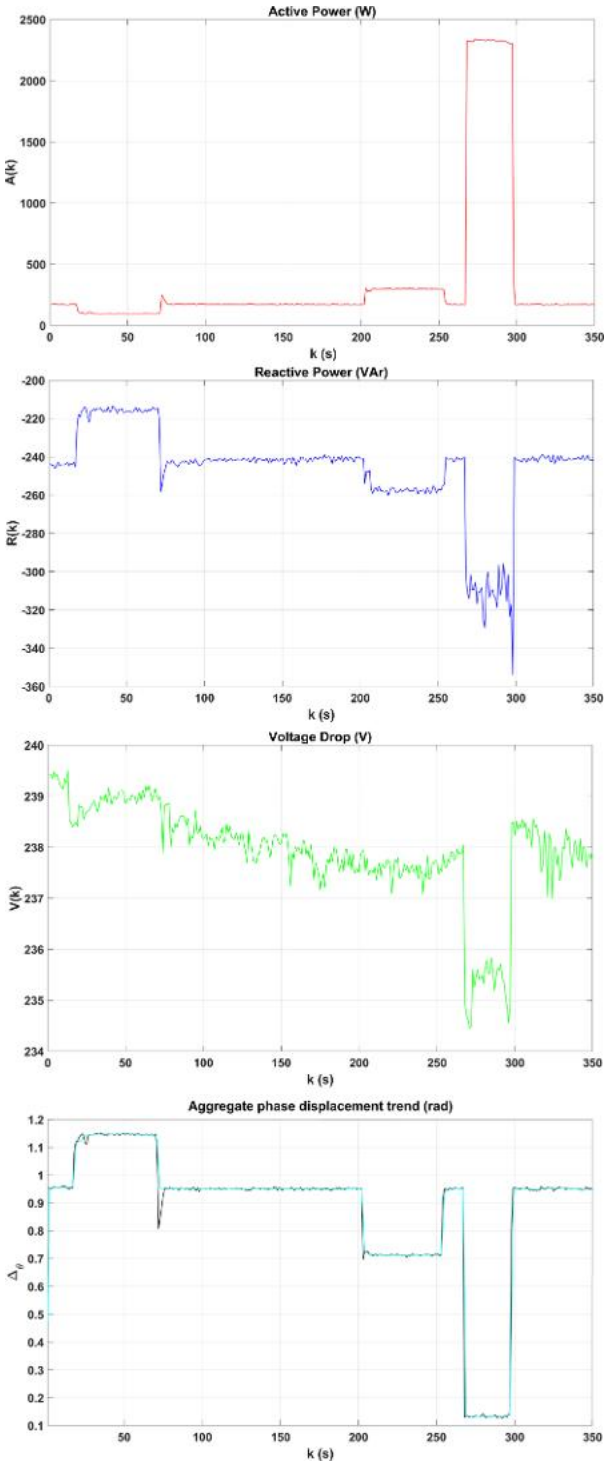

Fig. 1. Simple example of $A(k)$, $R(k)$, $V(k)$, $\Delta_\theta(k)/\overline{\Delta}_\theta(k)$ trends for 350 samples in a domestic environment.

where $\overline{\Delta}_\theta(k, q)$ is the qth element of $\overline{\Delta}_\theta(k)$, as defined in (7) and (8). Using (9), the monitoring system evaluates whether a change has occurred in the average value of the phase displacement samples ($\Delta(k)$) between the W-sized windows, evaluated at kT and $(k+1)T$. If $\Delta(k)$ is greater than θ_{Thr} and the quantity inside $|\cdot|$ is positive, then an OFF event occurred (if the term is negative, an ON event occurred). The relationship of (9) continues until a new, stable consumption

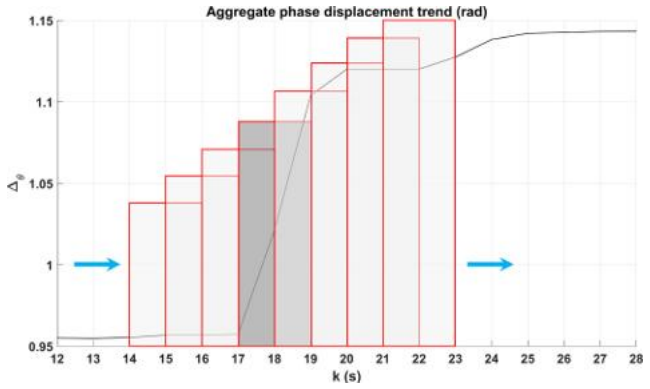

Fig. 2. Magnification of $\overline{\Delta}_\theta$ in Fig. 1, from $k = 12$ to $k = 28$, with the sliding window approach ($W = 3$), starting at $k = 15$.

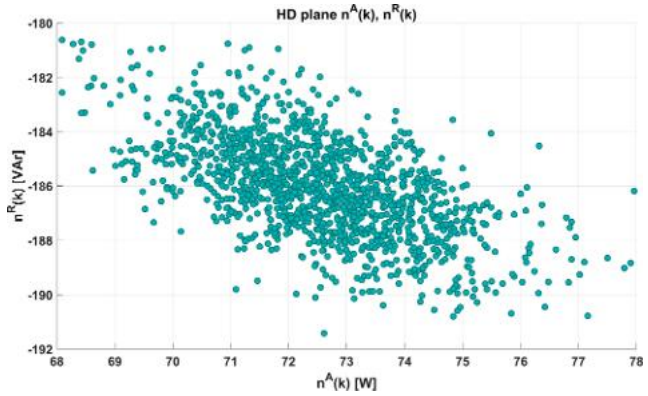

Fig. 3. Example of the BN HD plane for 1800 samples.

condition is reached, and thus event recognition is triggered once only, until $\Delta(k) \leq \theta_{Thr}$.

Fig. 2 illustrates how the sliding window approach works: the trend from $k = 15$ is monitored, and the window moves to the right each kT ($T = 1$ s in the example). The vertical size of each represented window increases to distinguish it from others. For each step, $\Delta(k)$ is evaluated and compared to $\Delta(k+1)$, and if the relationship of (9) is satisfied (we assume at $k+1 = 18$, for example), then an event is recognized. The sliding window approach may also be based on the standard deviation or the mean squared error (MSE) [39], but the critical matter is the choice of W and θ_{Thr}. The numerical results section below demonstrates how the best values for these two parameters are determined, taking into account historical traces.

Another assumption in the model is that the effects of energy sources (e.g., photovoltaic systems, inverters, or batteries) in the domestic environments are neglected. It is possible today to interface devices directly and know the exact energy production for each time kT. We therefore assume that no sources are present and that if one or more energy sources belong to the environment, their effects are neglected since it is always possible to know the instantaneous generated energy.

The main problem can then be formulated as follows: *given the triplet* $SM_{\text{out}}(k)$ *and* $F(k)$, *the elements of* $\vec{b(k)}$ are evaluated to determine which appliances are powered ON or OFF.

B. Background Noise Distribution, Appliance Behaviors, and Energy Consumption as Unnatural Mobility Process

Another important contribution of the research presented here in contrast to other works is in accounting for BN. As stated in Hart's proposal [16], each $a_i \in DAS$ behaves differently because of its nature. For example, a nonadjustable lamp a_i can switch between two states only, therefore the behavioral concept which applies to the lamp is the binary set $a_{i_1}^B = \{0, 1\}$. A different case is a heat pump a_{i_2}, which attains several consumption levels as a function of its required heating output, in this case $a_{i_2} = \{0, 1, 2, \ldots, l\}$.

For each moment in time, we therefore have $a_{i_1}(k) \in a_{i_1}^B$ and $a_{i_1}(k) \in a_{i_1}^B$. We also assume that the stand-by (SB) mode in electronic devices is different from the OFF status (although the device is not in use, it consumes a very small amount of energy).

It should also be clearly noted that some appliances are usually never powered OFF, for example a refrigerator (it functions autonomously), an electronic oven (it maintains a display which is ON at all times), an electronic heating/cooling machine (it autonomously enters SB mode), a television set (it waits for IR inputs), and many others.

Defining DAS_{noise} as a strictly contained subset of DAS, $DAS_{\text{noise}} \subset DAS$, with $|DAS_{\text{noise}}| = m < n$, the BN can be expressed as the triplet

$$n^A(k) = a_1^A(k) + \cdots + a_j^A(k) + \cdots + a_m^A(k) = \sum_{j=1}^{m} a_j^A(k)$$
$$n^R(k) = a_1^R(k) + \cdots + a_j^R(k) + \cdots + a_m^R(k) = \sum_{j=1}^{m} a_j^R(k)$$
$$n^V(k) = a_1^V(k) + \cdots + a_j^V(k) + \cdots + a_m^V(k) = \sum_{j=1}^{m} a_j^V(k) \quad (10)$$

which are the aggregate energy contributions of the appliances acting autonomously or remaining in STAND-BY (i.e., they are never OFF).

As an example of the typical trends obtained from (10), we collected data without any appliance activity, only DAS_{noise} devices, from the period 1 A.M. to approximately 1.30 A.M. (1800 samples, with $T = 1$) and obtained the BN HD plane (neglecting n^V for now) plotted in Fig. 3.

Fig. 3 gives a detailed view of what happens during the OFF modes of $n-m$ appliances. A well-defined consumption cluster is clearly distinguishable. With even more detail, the BN shape can be further investigated and provide deeper insight into what happens with *background appliances*. Figs. 4 and 5 graph the pdf of the data cluster depicted in Fig. 3, in terms of n^A and n^R, respectively: we can conclude that we are dealing with Gaussian distributions.

To correctly relate the separate trends of n^A and n^R depicted in Figs. 4 and 5, we recall the definition of complex power given by (3). In this manner, a precise match between the data in Figs. 3–5 can be found by reading the n^A pdf from left to right and the n^R pdf from right to left. The values obtained for

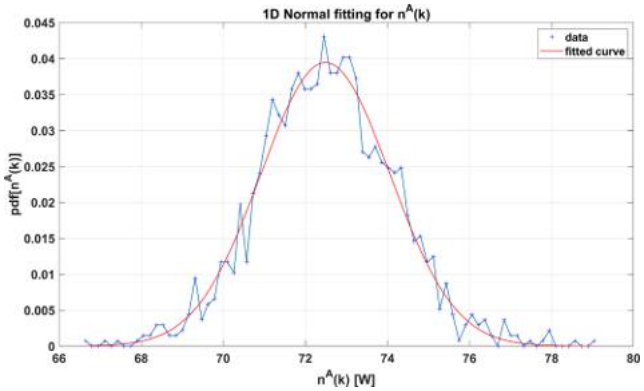

Fig. 4. Active *HD* pdf trend and its normal fit in relation to the BN generated by the m DAS_{noise} devices.

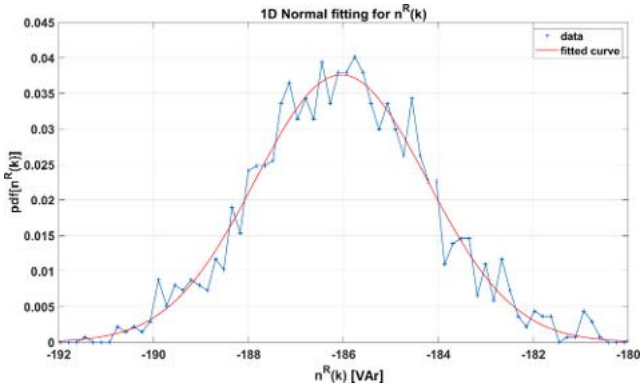

Fig. 5. Reactive *HD* pdf trend and its normal fit in relation to the BN generated by the m DAS_{noise} devices.

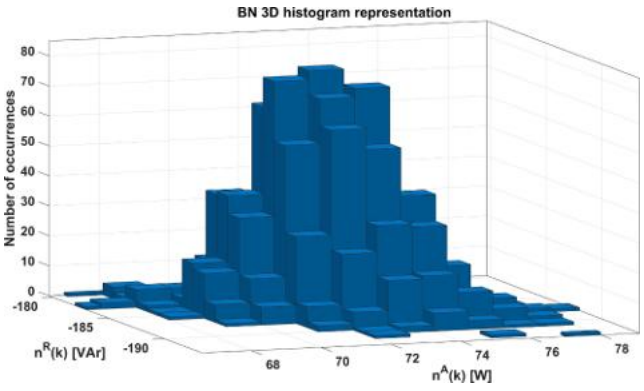

Fig. 6. 3-D Histogram of the BN data represented in Fig. 3.

the normal fit are $\mu^A = 72.5294$ and $\sigma^A = 1.6466$ for n^A, and $\mu^R = -186.0022$ and $\sigma^R = 1.8783$ for n^R. It is interesting to observe the correlation between the two processes; the data in Fig. 3 can represented differently with a 3-D histogram, as in Fig. 6. From (3) and Fig. 6, we can assume that $pdf(n^A)$ and $pdf(n^R)$ correlate (movements on the HD plane are possible only from left to right with a decreasing n^R trend and from right to left with an increasing n^R trend), therefore the joint density probability of the bi-variate variable (n^A, n^R) can be defined as (we avoid using the k argument for simplicity of

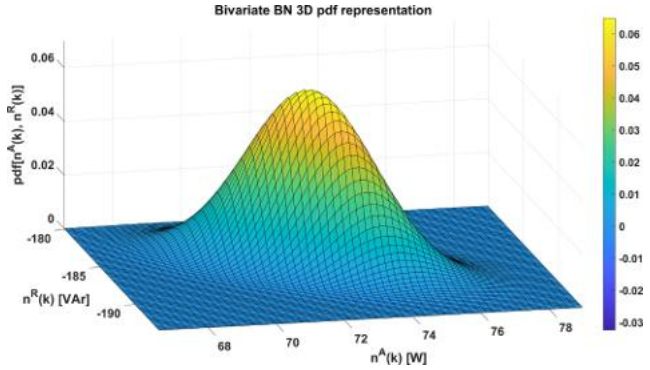

Fig. 7. Pdf trend for joint bivariate (n^A, n^R) variable.

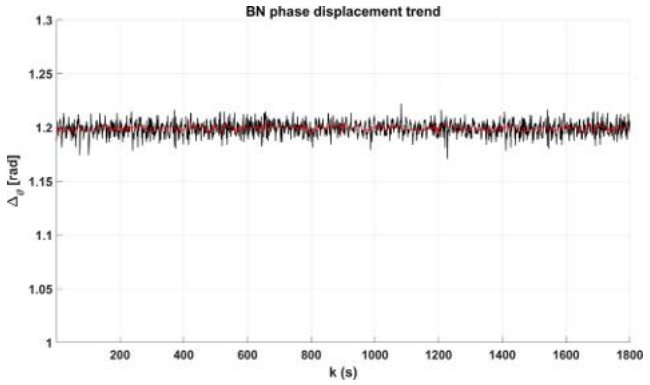

Fig. 8. $\Delta_\theta(k)$ (black) $\overline{\Delta}_\theta(k)$ (red) trends for BN bivariate data represented in Fig. 3.

notation)

$$f\left[\left(n^A, n^R\right), (\mu_A, \mu_R), \Sigma_A^R\right] = \frac{1}{2\pi\sqrt{|\Sigma_A^R|}} \cdot \exp \left\{-\frac{1}{2} \cdot \left[\left(n^A, n^R\right) - (\mu_A, \mu_R)\right] \cdot \left(\Sigma_A^R\right)^{-1} \cdot \left[\left(n^A, n^R\right) - (\mu_A, \mu_R)\right]'\right\} \tag{11}$$

where Σ_A^R is the covariance matrix of the joint process and $|\Sigma_A^R|$ is its determinant. We recall that the variances of the two variables of Σ_A^R are on the main diagonal and that the covariances between variables are off-diagonal elements. From (11) and the previously obtained parameters (the data plotted in Figs. 3 and 6), an ideal shape is obtained and depicted in Fig. 7 (to clarify the idea of the statistical trend).

Without additional perturbations (ON/OFF events), we expect the measured phase shift to be mostly constant. Fig. 8 in fact indicates that the average value of $\overline{\Delta}_\theta(k)$ is 1.199 rad, the maximum is 1.2064 rad, and the minimum is 1.1918 rad, with a negligible variance of 3.4736e-06 rad^2.

As a novel contribution, the current study proposes an extension to the *HD* plane: Hart's concept is based on data classification in a 2-D domain, i.e., a cluster of points on a surface refers to an appliance. Hart's concept describes how these points are positioned on the plane and the explicit manner in which the points create clusters on that *HD* plane

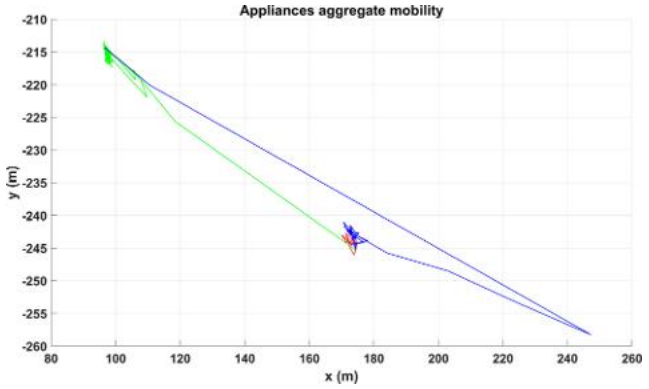

Fig. 9. Mobility CP for the first 100 samples of Fig. 1.

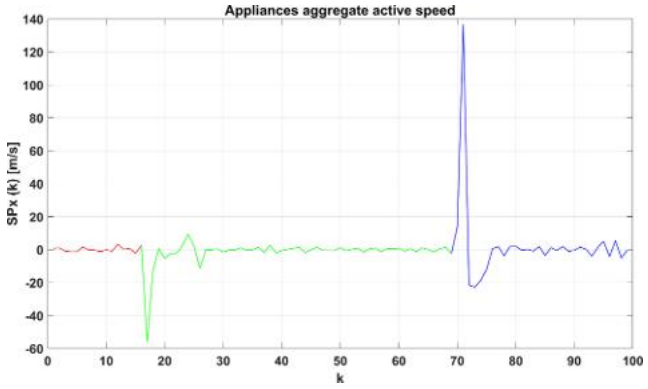

Fig. 10. Active speed ($SP(k)$'s x component) of the CP relative to the first 99 samples of Fig. 1.

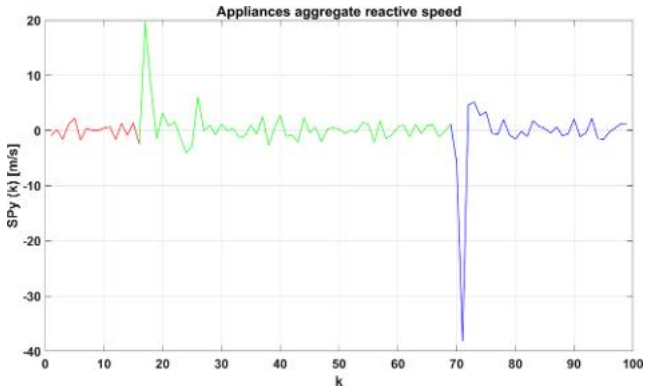

Fig. 11. Reactive speed ($SP(k)$'s y component) of the CP relative to the first 99 samples of Fig. 1.

(the manner in which they form their own cluster or move within that cluster).

To incorporate this behavior, each appliance is considered as a mobile node which moves into a 2-D geographical area (the dimensions depend on the minimum and maximum cluster locations). To avoid adding more variables, the notation used until now ($a_i \in DAS$) can be applied to indicate both appliances and mobile nodes. Each node possesses its own mobility behavior (not only composed of visited points), therefore it is possible to obtain a single *HD* plane for position, speed, and acceleration. For a single appliance/node, each point on the *HD* plane can be converted into a location in a geographical area according to the following relationship:

$$P_{a_i}(x, y, k) = \left(\alpha_p \cdot a_i^A(k), \beta_p \cdot a_i^R(k)\right) = P_{a_i}(k) \tag{12}$$

where $\alpha_p = 1$ m/W and $\beta_p = 1$ m/VAr are two coefficients for translating power units into Cartesian meters.

For the aggregate points on the Cartesian plane (CP), we have [recalling the notation of (2)]

$$\begin{aligned} P(k) &= \left(\alpha_p \cdot \sum_{i=1}^{n} b_i \cdot a_i^A(k),\ \ \beta_p \cdot \sum_{i=1}^{n} b_i \cdot a_i^R(k)\right) \\ &= \left(\alpha_p \cdot A(k),\ \ \beta_p \cdot R(k)\right). \end{aligned} \tag{13}$$

To illustrate this concept, let us examine the first 100 samples in Fig. 1 after conversion with (13).

Fig. 9 depicts the aggregate path $P(k)$ of $n = 3$ nodes (appliances) over a period of 100 s ($T = 1$). The red part of the trajectory describes the first 16 samples, then the active (reactive) power decreases (increases) and another mobility cluster is created (green trajectory), lasting 53 s, and again the active (reactive) power increases (decreases) and the trajectory returns to the starting cluster (blue curve) in the last 30 s.

At this point, although these obtained aggregated movements (13) are not natural, the concepts of appliance speed [$SP(k)$] and acceleration [$AC(k)$] can also be defined and analyzed in detail

$$\begin{aligned} sp_i^A(k) &= \frac{a_i^A(k+1) - a_i^A(k)}{T} \\ sp_i^R(k) &= \frac{a_i^R(k+1) - a_i^R(k)}{T} \end{aligned} \tag{14}$$

and then

$$SP(k) = \left(\alpha_{sp} \cdot \sum_{i=1}^{n} b_i \cdot sp_i^A(k), \beta_{sp} \cdot \sum_{i=1}^{n} b_i \cdot sp_i^R(k)\right) \tag{15}$$

where $\alpha_{sp} = 1$ m/(s·W) and $\beta_{sp} = 1$ m/(s·VAr) are the translating coefficients for the $SP(k)$ process.

The same definitions are valid for $AC(k)$

$$\begin{aligned} ac_i^A(k) &= \frac{sp_i^A(k+1) - sp_i^A(k)}{T} \\ ac_i^R(k) &= \frac{sp_i^R(k+1) - sp_i^R(k)}{T} \end{aligned} \tag{16}$$

and then

$$AC(k) = \left(\alpha_{ac} \cdot \sum_{i=1}^{n} b_i \cdot ac_i^A(k), \beta_{ac} \cdot \sum_{i=1}^{n} b_i \cdot ac_i^R(k)\right) \tag{17}$$

where $\alpha_{ac} = 1$ m/(s^2·W) and $\beta_{ac} = 1$ m/(s^2·VAr) are the translating coefficients for the $AC(k)$ process.

Using the definitions above, the appliance derivative behavior can be described (in terms of energy consumption) with the aggregate speed, as in Fig. 10.

Let us represent $SP(k)$ as two separate figures (one for the x coordinate and one for the y coordinate) for clarity and a better demonstration of its meaning. Figs. 10 and 11 reveal that ON/OFF (or changing state) events are obvious. Figures 12 and 13 illsutrates the acceleration trend. The three colors depict the three data clusters illustrated in Fig. 9, and

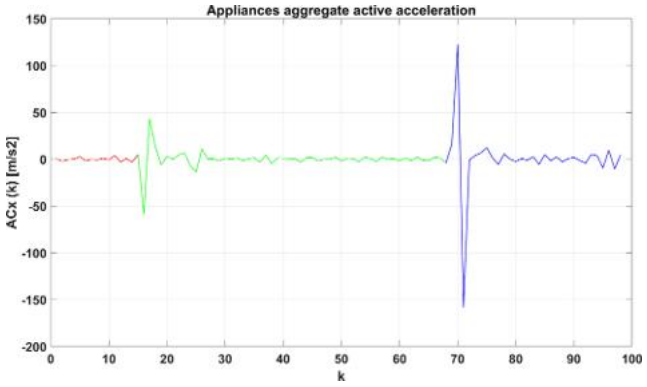

Fig. 12. Reactive acceleration ($AC(k)$'s x component) of the CP relative to the first 98 samples of Fig. 1.

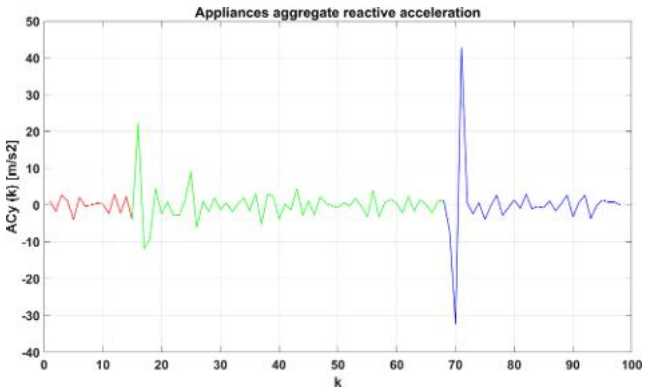

Fig. 13. Reactive speed ($AC(k)$'s y component) of the CP relative to the first 98 samples of Fig. 1.

the transient trends (positive/negative spikes) indicate how aggregated mobility migrates from one cluster to another. For a complete illustration, we also describe what happens with acceleration. Given the relative low sampling frequency ($T = 1$ s, i.e., $f = 1$ Hz), the transient mobility points (corresponding to ON, OFF, or a change in state) appear to be segmented. The mobility values here (especially speed and acceleration) are undoubtedly unnatural, but they are easy to evaluate in real time and allow additional features to be added to the *HD* plane.

C. Appliance Recognition Through Data-to-Image Encoding

Classic clustering methods, such as those applied in [30] and [31] are not suitable with NILM because, while the number of possible clusters (directly related to each node/appliance) is linearly dependent on $n-m$, the aggregated results are exponential, i.e., proportional to 2^{n-m}, and taking into account all possible combinations is not efficient from a computational point of view. For example, if $n-m = 15$ (i.e., few appliances in a house, excluding *background* devices), the maximum number of possible states (ON/OFF combinations) is exponential and equal to $2^{15} = 32\,768$. It is preferable to train and validate the classification module on a huge number of classes, and hence, it would not be feasible. Mainly for this reason, but also to propose a light and scalable approach, we propose a new appliance recognition scheme which avoids this

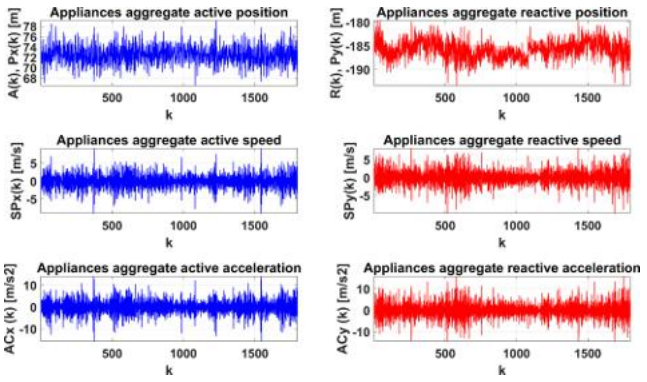

Fig. 14. Position, speed, and acceleration of the BN over a period of 1800 samples (s).

problem by working on a linear scale instead of an exponential scale. First, the following realistic assumptions apply.

1) The monitoring system activates only when *BN* is present, i.e., when the other $n-m$ nodes are powered *OFF*. This implies that the final part of the vector $\vec{b(k)} = [b^k_{n-m+1}, \ldots, b^k_n]$ is initially set to $[0, \ldots, 0]$ (equivalently, we can say that the indexes for all nodes which do not create *BN* are placed at the origin of the *HD*'s plane, with zero speed/acceleration). Similarly, the first part of the vector $\vec{b(k)} = [b^k_1, \ldots, b^k_m]$, is always set to $[1, \ldots, 1]$.
2) If two different appliances are not powered *ON*/*OFF* at the same time, at least W samples should be between two consecutive events.
3) The power *ON* status of each appliance should last for at least W seconds.
4) When the algorithm begins, it needs at least $2 * W$ samples before starting its analysis of event recognition, as in (9).

BN has already been characterized as a bivariate Gaussian process, hence its position, speed, and acceleration also require definition. The results for generic aggregate consumption are then generalized.

Several measurement campaigns suggest that *BN*'s related variables can be assumed to be Gaussian processes (given as the sum of each appliance contribution, which is also Gaussian). In other words, we already know that $n^A \sim \mathcal{N}(\mu_{n^A}, \sigma^2_{n^A})$, and $n^R \sim \mathcal{N}(\mu_{n^R}, \sigma^2_{n^R})$, but from (13), (15), and (17), we can also write, for DAS_{noise} nodes

$$\sum_{j=1}^{m} sp_j^{n^A} \sim \mathcal{N}\left(0,\ \sigma^2_{sp^{n^A}}\right),\ \sum_{j=1}^{m} sp_j^{n^R}, \sim \mathcal{N}\left(0,\ \sigma^2_{sp^{n^R}}\right)$$
$$\sum_{j=1}^{m} ac_j^{n^A} \sim \mathcal{N}\left(0,\ \sigma^2_{ac^{n^A}}\right),\ \sum_{j=1}^{m} ac_j^{n^R}, \sim \mathcal{N}\left(0,\ \sigma^2_{ac^{n^R}}\right) \quad (18)$$

where the variances are easily measurable.

Fig. 14 in fact shows that it is possible to evaluate the BN immediately (in terms of position, speed, and acceleration) and determine the distribution of its values (Fig. 15).

For complete details, we point out that, for the *BN* evaluation in the previous figures, m was set from 2 to 8 (a router and an energy meter), and the trend

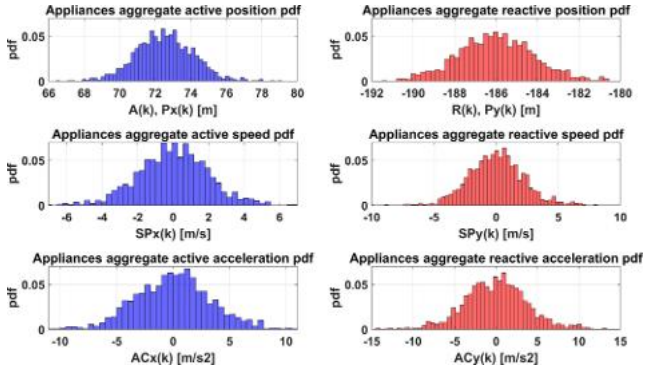

Fig. 15. BN position, speed, and acceleration pdfs in a period of 1800 samples (s).

was always Gaussian. Variances, however, tended to be higher as m increased (according to [32], [33], and [40]). The obtained statistical parameters were $\mu_{n^A} = 72.53$ m, $\sigma^2_{n^A} = 2.711$ m, $\mu_{n^R} = -186$ m, $\sigma^2_{n^A} = 3.528$ m, $\sigma^2_{sp^{n^A}} = 4.1628$ m/s, $\sigma^2_{sp^{n^R}} = 5.07$ m/s, $\sigma^2_{ac^{n^A}} = 12.08$ m/s^2, and $\sigma^2_{ac^{n^R}} = 15.0846$ m/s^2.

Generally, given a random variable X which follows a given distribution characterized by its probability mass function (or discrete density function) $p_X(x)$ or $f_X(x) = pdf_X(x)$ for continuous variables, the characteristic function of X, $\varphi_X(\omega)$ can be evaluated as a discrete Fourier transform [(DFT), or the FT in the case of continuous variables] of $p(x)$ or $pdf_X(x)$, i.e., $\mathcal{F}[pdf_X(x)] = \varphi_X(\omega)$. Here, we focus on continuous random processes derived from samples, but the variables can assume values which are neither discrete nor limited in any way.

In the case of a domestic environment, the variables $n^A(k)$ and $n^R(k)$, which represent the aggregate positions of BN on a plane, are normally distributed $\mathcal{N}(\mu_{n^A}, \sigma^2_{n^A})$. The random variables for BN's speed and acceleration [as defined in (18)] also follow this distribution, and therefore, we can write their pdfs as:

$$P\left(n^A(k) = x\right) = f_{n^A(k)}(x) = \frac{1}{\sqrt{2\pi\sigma^2_{n^A}}} \cdot e^{-\frac{\left(x-\mu_{n^A}\right)^2}{2\sigma^2_{n^A}}}$$

$$P\left(n^R(k) = x\right) = f_{n^R(k)}(x) = \frac{1}{\sqrt{2\pi\sigma^2_{n^R}}} \cdot e^{-\frac{\left(x-\mu_{n^R}\right)^2}{2\sigma^2_{n^R}}} \quad (19)$$

and their characteristic functions $\varphi_{n^A(k)}(\omega)$, $\varphi_{n^R(k)}(\omega)$ as the FT of $f_{n^A(k)}(x)$ and $f_{n^R(k)}(x)$ [32]

$$\varphi_{n^A(k)}(\omega) = \frac{1}{\sqrt{2\pi\sigma^2_{n^A}}} \cdot \int_{-\infty}^{+\infty} e^{-\frac{\left(x-\mu_{n^A}\right)^2}{2\sigma^2_{n^A}}} \cdot e^{j\omega x}\, dx$$

$$\varphi_{n^R(k)}(\omega) = \frac{1}{\sqrt{2\pi\sigma^2_{n^R}}} \cdot \int_{-\infty}^{+\infty} e^{-\frac{\left(x-\mu_{n^R}\right)^2}{2\sigma^2_{n^R}}} \cdot e^{j\omega x}\, dx \quad (20)$$

which can be written as in [33]

$$\varphi_{n^A(k)}(\omega) = e^{\frac{1}{2}\omega^2\sigma^2_{n^A} + j\omega\mu_{n^A}}, \quad \varphi_{n^R(k)}(\omega) = e^{\frac{1}{2}\omega^2\sigma^2_{n^R} + j\omega\mu_{n^R}} \quad (21)$$

where j is the imaginary unit. The expression in (21) is significant, because it provides the means to apply a denoising algorithm, as illustrated below. Avoiding a repetition of the above illustration for speed and acceleration, we can rewrite (21) as follows:

$$\varphi_{sp^{n^A}(k)}(\omega) = e^{\frac{1}{2}\omega^2\sigma^2_{sp^{n^A}}}, \quad \varphi_{sp^{n^R}(k)}(\omega) = e^{\frac{1}{2}\omega^2\sigma^2_{sp^{n^R}}}$$
$$\varphi_{ac^{n^A}(k)}(\omega) = e^{\frac{1}{2}\omega^2\sigma^2_{ac^{n^A}}}, \quad \varphi_{ac^{n^R}(k)}(\omega) = e^{\frac{1}{2}\omega^2\sigma^2_{a^{n^R}}} \quad (22)$$

where the imaginary component is null since the mean values are zero (18).

Highlighting some important properties of the characteristic function, we recall that given two independent, normally distributed random variables X and Y, the variable $Z = X + Y$ will have the characteristic function [32], [33]

$$\varphi_Z(\omega) = \varphi_X(\omega) \cdot \varphi_Y(\omega). \quad (23)$$

Taking into account (21), (23) becomes

$$\varphi_Z(\omega) = e^{\frac{1}{2}\omega^2(\sigma^2_X + \sigma^2_Y) + j\omega(\mu_X + \mu_Y)} \quad (24)$$

which corresponds to $Z = \mathcal{N}(\mu_X + \mu_Y, \sigma^2_X + \sigma^2_Y)$. In the same manner and under the same assumptions, it can be shown that if $Z = X - Y$, then $Z = \mathcal{N}(\mu_X - \mu_Y, \sigma^2_X + \sigma^2_Y)$. The expression can then be generalized as follows: given a set of independent, normally distributed random variables $\{X_1, \ldots, X_r\}$, if $Z = X_1 + \cdots + X_r$, then:

$$f_Z(x) = \mathcal{N}\left(\sum_{i=1}^{r} \mu_{X_i}, \sum_{i=1}^{r} \sigma^2_{X_i}\right) \quad (25)$$

and, in the same manner, if $Z = X_1 - \cdots - X_r$

$$f_Z(x) = \mathcal{N}\left(\mu_{X_1} - \sum_{i=2}^{r} \mu_{X_i}, \sum_{i=1}^{r} \sigma^2_{X_i}\right). \quad (26)$$

Hence, in terms of probability density functions and neglecting the voltage trend and discrete time variable k to improve readability, (10), (15), and (17) become

$$f_{n^A}(x) = \mathcal{N}\left(\sum_{j=1}^{m} \mu_{a^A_j}, \sum_{j=1}^{m} \sigma^2_{a^A_j}\right)$$
$$f_{n^R}(x) = \mathcal{N}\left(\sum_{j=1}^{m} \mu_{a^R_j}, \sum_{j=1}^{m} \sigma^2_{a^R_j}\right)$$
$$f_{sp^{n^A}}(x) = \mathcal{N}\left(0, \sum_{j=1}^{m} \sigma^2_{j,n^{sp^A}}\right)$$
$$f_{sp^{n^R}}(x) = \mathcal{N}\left(0, \sum_{j=1}^{m} \sigma^2_{j,n^{sp^R}}\right)$$
$$f_{ac^{n^A}}(x) = \mathcal{N}\left(0, \sum_{j=1}^{m} \sigma^2_{j,n^{ac^A}}\right)$$
$$f_{ac^{n^R}}(x) = \mathcal{N}\left(0, \sum_{j=1}^{m} \sigma^2_{j,n^{ac^R}}\right). \quad (27)$$

This allows us to define the second main contribution of this article.

Based on the properties of independent normal random variables described above, we propose that when one (or more) node $a_i \in \{DAS - DAS_{\text{noise}}\}$, for $i = n - m + 1, \ldots, n$, is "moving" (powered ON), the measured aggregate instantaneous consumption/position (both active/x and reactive/y) is

$$\begin{aligned} A(k) &= a_1^A(k) + a_2^A(k) \cdots + a_m^A(k) + a_{m+1}^A(k) \\ R(k) &= a_1^R(k) + a_2^R(k) \cdots + a_m^R(k) + a_{m+1}^R(k) \end{aligned} \tag{28}$$

where the first m terms refer to the BN appliances, and the $(m+1)$th term refers to $a_i = a_{m+1}$. The quantities and relative distributions on the left side of (28) are immediately measurable by a SM, and the distribution shape of the first m aggregate contributions is known from (27). Consequently, a method for dis-aggregating the energy/mobility distribution of the $(m+1)$th node is required.

Let us suppose that at time k_{m+i}, a generic node $a_{m+i} \in DAS \setminus DAS_{\text{noise}}$, for $i = 1, \ldots, n-m$, is powered ON/OFF. The transient dynamic is detected by $\Delta(k) > \theta_{Thr}$ (9) and also detected when finished after a delay δ_i^m. Therefore, at time $k_{m+i}+\delta_i^m$, a new stable state is reached, given by the movement (energy dynamics) of the $\{a_1, \ldots, a_m, a_{m+i}\}$ aggregate nodes. At time $k_{m+i} + \delta_{m+i}$, the following relations are valid:

$$\begin{aligned} A\left(k_{m+i} + \delta_i^m\right) &= n^A(k_m) \pm a_{m+i}^A\left(k_{m+i} + \delta_i^m\right) \\ R\left(k_{m+i} + \delta_i^m\right) &= n^R(k_m) \pm a_{m+i}^R\left(k_{m+i} + \delta_i^m\right) \\ sp^A\left(k_{m+i} + \delta_i^m\right) &= n^{sp^A}(k_m) \pm sp_{m+i}^A\left(k_{m+i} + \delta_i^m\right) \\ sp^R\left(k_{m+i} + \delta_i^m\right) &= n^{sp^R}(k_m) \pm sp_{m+i}^R\left(k_{m+i} + \delta_i^m\right) \\ ac^A\left(k_{m+i} + \delta_i^m\right) &= n^{ac^A}(k_m) \pm ac_{m+i}^A\left(k_{m+i} + \delta_i^m\right) \\ ac^R\left(k_{m+i} + \delta_i^m\right) &= n^{ac^R}(k_m) \pm ac_{m+i}^R\left(k_{m+i} + \delta_i^m\right) \end{aligned} \tag{29}$$

where the $+$ operation indicates a power ON event, and the $-$ sign indicates a power OFF event. From (29), it is *very simple* to statistically evaluate the added (subtracted) terms in the second members, given that the first members represent the new aggregate (position, speed, and acceleration) trend after the last event (i.e., the SM's current output), while the first terms in the second member represent the BN-related variables (position, speed, and acceleration) before the event (logged by the monitoring system). It must be noted that after the arrival/departure of a node (appliance) on/from the map, the new aggregate becomes the new BN for the next power ON/OFF event. The approach represented by (29) allows statistical characterization of a moving node, but a classification is still required once the added/subtracted term is statistically derived to recognize the particular appliance/node. To this aim, we exploit the same approach proposed in [11], where evaluated data is converted into an image format and then classified by a CNN which extracts the required image features. Specifically, we use the mobility to matrix encoding (MME) algorithm, whose details can be found in [11].

IV. Extensive Numerical Analysis

This section analyzes the proposed scheme. A domestic environment, where $n = 42$, is examined. Table II summarizes

Fig. 16. Shelly EM device used in our testbed.

Fig. 17. Shelly EM integration with the main control panel.

the DAS of the appliances, indicating *always ON* m devices that contribute to BN. The final two columns represent nominal (N) or SB power and maximum (MAX) active power, respectively. The SM is a Shelly EM [41] (a_6), linked wirelessly to the rest of the network while being wired to the general electricity meter. Fig. 16 shows the Shelly EM device while Fig. 17 shows its integration with the general control panel of the considered house.

Consumptions data has been retrieved and stored by a dedicated Python script, called future house energy management (FEHM), whose frontend and backend are shown in Figs. 18 and 19. The application has been based on the Kivy virtual environment [45].

FHEM can interact with Shelly by the HTTP protocol, periodically asking for the needed consumption data. It can also get data from the solar power inverter, but it is out of the scope of this work.

The experiment proceeded as follows: the quality of the CNN (the same as in [11]) was evaluated by powering a

TABLE II
COMPLETE LIST OF COMPLETE NODES/APPLIANCES IN THE DOMESTIC SCENARIO

Node	Name	Description	N/SB [W]	MAX [W]
		Always ON (no events possible)		
a_1	Notebook 1	Huawei Matebook D 14"	10	10
a_2	Notebook 2	Lenovo Thinkpad 14"	13	13
a_3	Shelly 1	Shelly 1 – WiFi Switch	4	4
a_4	Shelly 1	Shelly 1 – WiFi Switch	4	4
a_5	Aquarium	Wave 100l aquarium with pump	18	18
a_6	Shelly EM	Shelly EM Smart meter	4	4
a_7	TV amplifier	Meliconi 15dB 75Ohm TV Signal amplifier	8	8
a_8	IR Camera	Trust Recording IR IP camera	6	6
a_9	Intercom	Vimar Video Intercom audio/video	10	10
		Normally in STAND-BY mode (BN contributors)		
a_{10}	Led Bar	5m Led Bar –Cold white	<0.5	28
a_{11}	Heating Oven	REX Heating Grill Oven	<2	2170
a_{12}	Heating Pump	Beretta FE12 Heating Pump	<8	2865
a_{13}	Smart TV 1	Samsung 32" LED panel Smart TV	<1	280
a_{14}	Smart TV 2	LG 75" NanoCell IPS panel Smart TV	<3	200
a_{15}	Smart TV 3	Samsung 40" LED panel Smart TV	<1	250
a_{16}	RGB Bulb	WiFi RGB Bulb with remote control	<0.5	9
a_{17}	Microwave Oven	REX Microwave oven	<3	1700
a_{18}	HiFi System	Sony HiFi cassette – CD player	<2	130
a_{19}	Voice Control	Alexa Echo Dot 3rd gen.	<2	10
a_{20}	Refrigerator	LG WiFi Refrigerator	<12	85
		Normally OFF (explicitly powered ON, no STAND-BY mode)		
a_{21}	Lamp 1	Light LED lamp		19
a_{22}	Lamp 2	Light LED lamp		19
a_{23}	Lamp 3	Light lamp		40
a_{24}	Desktop	Asus and NVidia PC 16GB RAM		450
a_{25}	Lamp 4	Light LED lamp		19
a_{26}	Lamp 5	Light LED lamp		10
a_{27}	Lamp 6	Light LED lamp		19
a_{28}	Lamp 7	Light LED lamp		10
a_{29}	Lamp 8	Light LED lamp		19
a_{30}	Lamp 9	Light LED lamp		29
a_{31}	Led Strip	5m LED strip		30
a_{32}	Lamp 10	Light LED lamp		19
a_{33}	Dishwasher	REX Dishwasher		2000
a_{34}	Extractor Hood	Faber air extractor		400
a_{35}	Lamp 11	Light LED lamp		10
a_{36}	Washing Machine	REX Washing Machine		2000
a_{37}	Lamp 12	Incandescent lamp		80
a_{38}	Lamp 13	Incandescent lamp		80
a_{39}	Electric Door	Electric Garage Sectional Door		300
a_{40}	Hydraulic pump	External water pump		450
a_{41}	Hair dryer	Remington hair dryer		2100
a_{42}	Iron	Rowenta Iron		1800

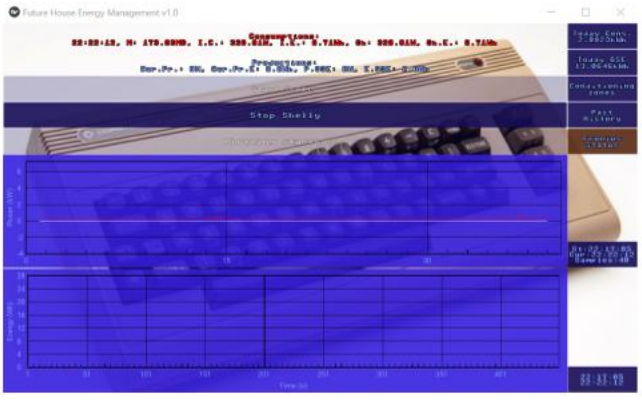

Fig. 18. Frontend of the FHEM application developed in Python on the Kivy virtual environment for implementing the DENARDO proposal.

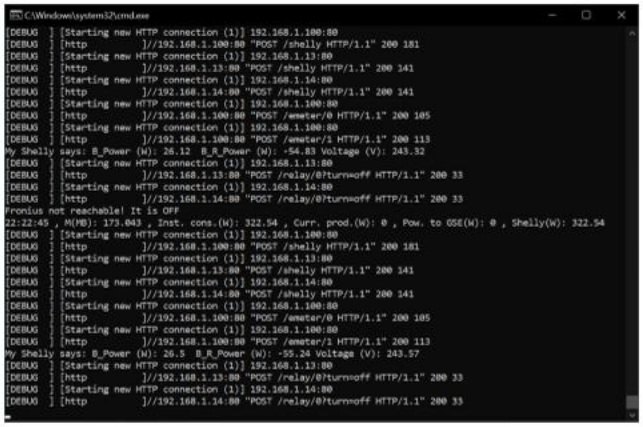

Fig. 19. Backend of the FHEM application developed in Python on the Kivy virtual environment for implementing the DENARDO proposal.

range of nodes/appliances ON and OFF, starting with the nine always-ON devices ($a_1, \ldots, a_9$) and then randomly selecting a greater number of contemporary nodes (12–42). Each power consumption trace was encoded into images by the *MME*, for different values of W. The CNN was then trained and validated.

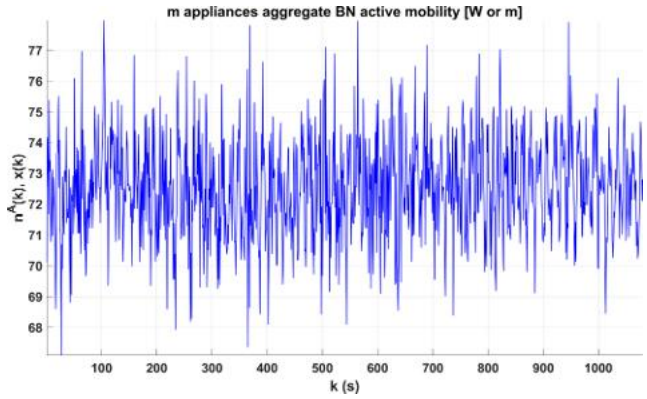

Fig. 20. $n^A(k)$ trend for the $m = 20$ devices $\in DAS_{\text{noise}}$.

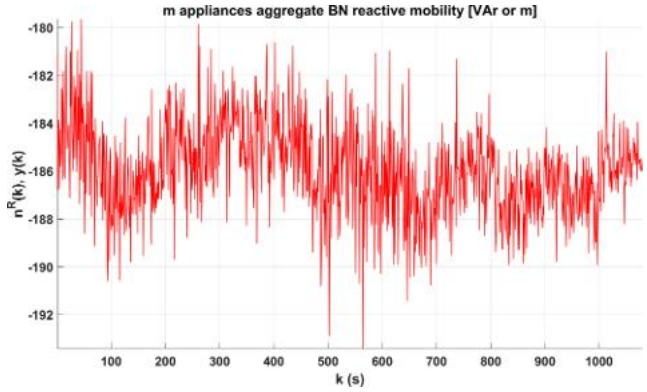

Fig. 21. $n^R(k)$ trend for the $m = 20$ devices $\in DAS_{\text{noise}}$.

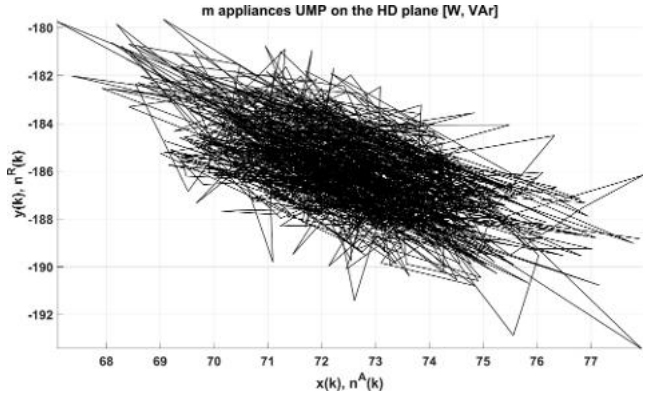

Fig. 22. UMP trajectory for the $m = 20$ devices $\in DAS_{\text{noise}}$.

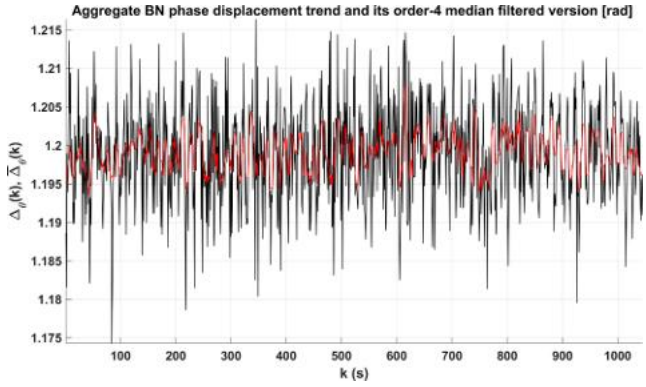

Fig. 23. $\Delta_\theta(k)$ (black) and $\overline{\Delta}_\theta(k)$ (red) trends for the $m = 20$ devices $\in DAS_{\text{noise}}$.

A. Background Noise Statistical Characterization

A first preliminary session of data collection was dedicated to DAS_{noise} *BN* analysis. Specifically, $m = 20$ appliances/nodes (nine always ON plus 11 in STAND-BY mode) were observed for 15 periods of 18 min (1080 sampling points, at $T = 1$ for each period, for a total of 270 min, i.e., 4.5 h) at night (starting from 1.30 A.M.). Each of the graphs presented below illustrates the results for one randomly selected period (no noticeable differences between the results for different periods was observed).

Figs. 20 and 21 indicate the active and reactive power consumption over the observed period (these can be also be seen as the x and y components of the aggregate node mobility). From the graphs, the trends described in the theory are not immediately and simply recognizable as power ON/OFF events, given their highly time-varying nature.

Fig. 22, however, instead illustrates how the concept of UMP can be immediately verified: appliances *move* on Hart's plane and create traces which represent the cluster of the *BN* process.

Fig. 23 shows the real aggregate phase displacement trend ($\Delta_\theta(k)$ in black) and its smoothed order-4 median filtered version ($\overline{\Delta}_\theta(k)$ in red). In the first case, the mean value is 1.1199 rad, with a variance of $3.8949 \cdot 10^{-5}$ rad^2 (with a minimum of 1.1742 rad and maximum of 1.2164 rad, and therefore a dispersion range of 0.0422 rad). The filtered signal instead shows a mean value of 1.1991 rad, a variance of $6.1641 \cdot 10^{-6}$ rad^2, with a minimum of 1.1934 rad and maximum of 1.2076 rad (dispersion range of 0.0142 rad). These values show that, by filtering the phase displacement, the mean value is not altered (the difference is only 10 μrad), and the variance and dispersion range are heavily reduced, giving the system a more stable representation of the aggregate signal and reducing the error probability in ON/OFF event recognition (as shown in the following results).

At this point, the *BN* process for the m DAS_{noise} devices can be fully statistically characterized, because its distributions, speeds and accelerations in terms of active and reactive components are easily evaluated. For readability purposes, in addition to the n^A and n_R processes, we show only detailed pdfs for *SP* and *AC* processes, given that their values follow the same trends illustrated in Fig. 14.

Fig. 24 plots the pdf trend for the n^A process, evaluated over 18 sampling periods. Its statistical parameters are $\sum_{j=1}^{m} \mu_{a_j^A} = 72.5173$ W (or m) and $\sum_{j=1}^{m} \sigma^2_{a_j^A} = 2.7262$ W^2. This means that 99% of the active power (or x position) values belong to the range $[\sum_{j=1}^{m} \mu_{a_j^A} - 3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{a_j^A}},\ \sum_{j=1}^{m} \mu_{a_j^A} + 3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{a_j^A}}]$, for $m = 20$, i.e., [67.5639, 77.4706] W (or m).

Fig. 25 plots the pdf trend for the n^R process, evaluated over 18 sampling periods. Its statistical parameters are $\sum_{j=1}^{m} \mu_{a_j^R} = -185.9467$ VAr (or m) and $\sum_{j=1}^{m} \sigma^2_{a_j^R} = 3.697$

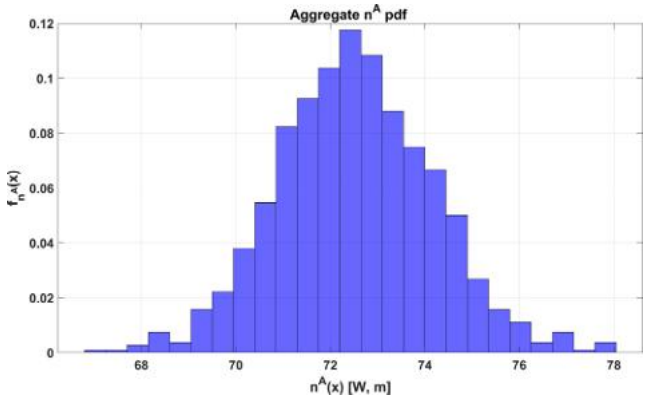

Fig. 24. *BN* position probability density function $f_{n^A}(x)$ for $m = 20$ devices $\in DAS_{\text{noise}}$.

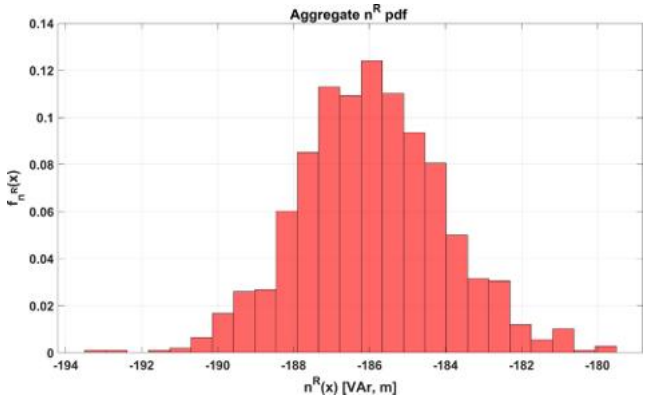

Fig. 25. *BN* position probability density function $f_{n^R}(x)$ for $m = 20$ devices $\in DAS_{\text{noise}}$.

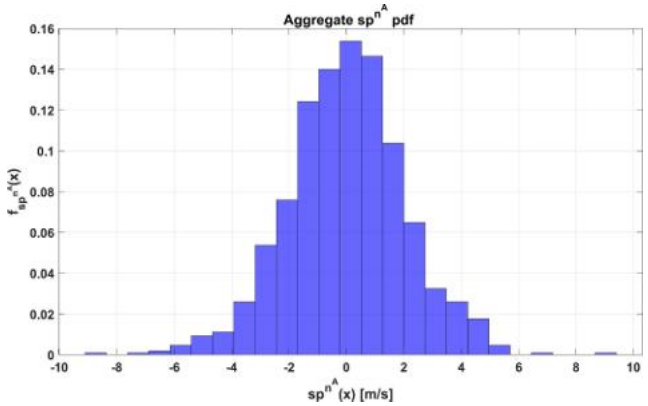

Fig. 26. *BN* speed probability density function $f_{sp^{n^A}}(x)$ for $m = 20$ devices $\in DAS_{\text{noise}}$.

VAr^2. This means that 99% of the reactive power (or y position) values belong to the range $[\sum_{j=1}^{m} \mu_{a_j^R} - 3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{a_j^R}},\ \sum_{j=1}^{m} \mu_{a_j^R} + 3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{a_j^R}}]$, for $m = 20$, i.e., $[-191.715, -180.1785]$ VAr (or m).

Fig. 26 plots the pdf trend for the sp^{n^A} process, evaluated over 18 sampling periods. Its unique statistical parameter is $\sum_{j=1}^{m} \sigma^2_{j,n^{sp^A}} = 4.1771$ m^2/s^2 (as expected, its mean value is

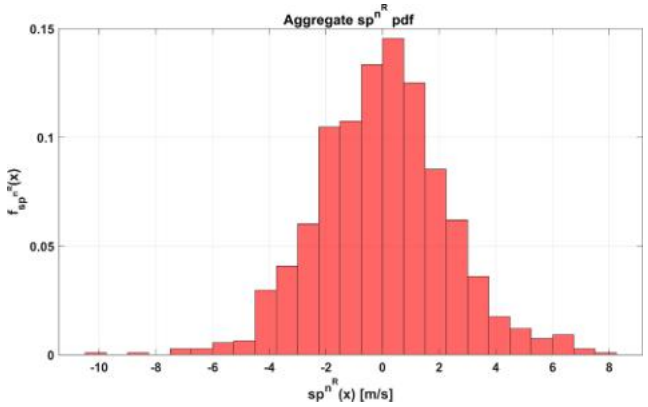

Fig. 27. *BN* speed probability density function $f_{sp^{n^R}}(x)$ for $m = 20$ devices $\in DAS_{\text{noise}}$.

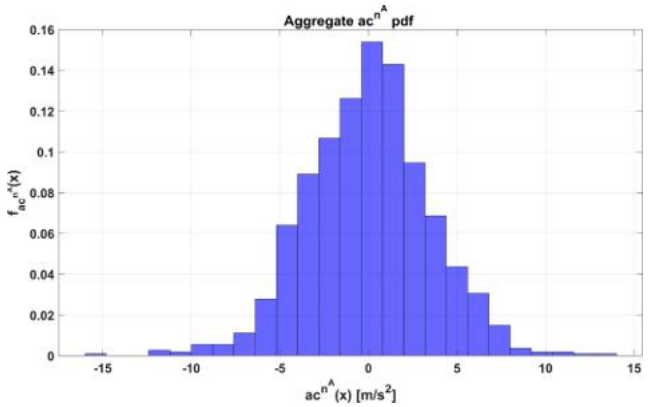

Fig. 28. *BN* Acceleration probability density function $f_{ac^{n^A}}(x)$ for the m=20 devices $\in DAS_{\text{noise}}$.

0.0025$\simeq$0). This means that 99% of the active speed values belong to the range $[-3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{j,n^{sp^A}}},\ 3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{j,n^{sp^A}}}]$, for $m = 20$, i.e., $[-6.1314, 6.1314]$ m/s.

Fig. 27 plots the pdf trend for the sp^{n^R} process, evaluated over 18 sampling periods. Its unique statistical parameter is $\sum_{j=1}^{m} \sigma^2_{j,n^{sp^R}} = 5.3613$ m^2/s^2 (as expected, its mean value is $-0.0045 \simeq 0$). This means that 99% of the active speed values belong to the range $[-3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{j,n^{sp^R}}},\ 3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{j,n^{sp^R}}}]$, for $m = 20$, i.e., $[-6.9464, 6.9464]$ m/s.

Fig. 28 plots the pdf trend for the ac^{n^A} process, evaluated over 18 sampling periods. Its unique statistical parameter is $\sum_{j=1}^{m} \sigma^2_{j,n^{acA}} = 12.18$ m^2/s^4 (as expected, its mean value is $-0.0031 \simeq 0$). This means that 99% of the active speed values belong to the range $[-3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{j,n^{acA}}},\ 3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{j,n^{acA}}}]$, for $m = 20$, i.e., $[-10.4657, 10.4657]$ m/s^2.

Fig. 29 plots the pdf trend for the ac^{n^R} process, evaluated over 18 sampling periods. Its unique statistical parameter is $\sum_{j=1}^{m} \sigma^2_{j,n^{acR}} = 16.0991$ m^2/s^4 (as expected, its mean value is $0.0053 \simeq 0$). This means that 99% of the active speed values belong to the range $[-3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{j,n^{acR}}},\ 3 \cdot \sqrt{\sum_{j=1}^{m} \sigma^2_{j,n^{acR}}}]$, for $m = 20$, i.e., $[-12.0371, 12.0371]$ m/s^2.

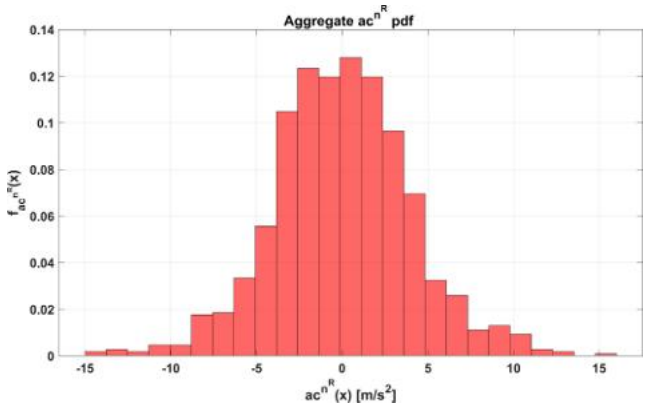

Fig. 29. *BN* acceleration probability density function $f_{ac^{nR}}(x)$ for the m=20 devices $\in DAS_{\text{noise}}$.

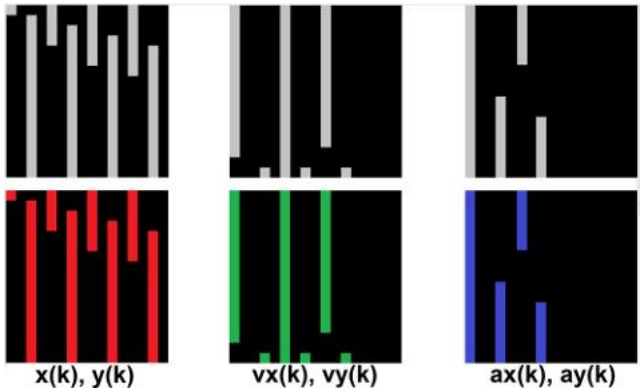

Fig. 30. Main process of data-to-image conversion and the creation of artificial features.

B. DENARDO Data to Image Conversion

This section references the work in [11] which demonstrates the process of classifying mobility data across a geographical area by converting the data into images. The concept is presented here only in its fundamentals. For more details, refer to [11] and the related data set [42]. As proposed in the current study, by considering appliance power consumption as an UMP, it is possible to compare the aggregate behavior of an appliance to the movement of a mobile node in a geographical area. As a new node joins (powered ON) or leaves (powered OFF) the aggregate, the mobility behavior changes drastically. A set of consumption trace-files (mobility data) permits conversion into images (not simply by converting an array into a matrix, but by finding a suitable data encoding method to create features within images for a CNN to recognize and thus classify the image [11]). Given the values for $n^A(k)$ [or $x(k)$] and $n^R(k)$ [or $y(k)$], it is possible to completely characterize the aggregate node position on Hart's plane. Similarly, given the values for $sp^{n^A}(k)$ [or $v_x(k)$] and $sp^{n^R}(k)$ [or $v_y(k)$], it is possible to completely characterize aggregate node speed during movement, and given ac^{n^A} and ac^{n^R}, also acceleration [$a_x(k)$ and $a_y(k)$, respectively]. Defining a spreading factor sf, mobility data can then be encoded into images, as illustrated in Fig. 30.

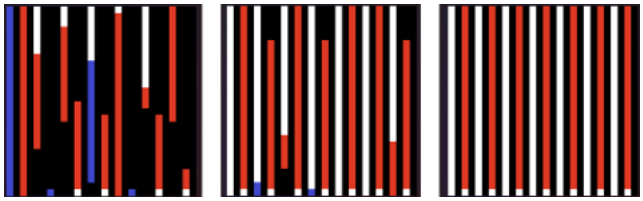

Fig. 31. Example of three 28×28 real *BN* images encoded by the MME algorithm, where $sf = 4$ and $u = 7$.

To summarize, mobility data (position, speed, and acceleration) sampled at T seconds is collected every $WIN{=}u \cdot T$ seconds (where $u \in \mathcal{Z}^+$). The images are square, with dimensions $u \cdot sf$. The red channel is reserved for position values, the green and the blue channels are reserved for speed and acceleration, respectively. For each channel, the mobility sample is normalized between 0 and $u \cdot sf$ and rendered into the image as consecutive pixels (from top to bottom for the x coordinate, from bottom to top for the y coordinate). The sf indicates the change in shifting the x and y columns: this should be at least equal to 2, and if $sf = 4$ for example, it means that the first column is dedicated to the x component, followed by a blank (black) column, then the third column dedicated to the y component, and again a black column. This structure is repeated u times to obtain $u{\cdot}sf$ columns. In Fig. 30, the upper square images represent the MME encoding mentioned above for the three channels, and the bottom images are their color versions. The final image combines the three channels (Fig. 31). For more details, refer to the MME algorithm proposed in [11].

Once a set of images (one every *WIN* seconds) is created, it is processed by a trained and validated CNN. The CNN's layering reflects an extension of classic LeCun's [9] structure. Layering the CNN is beyond the scope of the current study, therefore we incorporated the methods and results from our previous research in [11].

C. DENARDO Event Recognition

The proposed algorithm first required training and tuning to the environment. To this aim, we determined the optimal set of $\{W, \theta_{Thr}\}$ for which the probability of incorrectly recognizing an ON/OFF event (false positive) was below 0.01, referring to (9). Specifically, we trained the CNN on 20 sequences of 1500 s (T = 1 s, i.e., 25 min for each sequence for a total of more than 8 h), during which 35 of the n = 42 appliances were used normally (nine devices always remained ON, as illustrated in Table II, after the photovoltaic system shut down).

Fig. 32 shows the average heat map as a function of θ_{Thr} on the x-axis and W on the y-axis. The standard *hsv* MATLAB color map was customized to increase the contrast between dark-blue color (optimal) and other colors. The colored chart, where the optimal value is 0 (dark-blue), represents the average difference between the real number of ON/OFF events and the events recognized by DENARDO. The$x$-axis (θ_{Thr}) values are scaled by 10^{-4}.

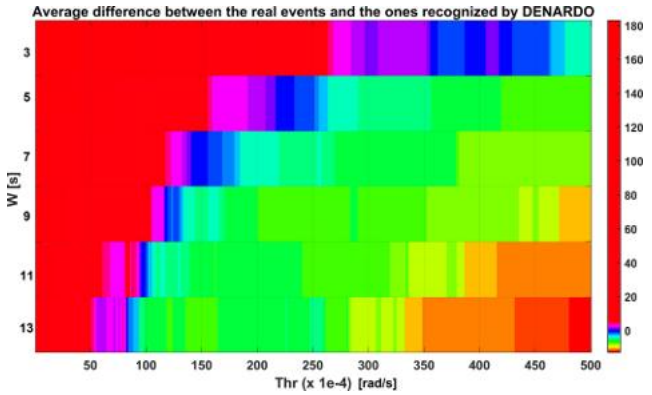

Fig. 32. Admissible regions for the DENARDO algorithm as a function of W and θ_{Thr} in (9).

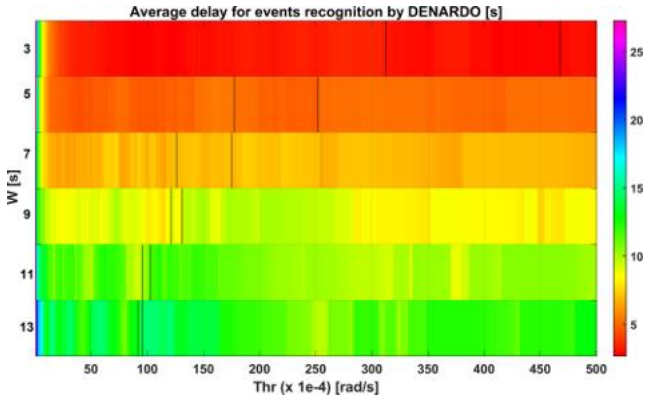

Fig. 33. Recognition delay of the DENARDO algorithm as a function of W and θ_{Thr}; the black vertical segments bound the chosen θ_{Thr} ranges for each value of W.

It can be observed that as the values of W increase, the admissible threshold range shrinks and decreases: for $W = 3$, we can choose $\theta_{Thr} \in [318e-4,\ 465e-4]$, for $W = 5$ we can choose $\theta_{Thr} \in [181e-4,\ 258e-4]$, for $W = 7$ we can choose $\theta_{Thr} \in [124e-4,\ 176e-4]$, for $W = 9$ we can choose $\theta_{Thr} \in [118e-4,\ 135e-4]$, for $W = 11$ we can choose $\theta_{Thr} \in [92e-4,\ 105e-4]$, and for $W = 13$ we can choose $\theta_{Thr} \in [83e-4,\ 90e-4]$ ($W = 1$ is meaningless, and higher values of W do not easily attain a recognition error probability below 0.01).

Execution of DENARDO on the 20 sequences and a selection of different window sizes W provided the results to evaluate the average delay in recognizing each ON/OFF event (i.e., the time measured from the moment of the ON/OFF event to its recognition). Fig. 33 depicts the delay trend, with the optimal threshold ranges highlighted for a precise view of the areas of delay for evaluation. It can be seen that within each identified range for a fixed W, the delay is almost constant. For example, at $W = 3$ the event is recognized in approximately 3.7 s, at $W = 5$ the delay is about 5.43 s, at $W = 7$ the delay is about 7.81 s, at $W = 9$ the delay is about 9.2 s, at $W = 11$ the delay is about 11.27 s, and at $W = 13$ the delay is 13.36 s.

The results illustrated in Figs. 32 and 33 suggest that W should be kept at a value between 3 and 7, given that the recognition accuracy does not change (it is compensated for by θ_{Thr}) and that delays greater than 8 s may not be useful if, for example, a domestic automation system requires activation according to user behavior. It should be noted that the long delay times in the example here relate strictly to the low sampling frequency of 1 Hz (inexpensive energy meters which use a sampling frequency of 10 Hz and obtain delays below 1 s are readily available on the market), which is the reason for considering domestic environments. For industrial environments, this approach can be easily adapted by obtaining a higher granularity in the collected samples.

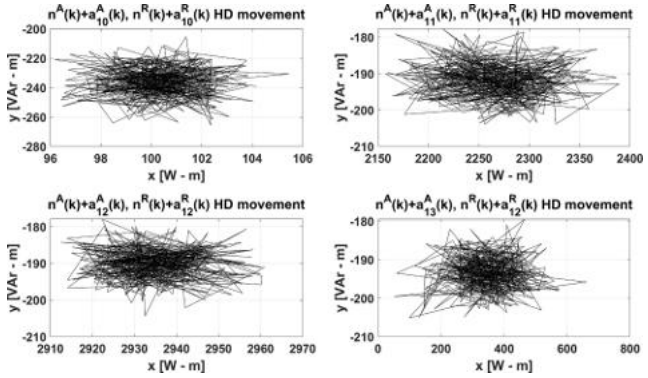

Fig. 34. Aggregate UMP position traces in the HD plane for BN and appliances a_{10}, a_{11}, a_{12}, and a_{13}, referring to the terms in (28) and (29).

D. DENARDO Appliance Classification

This section presents the core results of the simulations obtained from the DENARDO algorithm using the parameter pairs $W = 3$ and $\theta_{Thr} = 4e\text{-}2$, $W = 5$ and $\theta_{Thr} = 2e\text{-}2$, and $W = 7$ and $\theta_{Thr} = 1.5e\text{-}2$ (for notation simplicity, we refer only to the window size W; for the values of θ_{Thr}, refer to the previous section).

Fig. 34 depicts the clustering effects on the HD plane when different appliances are powered ON, calculated from (28) and (29). A comparison of these results with the results in Fig. 22 indicates that BN movements are shifted both horizontally (on the right since the active power consumption can only increase with respect to BN) and vertically in a different manner, according to Hart's theory [16]. The study's key innovation is the DENARDO algorithm's ability to determine each appliance's contribution to BN over time. As already discussed in the theory, the algorithm analyzes not only the node's position in the HD plane but also it's movement within that plane (speed and acceleration).

Figs. 35 and 36 depict the scatter plots for aggregated speeds and accelerations, respectively, of the BN and appliances. As expected, the mean values are 0 (the scatter points fade to white the farther away they are from 0), and the variances are different, confirming (27). In statistical terms, Figs. 34–36 are the quantities on the left sides of (28) and (29) and represent the new BN components after new ON events of appliances a_{10}, a_{11}, a_{12}, and a_{13} occur.

At this point, using the properties of characteristic functions of normal distributions (described in Section III-C) and because the BN before the events is known, DENARDO statistically disaggregates the current power consumption of

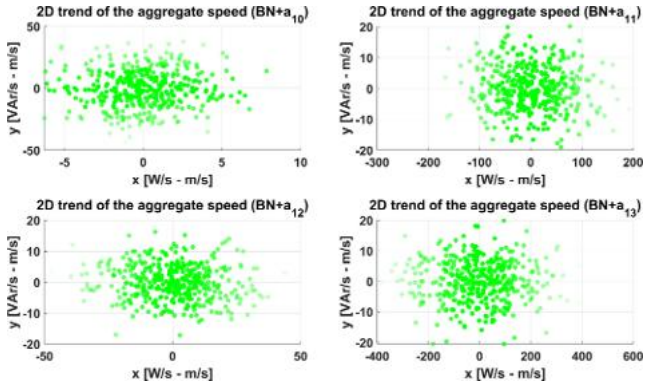

Fig. 35. Aggregate UMP speed traces for BN and appliances a_{10}, a_{11}, a_{12}, and a_{13}, referring to the terms in (28) and (29).

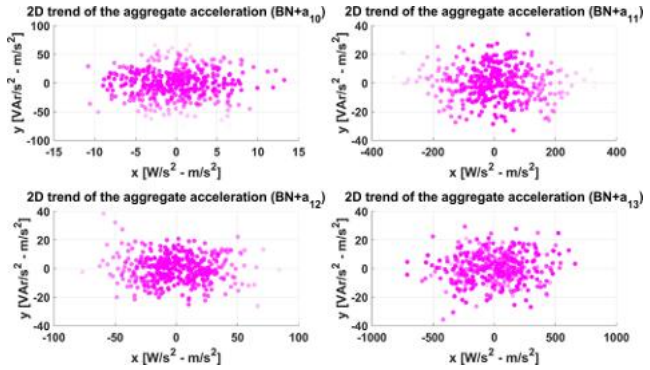

Fig. 36. Aggregate UMP acceleration traces for BN and appliances a_{10}, a_{11}, a_{12}, and a_{13}, referring to the terms in (28) and (29).

each appliance. Therefore, the algorithm calculates [from (29)]

$$\begin{aligned}
a^A_{m+i}\left(k_{m+i}+\delta^m_i\right) &= A\left(k_{m+i}+\delta^m_i\right) \pm n^A(k_m)\\
a^R_{m+i}\left(k_{m+i}+\delta^m_i\right) &= R\left(k_{m+i}+\delta^m_i\right) \pm n^R(k_m)\\
sp^A_{m+i}\left(k_{m+i}+\delta^m_i\right) &= sp^A\left(k_{m+i}+\delta^m_i\right) \pm n^{sp^A}(k_m)\\
sp^R_{m+i}\left(k_{m+i}+\delta^m_i\right) &= sp^R\left(k_{m+i}+\delta^m_i\right) \pm n^{sp^R}(k_m)\\
ac^A_{m+i}\left(k_{m+i}+\delta^m_i\right) &= ac^A\left(k_{m+i}+\delta^m_i\right) \pm n^{ac^A}(k_m)\\
ac^R_{m+i}\left(k_{m+i}+\delta^m_i\right) &= ac^R\left(k_{m+i}+\delta^m_i\right) \pm n^{ac^R}(k_m) \qquad (30)
\end{aligned}$$

where the negative $(-)$ operator refers to ON events and the positive $(+)$ operator refers to OFF events. To perform the operations in (30), if DENARDO recognizes an ON/OFF event, it evaluates the normal distribution parameters of the quantities on the right side every W seconds. The left side is obtained from the properties of normal distributions and the sum/difference between two normally distributed processes, as described in Section III-C.

As a final discussion, we evaluate the accuracies for different scenarios and compare the results with the performance achieved in several other works in the literature. First, we examined nine nodes/appliances comprising the static BN. The DENARDO algorithm was then applied to different scenarios, each time with an increase in the number of powered ON/OFF devices of 3 (i.e., 12, 15, 18, 21, . . . , 42; nodes were selected randomly in each step). Images were then created and their

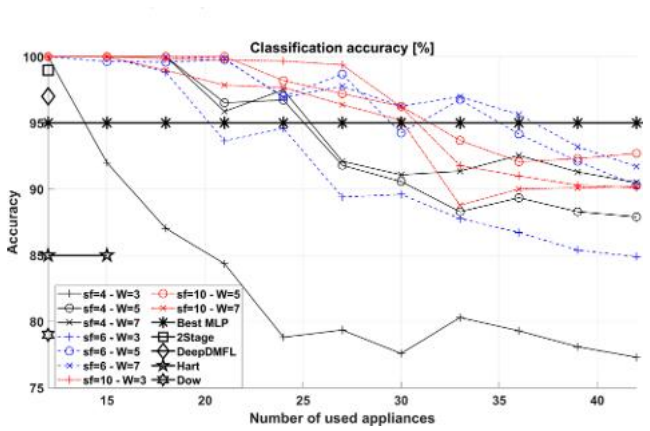

Fig. 37. DENARDO classification accuracy as a function of the number of powered ON/OFF devices, W, and sf.

dimensions (u and sf) varied to observe the effect on recognition accuracy. Fig. 37 shows the results for classification accuracy: thin lines refer to DENARDO; thick lines refer to other proposals in the literature; solid black lines refer to $sf = 4$ (image space occupancy of 98–150 B for $W = 3$, 122–168 B for $W = 5$, and 149–198 B for $W = 7$); dashed blue lines refer to $sf = 6$ (image space occupancy of 106–147 B for $W = 3$, 162–209 B for $W = 5$, and 203–256 B for $W = 7$); dotted/dashed red lines refer to $sf = 10$ (images space occupancy of 146–181 B for $W =$, 202–256 B for $W = 5$, and 281–325 B for $W = 7$). The lowest value (77.4%) and trend is reached in the first case ($sf = 4$ and $W = 3$) because the image size is too small (the side is only 12 pixels) and thus the CNN did not have sufficient data to extract the features as required.

In the other cases, except where $sf = 6$ and $W = 3$, accuracy exceeded 90% for up to 30 contemporary appliances. Generally, $W = 3$ is not suitable for low sf values. Increasing sf improves classification accuracy enormously, as indicated by the red dashed/dotted lines. Although $W = 3$, the images have enough space (in terms of pixels) to store the required features. Up to 27 contemporary nodes, accuracy was greater than 99.1%. A careful analysis of the classification accuracy curves shows reveals that the best parameters to set for the domestic scenario here are $W = 5$ and $sf = 10$, up to a maximum of 30 contemporary nodes. Another good choice is $W = 5$ and $sf = 6$: at these settings, DENARDO performed slightly worse than with $sf = 10$, but produced better performance on average, up to 39 contemporary nodes. All the images created for training and validation are available from [42]. To compare with the literature, we refer to the works which studied *Best MLP* [6], *Two-Stage Detection* [18], *Deep DMFL* [5], *Hart* [16], and *Dowalla* [17]. The results for Best MLP indicate good accuracy (best case of 95%), but a dedicated, advanced, and expensive measurement node is required, and the usable flash memory has some limitations. The number of appliances is also unspecified (because the proposed algorithm is based on appliance classification during switching between ON/OFF or OFF/ON). Therefore, we assume the best case also applies to the number of appliances included here. At up to 30–33 nodes, DENARDO outperforms

Best MLP. Two-stage detection applies to only 12 devices (as indicated in this article), performing with an accuracy of 98.9%, without good scalability. In this case, DENARDO returns an accuracy of 100%, with an incomparable level of scalability. Deep DMFL applies to only eight random devices (over the 27 offered by their considered data set), achieving an accuracy of 97% with a considerable sampling period (in the order of milliseconds). Also, in this case, DENARDO's performance is superior for some combinations of W and sf. DENARDO outperforms Hart and Dowalla's proposals in terms of both scalability and accuracy. Our created data set is available at [44]. It has a size of 115 MB and contains more than 325 000 images created by our approach, categorized in 597 folders, in function of the parameters of the encoding algorithm. It is available under request to the authors.

V. Conclusion and Future Work

The current study proposed a novel approach for NILM and appliance classification in a domestic IoT environment. It differs from other works by using inexpensive hardware and noncomplex ML approaches and introduces a completely new algorithm which considers appliances as mobile nodes and encodes their traces into images. The core concept is based on an approach which statistically models appliance trajectories and linearizes the classification process. The analysis also examined real device behavior (not synthetic behavior or behavior extracted from remote databases). In conclusion, the proposed algorithm outperforms the methods proposed in related works in the literature.

In future research, we plan several extensions to DENARDO, first by enhancing the data-to-image encoding algorithm (MME) to include more features in the output images (we have already commenced our investigation into image entropy levels needed by CNNs to learn image content more efficiently). A critical analysis of the obtained results lets us conclude with confidence that the algorithm's main parameters should be dynamic (not set a priori) and adapt themselves to the number and type of powered ON/OFF devices.

References

[1] A. Janik, A. Ryszko, and M. Szafraniec, "Scientific landscape of smart and sustainable cities literature: A bibliometric analysis," *Sustainability*, vol. 12, no. 3, p. 779, 2020, doi: 10.3390/su12030779.

[2] L. M. S. Campoverde, M. Tropea, and F. De Rango, "An IoT based smart irrigation management system using reinforcement learning modeled through a Markov decision process," in *Proc. IEEE/ACM 25th Int. Symp. Distrib. Simul. Real Time Appl.*, 2022, pp. 1–4, doi: 10.1109/DS-RT52167.2021.9576130.

[3] Y. Gao, S. Li, Y. Xiao, W. Dong, M. Fairbank, and B. Lu, "An iterative optimization and learning-based IoT system for energy management of connected buildings," *IEEE Internet Things J.*, vol. 9, no. 21, pp. 21246–21259, Nov. 2022, doi: 10.1109/JIOT.2022.3176306.

[4] L. Wang, S. Mao, and R. M. Nelms, "Transformer for nonintrusive load monitoring: Complexity reduction and transferability," *IEEE Internet Things J.*, vol. 9, no. 19, pp. 18987–18997, Oct. 2022, doi: 10.1109/JIOT.2022.3163347.

[5] L. D. S. Nolasco, A. E. Lazzaretti, and B. M. Mulinari, "DeepDFML-NILM: A new CNN-based architecture for detection, feature extraction and multi-label classification in NILM signals," *IEEE Sensors J.*, vol. 22, no. 1, pp. 501–509, Jan. 2022, doi: 10.1109/JSEN.2021.3127322.

[6] E. Tabanelli, D. Brunelli, A. Acquaviva, and L. Benini, "Trimming feature extraction and inference for MCU-based edge NILM: A systematic approach," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 943–952, Feb. 2022, doi: 10.1109/TII.2021.3078186.

[7] A. L. Davis, T. Krishnamurti, B. Fischhoff, and W. B. De Bruin, "Setting a standard for electricity pilot studies," *Energy Policy*, vol. 62, pp. 401–409, Nov. 2013, doi: 10.1016/j.enpol.2013.07.093.

[8] K.C. Armel, A. Gupta, G. Shrimali, and A. Albert, "Is disaggregation the holy grail of energy efficiency? The case of electricity," *Energy Policy*, vol. 52, pp. 213–234, Jan. 2013, doi: 10.1016/j.enpol.2012.08.062.

[9] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: 10.1109/5.726791.

[10] E. Chen, X. Wu, C. Wang, and Y. Du, "Application of improved convolutional neural network in image classification," in *Proc. Int. Conf. Mach. Learn., Big Data Busin. Intell. (MLBDBI)*, 2019, pp. 109–113, doi: 10.1109/MLBDBI48998.2019.00027.

[11] P. Fazio, M. Mehic, and M. Voznak, "A novel urban mobility classification approach based on convolutional neural networks and mobility-to-image encoding," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 35, no. 6, Jun. 2023, Art. no. 101561, doi: 10.1016/j.jksuci.2023.101561.

[12] T. Martin, S. Areibi, and G. Grewal, "Effective machine-learning models for predicting routability during FPGA placement," in *Proc. ACM/IEEE 3rd Workshop Mach. Learn. CAD (MLCAD)*, 2021, pp. 1–6, doi: 10.1109/MLCAD52597.2021.9531243.

[13] A. Singh, S. Sharma, and A. Gumaste, "Using deep reinforcement learning for routing in IP networks," in *Proc. Int. Conf. Comput. Commun. Netw. (ICCCN)*, 2021, pp. 1–9, doi: 10.1109/ICCCN52240.2021.9522197.

[14] R. Xu et al., "DarkTE: Towards dark traffic engineering in data center networks with ensemble learning," in *Proc. IEEE 29th Int. Symp. Qual. Service (IWQOS)*, 2021, pp. 1–10, doi: 10.1109/IWQOS52092.2021.9521298.

[15] S. Khan, H. Rahmani, S. A. Ali Shah, M. Bennamoun, G. Medioni, and S. Dickinson, *A Guide to Convolutional Neural Networks for Computer Vision*., San Rafael, CA, USA: Morgan Claypool, 2018.

[16] G. W. Hart, "Nonintrusive appliance load monitoring," *Proc. IEEE*, vol. 80, no. 12, pp. 1870–1891, Dec. 1992, doi: 10.1109/5.192069.

[17] K. Dowalla, P. Bilski, R. Lukaszewski, A. Wójcik, and R. Kowalik, "NILM application for real time monitoring of appliances energy consumption used," in *Proc. 11th IEEE Int. Conf. Intell. Data Acquis. Adv. Comput. Syst. Technol. Appl. (IDAACS)*, Cracow, Poland, 2021, pp. 1022–1025, doi: 10.1109/IDAACS53288.2021.9660389.

[18] W. Luan, Z. Liu, B. Liu, Y. Yu, and Y. Hou, "An adaptive two-stage load event detection method for nonintrusive load monitoring," *IEEE Trans. Instrum. Meas.*, vol. 71, Dec. 3, 2022, Art. no. 2504014, doi: 10.1109/TIM.2021.3132370.

[19] Y.-H. Lin and M.-S. Tsai, "Development of an improved time–frequency analysis-based nonintrusive load monitor for load demand identification," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 6, pp. 1470–1483, Jun. 2014, doi: 10.1109/TIM.2013.2289700.

[20] G. A. Raiker et al., "Energy disaggregation using energy demand model and IoT-based control," *IEEE Trans. Ind. Appl.*, vol. 57, no. 2, pp. 1746–1754, Mar./Apr. 2021, doi: 10.1109/TIA.2020.3047016.

[21] "A Dataset of voltage and current waveforms on a variety of single and multiple loads," LIT-DATASET. Accessed: Jan. 7, 2023. [Online]. Available: https://pessoal.dainf.ct.utfpr.edu.br/douglasrenaux/LIT_Dataset/

[22] C. Luo, X. Li, L. Wang, J. He, D. Li, and J. Zhou, "How does the data set affect CNN-based Image classification performance?" in *Proc. 5th Int. Conf. Syst. Informat. (ICSAI)*, 2018, pp. 361–366, doi: 10.1109/ICSAI.2018.8599448.

[23] A. Krizhevsky, "The CIFAR-10 and CIFAR-100 datasets," cs.toronto. Accessed: Feb. 12, 2023. [Online]. Available: https://www.cs.toronto.edu/ kriz/cifar.html

[24] "Facial expression recognition dataset," paperswithcode. [Online]. Available: https://paperswithcode.com/dataset/fer2013

[25] "Dogs vs Cats dataset," tensorflow. [Online]. Available: https://www.tensorflow.org/datasets/catalog/cats_vs_dogs

[26] Y. Pei, Y. Huang, Q. Zou, X. Zhang, and S. Wang, "Effects of image degradation and degradation removal to CNN-based image classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 4, pp. 1239–1253, Apr. 2021, doi: 10.1109/TPAMI.2019.2950923.

[27] Y. Xu, W. Liu, and K. F. Kelly, "Compressed domain image classification using a dynamic-rate neural network," *IEEE Access*, vol. 8, pp. 217711–217722, 2020, doi: 10.1109/ACCESS.2020.3041807.
[28] V. Tiwari, C. Pandey, A. Dwivedi, and V. Yadav, "G Image classification using deep neural network," in *Proc. 2nd Int. Conf. Adv. Comput., Commun. Control Netw. (ICACCCN)*, 2020, pp. 730–733, doi: 10.1109/ICACCCN51052.2020.9362804.
[29] M. Gogebakan, "A novel approach for gaussian mixture model clustering based on soft computing method," *IEEE Access*, vol. 9, pp. 159987–160003, 2021, doi: 10.1109/ACCESS.2021.3130066.
[30] T. Calinski and J. Harabasz, "A dendrite method for cluster analysis," *Commun. Stat.*, vol. 3, no. 1, pp. 1–27, 1974, doi: 10.1080/03610927408827101.
[31] J.-S. Wang and J.-C. Chiang, "A cluster validity measure with outlier detection for support vector clustering," *IEEE Trans. Syst., Man, Cybern., Part B (Cybernetics)*, vol. 38, no. 1, pp. 78–89, Feb. 2008, doi: 10.1109/TSMCB.2007.908862.
[32] J. S. Witte and R. S. Witte, *Statistics*, 11th ed. Hoboken, NJ, USA: Wiley, 2017, p. 48.
[33] W. Shaw and J. McCabe, "Monte carlo sampling given a characteristic function: Quantile mechanics in momentum space," in *Proc. Comput. Financ. (q-fin.CP)*, 2009, pp. 1–19. [Online]. Available: https://doi.org/10.48550/arXiv.0903.1592
[34] G. George, R. M. Oommen, S. Shelly, S. S. Philipose, and A. M. Varghese, "A Survey on various median filtering techniques for removal of impulse noise from digital image," in *Proc. Conf. Emerg. Devices Smart Syst. (ICEDSS)*, 2018, pp. 235–238, doi: 10.1109/ICEDSS.2018.8544273.
[35] C. Pasquini, G. Boato, N. Alajlan, and F. G. B. De Natale, "A deterministic approach to detect median filtering in 1D data," *IEEE Trans. Inf. Forensics Security*, vol. 11, pp. 1425–1437, 2016, doi: 10.1109/TIFS.2016.2530636.
[36] H. Alyammahi and P. Liatsis, "Non-intrusive appliance identification using machine learning and time-domain features," in *Proc. 29th Int. Conf. Syst., Signals Image Process. (IWSSIP)*, 2022, pp. 1–5, doi: 10.1109/IWSSIP55020.2022.9854459.
[37] Y. Liu and M. Chen, "A review of nonintrusive load monitoring and its application in commercial building," in *Proc. 4th Annu. IEEE Int. Conf. Cyber Technol. Autom., Control Intell.*, 2014, pp. 623–629, doi: 10.1109/CYBER.2014.6917536.
[38] H.-H. Chang, K.-L. Lian, Y.-C. Su, and W.-J. Lee, "Power-spectrum-based wavelet transform for nonintrusive demand monitoring and load identification," *IEEE Trans. Ind. Appl.*, vol. 50, no. 3, pp. 2081–2089, May/Jun. 2014, doi: 10.1109/TIA.2013.2283318.
[39] R. Killick, P. Fearnhead, and I. A. Eckley, "Optimal detection of changepoints with a linear computational cost," *J. Am. Stat. Assoc.*, vol. 107, no. 500, pp. 1590–1598, 2012, doi: 10.1080/01621459.2012.737745.
[40] M. Cardone, A. Dytso, and C. Rush, "Entropic central limit theorem for order statistics," *IEEE Trans. Inf. Theory*, vol. 69, no. 4, pp. 2193–2205, Apr. 2023, doi: 10.1109/TIT.2022.3219344.
[41] "Shelly Italia." shellyitalia. Accessed: Nov. 1, 2023. [Online]. Available: https://www.shellyitalia.com/shelly-em/
[42] P. Fazio, "Mobility to image dataset," peppinofazio. Accessed: Oct. 10, 2023. [Online]. Available: peppinofazio.com/datasets
[43] S. A. Nene, S. K. Nayar, and H. Mur, "Columbia object image library COIL-100," kaggle. [Online]. Available: https://www.kaggle.com/jessicali9530/coil100
[44] P. Fazio, "Energy to image dataset 2023," peppinofazio. Accessed: Oct. 10, 2023. [Online]. Available: peppinofazio.com/datasets
[45] Digital Ocean, *The Open Source Python App Development Framework*. kivy. Accessed: Aug. 12, 2023. [Online]. Available: https://kivy.org/index.html

Peppino Fazio (Member, IEEE) received the master's degree in computer science and the Ph.D. degree in electronics and telecommunications engineering from the University of Calabria, Cosenza, Italy, in May 2004 and January 2008, respectively.

In June 2004, he was with the Telecommunications Group, University of Calabria. From May 2008 to September 2008, he was a Visiting Researcher with the GRC Research Group, UPV of Valencia, Valencia, Spain. He is an Assistant Professor with the Ca' Foscari University (DSMN), Venice, Italy, and a Senior Researcher with the VSB—TUO, Ostrava, Czechia. He has authored and coauthored more than 130 papers for international journals and conference proceedings. His research interests include IP QoS architectures, adaptive wireless networks, mobile ad hoc networks, quantum key distribution networks, and vehicular ad hoc networks.

Dr. Fazio is the recipient of international awards, for example, the Intel Business Challenge Award in 2013. He has been a member of various startup companies.

Miralem Mehic (Senior Member, IEEE) received the Ph.D. degree in telecommunications from the VSB—Technical University of Ostrava, Ostrava, Czechia, in 2017.

Also, he studied at the AGH University of Science and Technology, Krakow, Poland, Alpen-Adria-Universitat Klagenfurt, Klagenfurt, Austria, and Austrian Institute of Technology (AIT), Department of Digital Safety & Security Business Units—Optical Quantum Technology, Vienna and Klagenfurt, Austria. He is the National Principal Investigator of EU H2020 OPENQKD and the NATO SPS G5894 QUANTUM5 projects. He has authored the unique QKD network simulator QKDNETSIM. His field of research is related to the quality of service and management of QKD networks with a focus on real-time traffic and the utilization of network resources.

Miroslav Voznak (Senior Member, IEEE) received the Ph.D. degree in telecommunications from the Faculty of Electrical Engineering and Computer Science, VSB—Technical University of Ostrava (VSB-TUO), Ostrava, Czechia, in 2002, and the Habilitation degree from the Department of Telecommunications, VSB-TUO in 2009.

In 2017, he was appointed as a Full Professor of Electronics and Communications Technologies. He has authored and coauthored over 100 articles indexed in SCI/SCIE journals. His research interests include information and communication technologies, quality of service and experience, network security, wireless networks, and big data analytics.

Prof. Voznak is one of the World's Top 2% of Scientists in Networking and Telecommunications and Information and Communications Technologies, according to Stanford University study released in 2020.

Open Access funding provided by 'Università Ca' Foscari Venezia' within the CRUI CARE Agreement