

Article

A Multi-Agent NILM Architecture for Event Detection and Load Classification

André Eugenio Lazzaretti ¹, Douglas Paulo Bertrand Renaux ^{1,*} and Carlos Raimundo Erig Lima ¹, Bruna Machado Mulinari ¹, Hellen Cristina Ancelmo ¹, Elder Oroski ¹, Fabiana Pöttker ¹, Robson Ribeiro Linhares ¹, Lucas da Silva Nolasco ¹, Lucas Tokarski Lima ¹, Júlio Shigeaki Omori ² and Rodrigo Braun dos Santos ²

- ¹ LIT—Laboratory of Innovation and Technology in Embedded Systems and Energy, Universidade Tecnológica Federal do Paraná—UTFPR, Sete de Setembro, 3165, Curitiba 80230-901, Brazil; lazzaretti@utfpr.edu.br (A.E.L.); erig@utfpr.edu.br (C.R.E.L.); brunamachadomulinari@gmail.com (B.M.M.); hellen@alunos.utfpr.edu.br (H.C.A.); oroski@utfpr.edu.br (E.O.); fpottker@utfpr.edu.br (F.P.); linhares@utfpr.edu.br (R.R.L.); lucasnolasco.5@gmail.com (L.d.S.N.); tokarski.lima@gmail.com (L.T.L.)
- ² COPEL—Companhia Paranaense de Energia, José Izidoro Biazetto, 158, Curitiba 82305-100, Brazil; julio.omori@copel.com (J.S.O.); rodrigo.braun@copel.com (R.B.d.S.)
- * Correspondence: douglasrenaux@utfpr.edu.br; Tel.: +55-041-3310-4427

Received: 28 July 2020; Accepted: 23 August 2020; Published: 26 August 2020



Abstract: A multi-agent architecture for a Non-Intrusive Load Monitoring (NILM) solution is presented and evaluated. The underlying rationale for such an architecture is that each agent (load event detection, feature extraction, and classification) outperforms others of the same type in particular scenarios; hence, by combining the expertise of these agents, the system presents an improved performance. Known NILM algorithms, as well as new algorithms, proposed by the authors, were individually evaluated and compared. The proposed architecture considers a NILM system composed of Load Monitoring Modules (LMM) that report to a Center of Operations, required in larger facilities. For the purpose of evaluating and comparing performance, five load event detect agents, five feature extraction agents, and five classification agents were studied so that the best combinations of agents could be implemented in LMMs. To evaluate the proposed system, the COOLL and the LIT-Dataset were used. Performance improvements were detected in all scenarios, with power-ON and power-OFF detection improving up to 13%, while classification accuracy improved up to 9.4%.

Keywords: event detection; feature extraction; load classification; NILM; non-intrusive load monitoring; NILM architecture

1. Introduction

Non-Intrusive Load Monitoring (NILM) is a technique based on centralized measuring of electrical energy consumption and, by a disaggregation process, determining the individual consumption of each electrical load. NILM uses a database of known power signatures of devices to analyze the aggregated power consumption and identify individual loads' contributions. Therefore, NILM is a low cost, easily deployed, flexible and viable solution that provides consumers with detailed information about their energy consumption [1]. NILM provides essential information for use in Smart Grids, in Energy Management Systems and for Energy Efficiency Initiatives, justifying the current global research effort on the subject. NILM may well become a widespread diagnostic and energy management solution, in the context of Electrical Efficiency, available to every end-user. As a diagnostic tool, it identifies energy waste and improper use; for energy management, it may be used by residential users and by commercial/industrial users.

The proposed solution's physical organization may vary from a single unit, which would be the case for a typical residential consumer, to a multi-unit scenario, each reporting to a Center of Operations that consolidates the processed data from the individual units. Such an organization would be used in larger commercial, industrial, government, and services consumers. The proposed multi-agent (In this work, the term agent refers to each method used in the different stages of the proposed NILM system, namely: event detection, feature extraction, and classification) architecture presented here applies to each unit, named LMM (Load Monitoring Module). LMMs are edge computing devices whose purpose is voltage-current sensing, load disaggregation, and load event reporting.

In [2], a Multi-Agent System is defined as "a loosely coupled network of problem-solving entities (agents) that work together to find answers to problems that are beyond the individual capabilities or knowledge of each entity (agent)". In this sense, the proposed architecture has multiple load event detection agents, feature extraction agents, and load classification agents. Each of these agents works autonomously, examining the data collected from the voltage and current sensors. Their findings are then combined to obtain more accurate results than what is achieved by any individual agent.

Next, a contextualization of NILM systems in the area of Electrical Energy Efficiency is presented.

1.1. Contextualization

The worldwide electricity demand is currently 29,000 TWh and will increase to 42,000 TWh in 2040 at about 2.1% per year, according to the International Energy Agency [3]. Under current Stated Policies, less than 50% of this energy will come from renewable sources: mainly solar, wind, and hydro. Improvements of this scenario may be attainable by reducing the electrical energy needs, particularly by improving electrical devices' efficiency. Hence, the importance of Electrical Energy Efficiency: aiming at the reduction in power and energy demands of electrical systems without effect on their functionality. Improving energy efficiency does reduce the predicted growth rate in electricity demand and, most importantly, has a significant effect on the environment, reducing the world's carbon emissions.

To improve electrical energy efficiency, governments and associations have set policies and programs that drive the implementation of projects aiming at energy efficiency. The LEED initiative (Leadership in Energy and Environmental Design) certifies buildings as to several levels: certified, silver, gold, and platinum; one of the aspects of this certification being the building's energy efficiency.

Among the many objectives of Smart Grids are the improved efficiency, use of digital information and control, and access to detailed information about the energy usage by consumer appliances. NILM is, therefore, a technology that can be fully applied to the development of Smart Grids due to its capabilities of collecting energy usage data and communicating it upstream, allowing for better energy management decisions.

1.2. Rationale

To pursue the objective of Energy Efficiency, an individualized breakdown of current energy consumption is needed. Upon the availability of such a survey, electrical loads can be prioritized in terms of consumption and efficiency, providing input to a replacement plan of the most consuming and less efficient devices among those currently in use.

In Brazil, as in many other countries, government agencies and power utility companies are funding the replacement of inefficient devices and the construction of small power generation plants based on renewable energy sources, mainly photovoltaic panels, wind turbines, and small hydroelectric power plants, particularly those that are installed in the customer premises. The starting activity in such projects is a detailed analysis of current electrical energy usage, providing basic information for a technical and economic viability study. The International Performance Measurement and Verification Protocol [4] defines a process to determine the energy savings resulting from an efficiency project. The process is based on initial estimates of performance improvements (ex-ante) followed by Measurement and Verification (M&V) that provides measured data for the ex-post estimates in the M&V report.

Traditionally, M&V is performed by connecting measuring equipment, such as a power/energy meter, to several devices in the consumer premises, and perform measurements of energy and usage-over-time for a period long enough to achieve an accuracy of 10% with a 95% confidence level. A NILM system does provide a solution for such an analysis in a much faster, less costly, and over a more representative period than traditional methods based on individually measuring the consumption of samples of each load type.

To guide organizations pursuing improvements in energy efficiency, the International Organization for Standardization (ISO) published in 2011 the ISO 50,001, a standard for Energy Management Systems, currently in the 2018 edition [5]. The proposed management cycle consists of an initial planning and implementation, followed by a PDCA lifelong cycle of continuous improvement. Again, NILM provides a solution both for the data collection during the initial planning phase and for continuously monitoring of the organization's electrical energy usage providing the required data for the continuous improvement cycle.

Another application for NILM is as a tool to support energy management; such a tool may be used from small applications, such as remotely managing a beach house, to supporting large organizations with centralized operations management that can pinpoint the usage of each electrical device in its premises.

There are many other applications for NILM under evaluation, including the prediction of maintenance needs based on altered power signatures of devices as well as alarms for unusual energy consumption that could indicate security faults, accidents, or simply devices that were not powered-off appropriately.

1.3. Contributions

Our proposed structural organization consists of a Center of Operations that consolidates data from several LMM units. Typically, each LMM is installed in an electrical distribution sub-panel. In such a structural organization, an LMM must sense the voltage and aggregated current flowing into the distribution sub-panel, detect power change events, identify the load(s) causing such events, store a log, and, if a Center of Operations exists, upload the event log. Another function of the LMM is to learn the power signature of each load connected to this distribution sub-panel. As new loads are constantly added, this training process takes place continuously. In typical residential locations, a single LMM is used, and no Center of Operations is required. To operate independently, an LMM has a local user interface either on the device itself or via a wireless connection to a smartphone or notebook.

In such a scenario, a multi-agent architecture for the LMM units is proposed. This architecture is comprised of a set of load event detection agents that identify the instant when devices are powered on, or off, or when state changes occur. For every detected event, signature analysis agents are activated to extract features and classify loads based on a feature database populated during the training process. Every event is logged, indicating the time of occurrence, the corresponding load, and associated power level. Such an architecture provides better event detection rates and better load identification rates due to the combination of the "expertise" of the individual agents.

Several agents were implemented, and their performance compared, both in standalone mode and when combined with other agents. For adequate comparison of the agents, we defined a set of metrics for detection agents and another for signature analysis agents. While the algorithms used in some of the agents were already presented in the literature, three novel methods are proposed: detection based on half-cycle apparent power, signature analysis based on the power envelope during a power transition, and on a vectorization technique. Also, contributions concerning training include a technique where ML-based agents are trained on single loads and then operate on multiple loads. Two methods to combine the individual results of agents were proposed and evaluated: Weighted Combination and Dilation-Erosion.

1.4. Paper Organization

The remainder of this paper is organized as follows: Section 2 presents a review of the literature concerning the aspects discussed here. Also, previous papers by the authors are listed. Section 3 presents the proposed architecture for the LMM. The five load event detection agents are described: Discrete Wavelet Transform, Kalman Filter, High-Accuracy NILM Detector, Half-Cycle Apparent Power Analysis, and the Vectorization Technique. Also, the five feature extraction agents (V-I Trajectories, Wavelet, Prony, Power Envelope, and Vectorization) as well as the five load classification agents (k-NN, Decision Tree, SVM, Ensemble, and Linear Discriminant). This section also examines the combination of the individual results of these agents. Section 4 presents the datasets used during the evaluation of the proposed architecture: COOLL and LIT-Dataset. Furthermore, the metrics used to evaluate the detection and load identification agents are defined here, as well as the training procedure. In Section 5 the performance evaluation of each individual agent and the performance of the combined result are presented. The conclusions of this paper are then presented in Section 6.

2. Review of Techniques for Load Event Detection and Power Signature Recognition

Although the completely automatic power signature analysis is still an open problem in recent literature, the vast majority of high-frequency methods use a topology based on the following sequential stages [6]: (i) signal acquisition; (ii) data processing; (iii) event detection; (iv) feature extraction; (v) classification to perform load disaggregation; with the last three being the most relevant in terms of accuracy for NILM systems, as discussed next.

The event detection stage mainly comprises identifying the time segments in a waveform where power transients occur. As storing and processing all the captured data is neither efficient nor practical, due to the high sampling frequency (order of kHz), several methods are available to perform transient extraction from waveforms [7]. In general, these methods are grouped according to their nature and can be classified into: [8]: (i) changes in power or current; (ii) filtering; (iii) prominent residuals from models.

Changes in power or current are associated with signal analysis in time-domain, in which the event detection is based, for instance, on the apparent power to determine the instant of an event, as presented in the Half-Cycle Apparent Power (HCApP) method, proposed in [9]. A similar approach, using the envelope of electrical current and an adaptive threshold-based method, is presented in [10,11], defined as High-Accuracy NILM Detector (HAND). Furthermore, considering the need to detect signatures based on state change characteristics, in [12], an event detection is presented for multiple state loads, using the concept of powerlets [13], which were defined as groups of short sequences that characterize an appliance. Other methods can be highlighted, such as goodness-of-fit (GOF) [13], with similar detection accuracy for publicly available datasets. The performance of these methods is suitable for events where significant power variations occur in a cycle or semi cycle of the power network. However, for less significant power variations and for applications with higher resolution requirements, the performance of such methods can reduce the accuracy of the temporal location of events and, consequently, their proper detection.

A possible solution to improve detection accuracy and resolution is the use of Discrete Wavelet Transform (DWT). In [14], wavelet decomposition is applied to automatically establish the segmentation of waveforms. A universal threshold calculation procedure is used in the sequence, followed by a smoothing filter. This ensures that the entire detection process is performed without the need to define empirical thresholds. Similarly, in [15], the start and end times are determined using values from wavelet levels in combination with a non-steady state frequency band of Short-Time Fourier Transform, mainly for turn-off events. With that, detection errors in the order of 5 ms can be obtained for both simulated and real data. Another approach to accurate event detection is the use of prominent residuals from models, such as Kalman Filter [8]. Such filters use the innovation signal as a reference for highlighting switching transients related to turn-on and turn-off events, with small temporal errors (around 1 ms). Even with smaller temporal differences in the location of events

compared to RMS methods, a common limitation of DWT and Kalman filters, as discussed in [16], is their relatively low performance for certain types of turn-off events, in addition to a high rate of false positives, i.e., events falsely indicated as turn-on or turn-off.

After event detection, the next stage is the extraction of features that will be used to characterize the loads. These extraction methods usually are divided into steady-state, transient, or a combination of these two approaches [17].

The steady-state methods extract characteristics from the permanent regime segments of a waveform. Examples of this approach are power variation [18], current harmonics [19], current waveforms [20], and Voltage-Current (V-I) trajectories [21]. Generally, these methods are less susceptible to noise and present load recognition accuracies above 90%, even when classifying multiple loads, as discussed in [21]. In [9,16], the HCApP can also be used to extract features based on active, reactive, and apparent power of a signal to perform classification, even though some features can be extracted during transients. Still in this context, a Long-Short-Term Memory Recurrent Neural Network (LSTM-RNN) model is proposed in [22], to directly disaggregate load power signals. Additionally, the authors present an improvement of the proposed model for multistate appliances, which is derived from features learned with the LSTM-RNN, improving results for UK-DALE and REDD datasets. A limitation of steady-state methods is the lack of information extracted during load switching instants, such as inrush currents from electrical motors.

Transient methods of feature extraction, on the other hand, use characteristics observed during the power-transient segments of a waveform, such as their shape, size, duration, and envelope. Some examples of transient methods are the Wavelet transform [15,23] and the S-transform [24]. The use of the transient segments for certain types of electrical loads, such as motors, may also improve the classification and identification, compared to steady-state methods since the transient segment can carry more discriminative information [1]. In [25], four new wavelets, based on a wavelet design procedure, have been presented and provided the best matching to the analyzed load signals, when compared to standard wavelets, in a dataset with up to four loads. Similarly, in [26], wavelet transforms are combined with the technique of “delay coordinate embedding” to extract features that are more suitable for multi-label classification, i.e., classification in the presence of multiple simultaneous loads. However, using only transient information may result in relevant loss of information, such as the rated load power.

To combine the advantages of steady-state and transient feature extraction methods, other approaches combine features from both the steady-state and the transient segments of the waveforms. In [27], for instance, new features in the steady-state, in addition to a new set of characteristics in the transient state from the mutual locus of instantaneous voltage and current waveforms, defined as V-I trajectory, are presented. By combining and selecting a proper set of features, results above 98% of accuracy are obtained, improving the performance of steady-state V-I features presented in [21].

In [10], a modification of the Fourier Transform, incorporating an exponential damping, is proposed. The authors demonstrate that the identification of the damping observed during the transients improves the classification performance when combined with Fourier coefficients, achieving an accuracy of 90–99% for the COOLL dataset, even for unsupervised approaches. However, the implementation of this technique may present an instability during the exponential estimation [28] that causes an imprecise estimation of the harmonic contents. An analytic solution, based on the Prony’s Method, to the problem of estimating the harmonic content with an exponential damping factor, presented in [28], improves the accuracy of the estimation of frequency, phase, amplitude, and damping factor for current signals.

The last stage of the process is load disaggregation, where loads are individually identified using pattern classification. There are different techniques to distinguish among classes of loads, and the main focus of these techniques lies in the two different approaches that can be employed [29]: supervised and unsupervised training. In the case of the unsupervised approach, the objective is to group similar patterns based on the characteristics extracted from a previously unknown data set,

associating similar signatures, and, subsequently, identifying each of the loads. On the other hand, supervised models consider a previously established database with a limited set of loads features, properly and previously labeled, and the classification is made within this context. The most common techniques are Hidden Markov Models (mainly for steady-state), k-nearest-neighbors, support vector machines, decision trees, and random forests [13]. The supervised training approach is usually applied since one seeks to focus on the loads that are expected to be recognized in a given system [26].

As can be observed, different methods present different advantages in terms of load detection and identification. In this sense, the main idea of this work is to present a combination of different state-of-the-art load event detection agents, feature extractors, and classifiers, extending our initial work [16], to improve the identification of different loads in a NILM system. To the extent of our knowledge, such combination is still unexplored in the NILM literature, although already used in other energy-related problems [30–32], and it can be seen as a significant advantage for NILM implementations.

2.1. Previous Publications

The methods and results hereby presented have been developed under an ongoing research and development project, funded by COPEL and ANEEL. Previous publications and patent requests, resulting from this research project, are listed in Table 1. The publications encompass the LIT-Dataset and its subsets, as well as power signature analysis methods, which are the basis of the agents and the multi-agent architecture presented here.

Table 1. Related publications from the current research project.

Reference	Year	Publication	Description
-	2017	INPI	Patent request for NILM device.
[33]	2018	PEAC	Initial proposal of HCApP and multi-agent architecture.
[34]	2018	EECS	Initial Prony-based method proposal.
[16]	2018	EECS	Improved Multi-agent architecture.
[35]	2018	SBESC	Synthetic subset proposal.
[36]	2019	ISAP	Simulated subset proposal.
[28]	2019	ISAP	Prony-based proposal and comparisons.
[27]	2019	ISGT-LA	Steady-state and transient V-I features proposal.
[37]	2019	SBESC	Natural subset proposal.

3. A Multi-Agent NILM Architecture

The proposed multi-agent architecture for a Load Monitoring Module (LMM) is depicted in Figure 1. Initially, voltage and current samples are collected and stored in a circular buffer. A zero-crossing detector is used to identify the first and last samples of each voltage semi cycle and informs the Event Detection Agents (EDAs) and Feature Extraction Agents (FEAs). Additionally, the active, reactive, and apparent power in each semi cycle are calculated from the voltage and current samples. A 21st order Gaussian filter is also available in the pre-processing stage, to attenuate noise components of the acquired signals and improve the detection accuracy, as detailed in [9].

Next, five Event Detection Agents, concurrently, monitor the incoming data to detect a load event: power-on, power-off, or state change. Their findings are reported to the Event Detection Composition block, whose decision of whether a load event occurred or not is based on a combination of the individual contributions of the EDAs. In this work, the following EDAs were selected: (i) Discrete Wavelet Transform (DWT) [14,15]; (ii) Kalman Filter (KF) [8]; (iii) Half-Cycle Apparent Power (HCApP) [9]; (iv) High-Accuracy NILM Detector (HAND) [10,11]; (v) Vectorization (VEC). The criterion for choosing the EDAs was to have at least one method representing each group (see Section 2): (i) changes in power or current (HCApP, HAND, and VEC), (ii) high-pass filters (DWT), (iii) and

prominent residuals from models (KF). In this manner, different approaches are compared as well as their possible combinations.

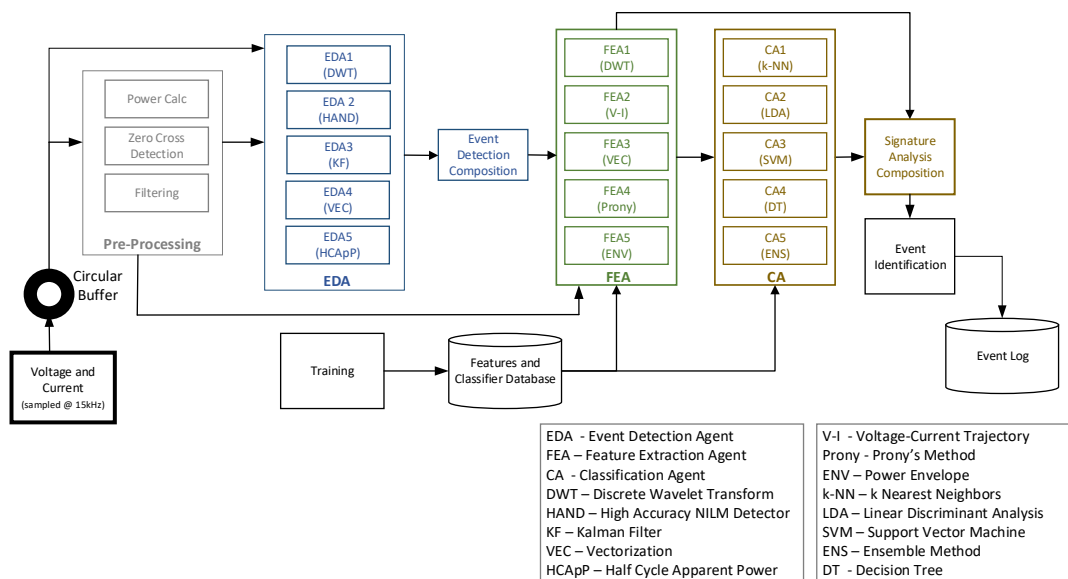


Figure 1. Functional multi-agent architecture.

Once a candidate event is detected, the FEAs examine the samples in the circular buffer to extract relevant transient and steady-state features. Five feature extractors, based on steady-state, transient and their combination, were selected: (i) V-I trajectories (V-I); (ii) Prony's Method; (iii) Energy Levels of Discrete Wavelet Transform; (iv) State information from Vectorization; and (v) Power Envelope method. The identification of loads can be accomplished using machine-learning-based classification agents (CA). Five machine learning methods were evaluated: (i) k-Nearest-Neighbors (k-NN); (ii) Linear Discriminant Analysis (LDA); (iii) Support Vector Machine (SVM); (iv) Decision Tree (DT); (v) Ensemble (ENS). These methods are all based on supervised learning, which means that all the class labels are known and previously defined during the training process. Hence, a database of load features and trained classifiers supports the identification process, as presented in Figure 1. Finally, individual results from the FEAs and the CAs are combined in the Signature Analysis Composition that produces the final load classification result. Each of these EDAs, FEAs, and CAs are detailed below.

3.1. Load Event Detection Agents

In this section, the five detection agents that compose the proposed architecture are presented: Discrete Wavelet Transform, Kalman Filter, High-Accuracy NILM Detector, Half-Cycle Apparent Power Analysis, and Vectorization Method.

3.1.1. Discrete Wavelet Transform

The main motivation for DWT analysis is the time-domain signal decomposition into frequency sub-bands, using orthonormal bases obtained from digital filter banks [14]. The signal is processed by a series of low and high-pass filters that separate frequency components in different subspaces. The construction of the filters is based on the wavelet function properties (mother-wavelet). Thus, orthonormal bases of discrete wavelet functions are associated with the mother Wavelet function, aiming at signal details (high-pass filters), as well as a scale function observing signal approximations (low-pass filters), forming an orthonormal basis. Different mother-wavelets can be used to analyze electrical signals, the most frequent are Daubechies and Symmlets [15].

The wavelet decomposition procedure presents good time-domain resolution, for high-frequency components, and good frequency-domain resolution, for low-frequency components [38]. In the transition segments, caused by events (ON or OFF), the coefficients along the different wavelet levels may show significant variations, corresponding to the frequency variation at a given time. Hence, the detection can be performed using a threshold comparison in each wavelet level, marking the significant transition moments in a waveform, as discussed in [15].

There are two approaches to threshold comparison: static (fixed level) and adaptive. In the static case, the processed signal (wavelet level) is compared with a given fixed value, as presented in Figure 2. As can be observed, the optimal threshold selection for the first event (ON) is inefficient for another (OFF).

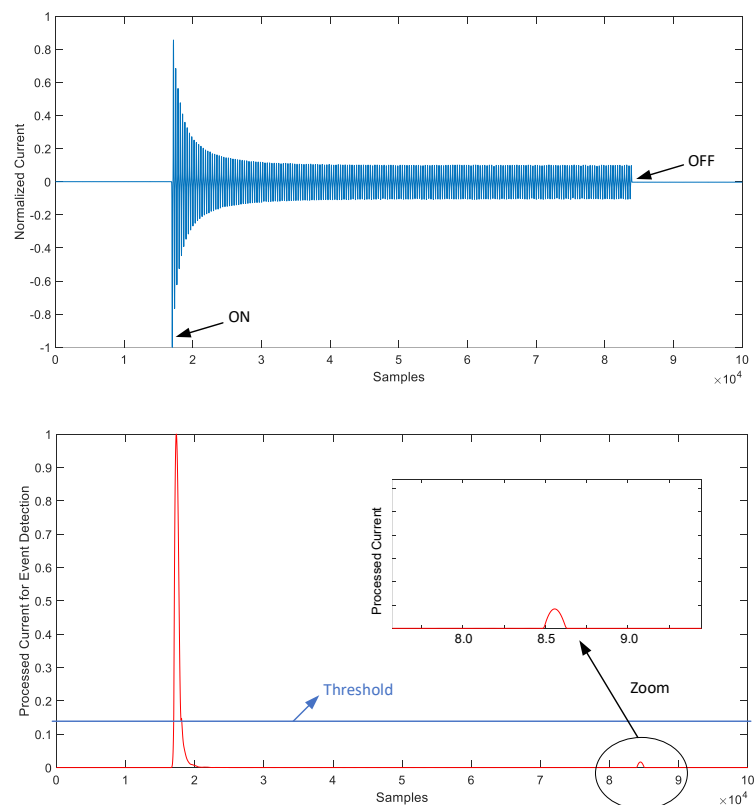


Figure 2. Detection based on a static threshold comparison procedure. The normalized current of an electric drill is presented on top with two events (ON and OFF). The processed current signal (wavelet level) is presented in the bottom part of the figure, indicating the limitations of a fixed threshold.

Diversely, in the adaptive case, the threshold is proportional to the dispersion of the signal amplitude window [16]. By analyzing if the amplitude of a sample is higher than the threshold value, points of interest, such as load insertion or removal, can be identified. The mean $\bar{x}[k]$ and standard deviation $\sigma[k]$ in their recursive forms are updated for every sample using the values of the estimated mean $\hat{x}[k-1]$ and estimated standard deviation $\hat{\sigma}[k-1]$ for the previous iteration, thus without the need of recalculating these values for the whole set of previous samples.

In Equation (1), a forget factor $0 \leq \lambda \leq 1$ is also implemented to insert a decay rate in the sample weight, provided that the mean $\bar{x}[k]$ and standard deviation $\sigma[k]$ equally weight the last samples and the previous samples in their recursive form.

$$\hat{x}[k] = \hat{x}[k-1] + \frac{1}{k}(x[k] - \lambda\hat{x}[k-1]), \quad (1)$$

$$\hat{\sigma}^2[k] = \frac{2\lambda-1}{\lambda} \hat{\sigma}^2[k-1] + (1-\lambda)[x(k) - \bar{x}(k)], \quad (2)$$

being $k \in \mathbb{Z}$ the number of the sample (iteration), and $\hat{x}[\cdot]$ and $\hat{\sigma}[\cdot]$ estimates of the real value of $x[\cdot]$ and $\sigma[\cdot]$, respectively. An event detection, $d[k]$, occurs when $|x[k]|$ overflows the threshold $\bar{x}[k] \pm \times \sqrt{\sigma^2[k]}$, i.e.,

$$\begin{cases} \text{if } |x[k]| > \max(|\bar{x}[k]| \pm \times \sqrt{\sigma^2[k]}), & d[k] = 1 \text{ (event detected);} \\ \text{otherwise,} & d[k] = 0 \text{ (no event detected),} \end{cases} \quad (3)$$

The selection of the value \times was done empirically, aiming at maximizing detection rate. Henceforth, wavelet levels can be analyzed using two threshold comparison approaches, extending the standard approach that is only based on a fixed level threshold, as presented in [14,15].

3.1.2. Kalman Filter

The Kalman Filter is characterized by optimal recursive estimation of the states and model parameters [8]. This stochastic filter alternates prediction and correction stages. For the prediction stage, the filter needs a dynamic model of the process, which in this case is based on the composition of a signal, composed of N harmonics, as presented in [8]. Hence, the Kalman filter estimates the parameters associated with these harmonics (magnitudes and phase angles) and reconstructs the signal with these estimated parameters. From the input samples and the reconstructed signal, the residue can be estimated. In the perspective of event detection, when a load event occurs, there are prominent values in the residue signal, indicating transients due to the error between estimated and actual signals. Such differences occur because the Kalman filter model only takes into account a given number of harmonics, and transient segments normally contain high-frequency content, differing from the filter estimates, but emphasizing the detection of events. In the Kalman filter, it is also possible to use static and adaptive thresholds for the residue signal, as previously presented in Equations (1)–(3).

3.1.3. High-Accuracy NILM Detector

The HAND (High-Accuracy NILM Detector) approach, presented in [10,11], tracks the variation of the standard deviation of the current signal envelope using a moving window. The peak value of the current in each mains cycle is extracted; then, a cubic interpolation among all the peaks is performed. As the interpolation causes the smoothing of the abrupt changes of the current signal, the authors proposed a second-level threshold comparison based on the median of the absolute value of the current signal. Therefore, the envelope presents a better characterization of the variations during state transitions in the current signal.

Based on this processed envelope, the recursive mean and standard deviation are estimated. Then, an adaptive threshold-based approach classifies transient and steady-state segments using the amplitude variation of the standard deviation. Next, post-processing is applied to correct signal sections barely exceeding the threshold (hysteresis) and the time-delay that appears in the stop time, due to the windowing characteristic of the proposed method. Finally, the algorithm outputs the starting and stopping times of each detected event. It is noteworthy that this method is intrinsically based on the adaptive threshold previously presented.

3.1.4. Half-Cycle Apparent Power Analysis

This technique is based on the calculation of the aggregate active, reactive, and apparent power in each semi-cycle of the power network. The analysis over a semi-cycle is a unique contribution of this technique, allowing the proper treatment of asymmetrical loads, such as those loads controlled by power electronics devices.

The HCApP (Half-Cycle Apparent Power), initially presented in [9], is a windowing technique that segments the apparent power waveform into “stable” and “transient” sections of varying lengths. Apparent power, active power, reactive power, as well as the asymmetry among positive and negative half-cycles are used to aid this segmentation process.

To illustrate the HCApP method, Figure 3 presents load 37 of the COOLL dataset: a vacuum cleaner. The horizontal axis has a duration of 6 seconds (the scale used is in semi cycles of 50 Hz mains). Apparent power is presented in red, active power in yellow and reactive power in purple. The analysis performed with the HCApP method indicates the periods where power is changing, shown in light blue. The first power transient period corresponds to power-on, and the second power transient period corresponds to power-off. The two spikes in the light blue line indicate the power-on and power-off events, respectively. These are the load events reported by this EDA.

The power transient windows are constructed using a sliding window technique that detects power changes along this window. Once a power change is detected, a second phase of this analysis identifies the width of the power transient. Next, an analysis of power triangle components (apparent power, active power, and reactive power) results in the detection of the load event.

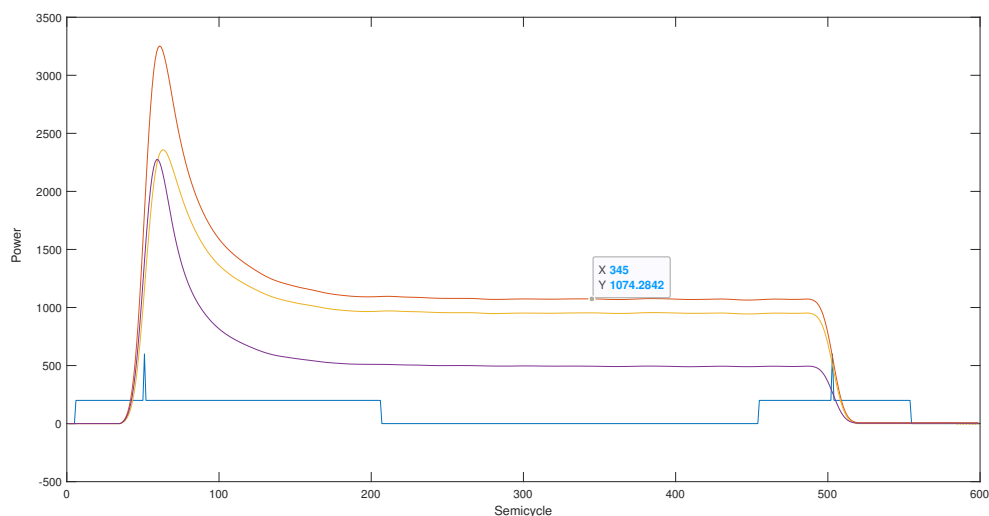


Figure 3. The power profile of a vacuum cleaner (COOLL load 37). Apparent power (VA) in red, active power (W) in yellow, and reactive power (VAR) in purple. The light blue line shows stable power periods (0), power transient periods (200), and load events (600).

3.1.5. Vectorization

By extracting characteristics from voltage and current signals, in successive cycles of the power network, it is possible to obtain dynamic curves of those characteristics [12]. The Vectorization algorithm aims to generate simplified mathematical representations (vectors) of curves with complex dynamic and static behavior, with a low computational cost. This method generates a simplified vector representation of any curve related to power systems, such as apparent power or active power. Furthermore, it can be used for both for event detection and for load classification, as will be presented in Section 5.

The principle of the Vectorization algorithm is to group successive samples into valid and non-valid states. The characterization of a valid state considers two borderline parameters: the minimum number of samples needed to characterize a valid state (n_{min}) and the limit amplitude value (v_{lim}) in relation to the average value of previous samples in the same state, as depicted in Figure 4, for tuning parameters $n_{min} = 5$ and v_{lim} delimited by the color dashed boxes. If a set of successive samples does not meet these two parameters, then such a set is tagged as an invalid state. An invalid state may provide useful information with respect to the nature of the observed signal, such as abrupt transitions of the signal, typical of transients with high derivatives or segments with high-frequency oscillations. The parameters (n_{min}) and (v_{lim}) can be adjusted, automatically, for each segment analyzed.

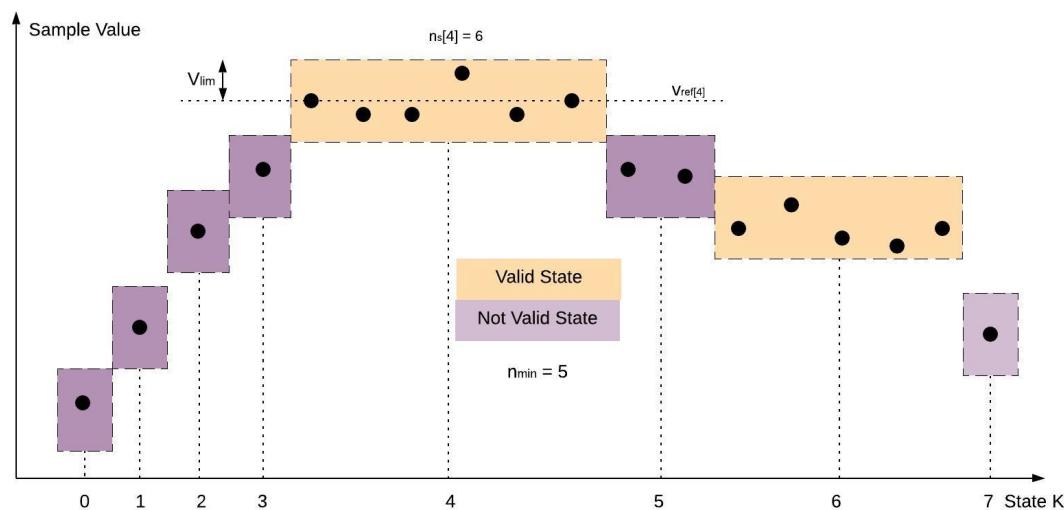


Figure 4. A sequence of invalid states and valid states, for a minimum number of samples per state greater than 5 ($n \geq 5$) and samples restrict to limit value (v_{lim}).

Considering a discrete signal to be analyzed, represented by the vector $v_{in}[n]$, with n samples, $n_s[k]$ being the number of samples in state k , and $v_{ref}[k]$ the average value of the consecutive samples in state k , it is defined that:

- A sample $v_{in}[j]$, within a state k , meets the criterion of the limit value v_{lim} , if $|v_{in}[j] - v_{ref}[k]| < v_{lim}$.
- A state is classified as a valid state if the following conditions are met simultaneously: (a) All $n_s[k]$ samples of a state meet the criterion of the limit value v_{lim} ; (b) $n_s[k] \geq n_{min}$.
- A state is classified as a non-valid state if $n_s[k] < n_{min}$.

Thus, a discrete signal $v_{in}[n]$ can be represented by a vector $[n_s[0], v_{ref}[0] \quad n_s[1], v_{ref}[1] \dots, n_s[k-1], v_{ref}[k-1] \quad n_s[k], v_{ref}[k]]$ of k states, called a states vector. Alternatively, the states vector can be represented by $[n_s[0] + jv_{ref}[0], \quad n_s[1] + jv_{ref}[1], \dots, n_s[k-1] + jv_{ref}[k-1], \quad n_s[k], +jv_{ref}[k]]$; such a representation is used as a feature for load classification (Section 3.3.5).

A signal can thus be represented by a sequential set of valid and invalid states, in which each state is represented by the number $n[k]$ of samples in that state (for valid states, $n[k]$ is equal or greater than n_{lim} , for invalid states, $n[k]$ is less than n_{lim}) and the average value $v_{ref}[k]$ of the samples within a state. To indicate that a state is invalid, negative values of $n[k]$ are used. Hence, the state representation of the samples in Figure 4 is: $[-1 + jv_{ref}[0] \quad -1 + jv_{ref}[1] \quad -1 + jv_{ref}[2] \quad -1 + jv_{ref}[3] \quad 6 + jv_{ref}[4] \quad -2 + jv_{ref}[5] \quad 5 + jv_{ref}[6] \quad -1 + jv_{ref}[7]]$.

Figure 5a represents a sample window of the RMS power (W) of a device. The states vector representation of the samples window, considering tuning parameters $n_{min} = 1000$ and $v_{lim} = 30$ W is

shown in Figure 5b. For illustration purposes, the non-valid states are represented by hatched areas in blue.

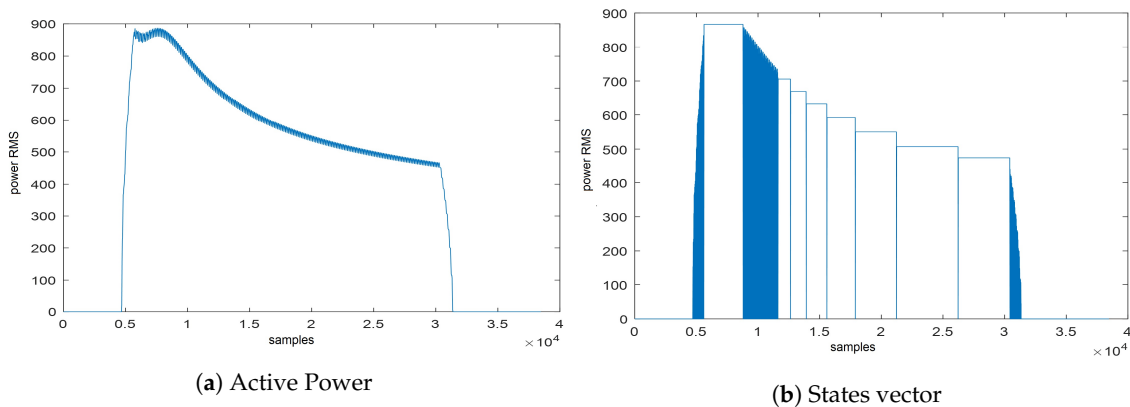


Figure 5. (a) Active Power (W) waveform. (b) States vector (valid states and not-valid states) representation of the waveform in (a) for $n_{min} = 1000$ samples and $v_{lim} = 30$ W.

3.2. Combination of Detection Agents

Each detection agent, to the best of its abilities, identifies load events and informs such occurrences to the “Event Detection Composition” block. As well as indicating that a load event occurred, each detection agent also indicates the level of certainty it has on this indication.

The Event Detection Composition (EDC) block combines the individual detections of the EDAs to produce a single output that identifies the time instances of a high likelihood of a load event. Two approaches to Event Detection Composition were implemented. The first proposed method, referred to as Weighted Combination, is based on a weighted sum of three values that characterize the individual results of the detection agents. The second proposed method, referred to as Dilation and Erosion (DE), is based on an image processing technique of dilation and erosion.

The principle of operation of the Weighted Combination is that of weighted voting among the individual EDAs, in which the reported likelihood is used as the weight of the vote, as described by the equation below:

$$pd_{ji} = \gamma_1 \left(\frac{1}{n_{ag}} \sum_{l=1}^{n_{ag}} I(Aout_{jl} = out_i) \right) + \gamma_2 \left(\frac{1}{n_{cor}} \sum_{l=1}^{n_{ag}} I(Aout_{jl} = out_i) ac_{il} \right) + \gamma_3 \left(\frac{1}{n_{ag}} \sum_{l=1}^{n_{ag}} I(Aout_{jl} = out_i) sd_{il} \right), \quad (4)$$

in which $Aout$ is the output of each agent for each semi cycle $\{Aout \in \mathbb{Z}^{n_{sc} \times n_{ag}} \mid Aout_{jl} \in \{-1, 0, 1\}\}$, sd is the event detection likelihood (degree of confidence) of each agent in each semi cycle $\{sd \in \mathbb{R}^{n_{sc} \times n_{ag}} \mid 0 \leq sd_{il} \leq 1\}$, and ac , defined as $\{ac \in \mathbb{R}^{n_{labels} \times n_{ag}} \mid 0 \leq ac_{il} \leq 1\}$, is the performance of each agent for each one of the three event labels (n_{labels}): ON (in our case, +1), OFF (in our case, -1) and steady-state (0), which is not associated with load switching events. $I(x)$ is an operator that returns 1 if the condition in x is true and zero otherwise, while n_{sc} is the total number of semi cycles, n_{ag} is the total number of agents, $n_{cor} = \sum_{l=1}^{n_{ag}} I(Aout_{jl} = out_i)$, and $out = [-1, 0, 1]$. Finally, pd is a matrix with the number of rows equals to the number of samples (in semi cycles) to be detected, and the number of columns equals the number of classes (three, in this case), i.e., $\{pd \in \mathbb{R}^{n_{sc} \times n_{labels}} \mid 0 \leq pd_{ij} \leq 1\}$. The selected final class (ON, OFF or steady-state) corresponds to the column with the highest pd_{ji} value.

Each portion of the equation takes into account a weight (γ_1 , γ_2 , and γ_3 subject to $\gamma_1 + \gamma_2 + \gamma_3 = 1$), defined a priori and selected to maximize the final detection accuracy. Once these parameters are defined, Equation (4) is applied for $i = [1, 2, 3]$ and $j = [1, 2, 3, \dots, n_{sc}]$ to fully calculate pd .

The second proposed method, Dilation and Erosion, considers the fact that distinct EDAs may detect the same power event with slight time differences, yet, they are reporting the same event.

Hence, individual detections are first dilated, then combined using a weighted voting method, and, finally, the result is eroded to produce a sharp identification of the event occurrence in time. The weighted voting functionality considers the event type (power-on or power-off) and also particular combinations of votes. For instance, if HCApP and HAND vote simultaneously for a power-on event, then there is a high likelihood that such an event occurred, and the EDC (Event Detection Composition block) will report it.

Figure 6 shows a particularly difficult waveform to detect events due to the combination of high power and low power devices. Several false positive and false negative detections are shown. Figure “Eroded” in Figure 6 shows the final result of the EDC after dilation and erosion.

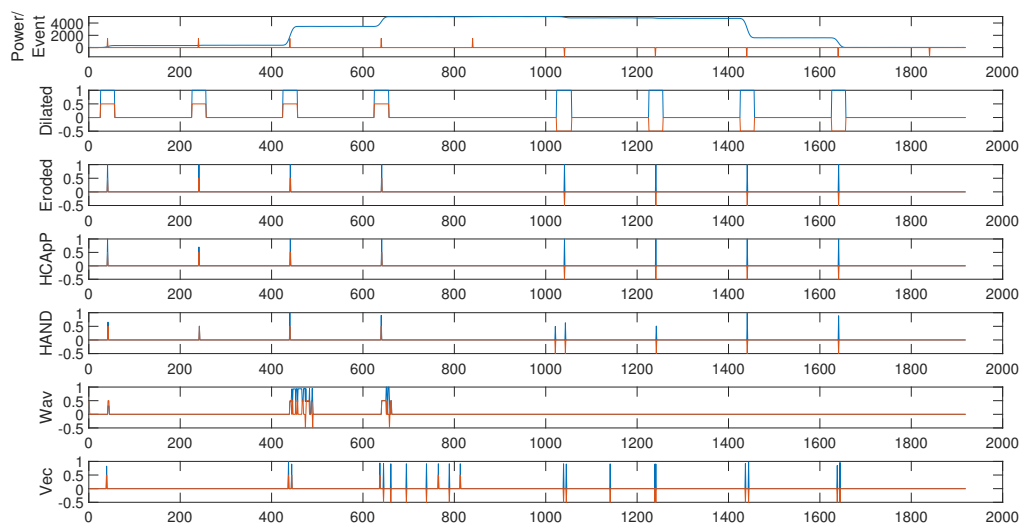


Figure 6. Overview of the Dilatation and Erosion method. The upper figure shows Power (VA) (blue) and events (red). All other figures show events in red (negative means power-OFF), and event probability in blue; in this order: combined events after dilation, combined events after erosion, detections by HCApP, HAND, Wavelet and Vectorization. Horizontal axis is time in seconds \times 120.

3.3. Feature Extraction Agents

The FEAs are based on steady-state and transient features extracted from voltage and current waveforms, as illustrated in Figure 7. The FEAs are signaled by the EDC of the occurrence of a load event. In the case of multiple loads, some FEAs perform the subtraction of the current previous to the event and the average of the voltage, as suggested in [21]. Each FEA is detailed as follows.

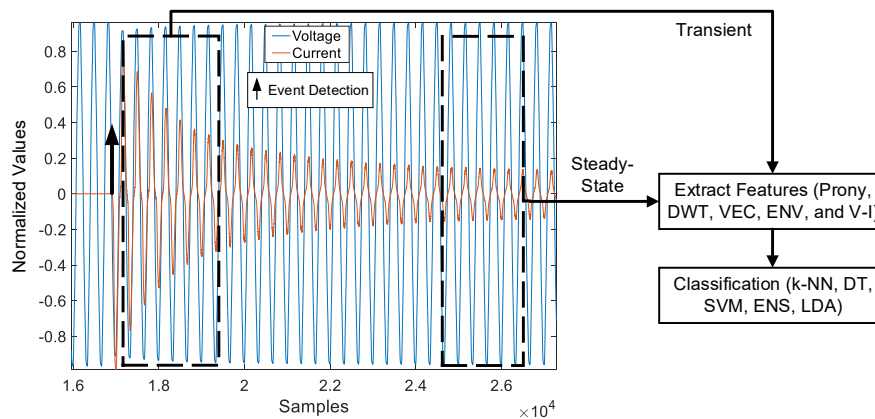


Figure 7. Overview of the Feature Extraction and Classification Procedure. Voltage and Current Values are normalized only for illustration purposes.

3.3.1. V-I Trajectories

The V-I trajectory consists of the graphical representation of the pair (V,I), which represents the instantaneous voltage and current values, along with their respective waveforms. This trajectory has distinct characteristics for different classes of devices (e.g., resistive, motorized, or electronic) depending on their operating principles [35], which allows the extraction of different parameters. In [35], Lam and his colleagues proposed shape features associated with the physical characteristics of a load:

- *Asymmetry*: characteristic presented by loads that have different shapes and peak current in the positive and negative semi cycles. This feature is represented in Figure 8a by the blue trajectory (*asy*).
- *Area*: the area of the V-I curve, which is proportional to the phase difference between V and I. In Figure 8a, this feature is represented by the pink trajectory.
- *Curvature of the mean line (similar to harmonic distortion)*: represents the non-linearity of load, as shown by the dashed line (*r*) in Figure 8a.
- *Area of left and right segments*: relates to the time displacement between the peaks of the V and I waveforms. In Figure 8b, this feature is the sum of parts (*alr*).
- *Peak of middle segment*: represented by *mi* in Figure 8b and very significant in loads with small power consumption.
- *Looping direction*: if the phase angle between the V and I for this load is positive, the direction of points in the trajectory is clockwise; otherwise, the direction is counterclockwise. In Figure 8a this feature is represented by *lpa*.
- *Slope of the middle segment*: helps on distinguishing power electronic loads from other loads, since power electronics usually present a horizontal middle segment. Represented by the standard deviation of all points in part *sh* in Figure 8b.
- *Self-intersection*: a significant feature when the load is highly nonlinear as it is proportional to the order of harmonics that the load contains. It is represented by *sc* in Figure 8a.

In [40], the authors presented load classification results using the features proposed in [35,41]. One extra feature was also introduced, consisting of the *Distance between the maximum and minimum points* in V-I trajectory, *dpb* in Figure 8a.

Later on, in [21], the authors discussed a possible quantification and formalization of the mentioned features. They also proposed two new steady-state features: *Current span*, which is

associated with the active power magnitude of a load; and *Variation of instantaneous admittance*, which allows distinguishing between resistive and non-resistive appliances loads.

As these are steady-state features, in [27], the authors presented a set of new features for the V-I trajectories both for the steady-state and for transient segments:

- *Angle between the maximum and minimum points* of a V-I trajectory in steady state, represented by $angp$ in Figure 8a.
- *Length* of the trajectory V-I in steady state, by calculating the sum of the distances between the consecutive points that define the trajectory.
- *Maximum distance*, which consists of the maximum amplitude between all the vectors formed by the V-I trajectory points and the origin (0,0), md in Figure 8a.
- *Variations of the signal slope*, which consist of the number of times the signal slope changes between positive and negative slopes. This signal change is another way of representing signal frequency information, as presented in [42].
- *Average slope value*, which is calculated by using the sum of the inclinations among all points of the trajectory, is divided by the number of points. This feature is the mean of ω , between all the points, in Figure 8a.
- *Trajectory Centroid*, from which three features are extracted: (i) the distance between the origin point of the plane (0,0) and the centroid of the trajectory in steady-state, dc in Figure 8a; (ii) the angle formed between the two points, $angc$ in Figure 8a; (iii) the distance between the centroid of the transient trajectory and the centroid of the stationary trajectory.
- *Overshoot*, which can be estimated for the current signal as the difference between the maximum point of the current in the trajectory in the transitory state and the maximum point of the current in the trajectory in the steady-state. In Figure 8b, this feature is represented by ov .
- *Difference between points distance*, which is estimated by using the distance and the angle between the maximum and minimum points of the V-I trajectories in transient (β) and stationary (θ) states, as shown in Figure 8b, allowing the calculation of the difference between the distances and the difference between the angles.

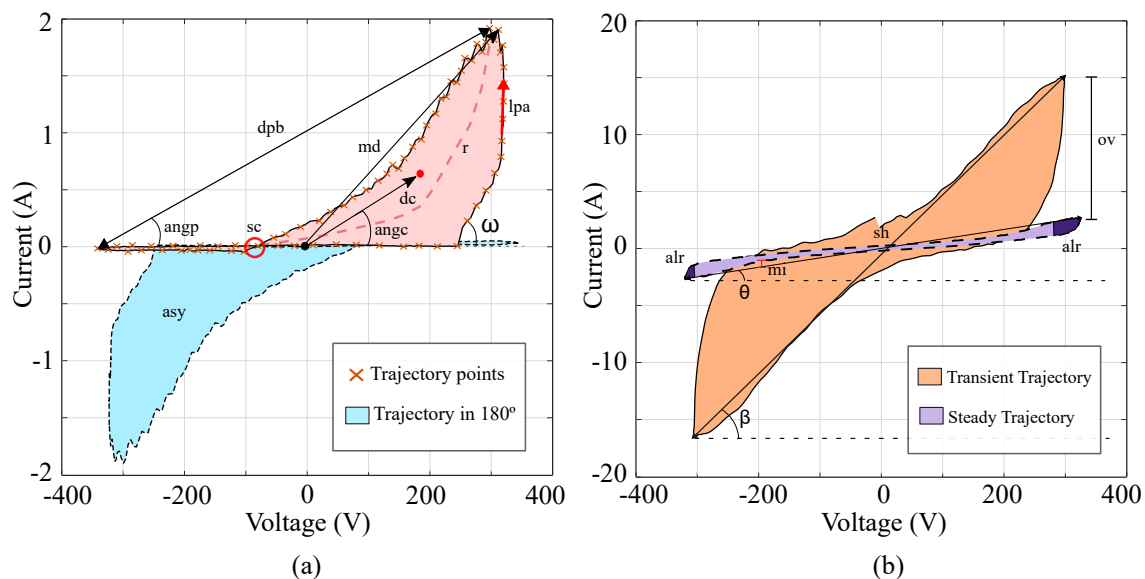


Figure 8. V-I trajectories for COOLL dataset, where (a) Drill at Half-speed, and (b) Drill at Full-speed.

3.3.2. Wavelet

The wavelet decomposition used for the event classification is similar to the approach used in the context of detection. However, in this second stage, the energy of each wavelet level is used as a feature for subsequent classification. Each signal is decomposed using a Discrete Wavelet Transform in a given number of levels [15,23,25]. Then, the energy contents in each DWT level is computed to reduce dimensionality. This is typically done based on the energy contents of the signal during the occurrence of the event. For instance, if a signal has a sampling frequency of 15,360 Hz, and a five-level DWT decomposition is applied with a given mother-wavelet, then the following frequency distribution is observed:

- Level 1 (Detail-1): 7680 Hz–3840 Hz;
- Level 2 (Detail-2): 3840 Hz–1920 Hz;
- Level 3 (Detail-3): 1920 Hz–960 Hz;
- Level 4 (Detail-4): 960 Hz–480 Hz;
- Level 5 (Detail-5): 480 Hz–240 Hz;
- Approximation signal: 240 Hz–0 Hz.

By calculating the energy content in each level, the frequency content of the signal in different bands can be extracted. As discussed in [23], such frequency distribution presents a high discriminatory property to classify electrical signals.

3.3.3. Prony

As previously shown in [10,43], the transient currents of electrical equipment (also called inrush currents) can be used to classify loads. These currents are characterized by their high values right after the load starts consuming energy and decreasing values as the steady-state is reached.

The Prony's method can be used to estimate the damping and harmonic content of the current signal, as it models a signal as a linear composition of damping exponentials applied to sinusoidal signals of varying frequencies. The Prony is implemented in three basic steps: (i) determine the linear prediction parameters that fit available data, (ii) estimate the damping and sinusoidal frequencies of the exponential terms, and (iii) estimate the exponential amplitude and initial phase of sinusoidal component [44].

Assuming a sampled signal x composed of N data samples $x[1], \dots, x[N]$, the Prony's method estimates $x[n]$ with a p -term complex exponential as

$$\hat{x}[n] = \sum_{k=1}^p A_k \exp[(\alpha_k + j2\pi f_k)(n-1)T_s + j\theta_k], \quad (5)$$

in which T_s , A_k , α_k , f_k , and θ_k are the sampling period, amplitude, damping factor, frequency, and phase, respectively. Throughout the estimation of parameters, different linear systems must be solved. Several approaches have been proposed to simplify the solution, apart from the classical Polynomial method, such as Least Squares, Total Least Squares, and Matrix Pencil Method. As presented in [28], the Matrix Pencil approach results in a lower reconstruction error and a higher classification accuracy for different NILM datasets, being the most appropriate method to extract Prony's features of electrical current signals.

3.3.4. Power Envelope

The waveforms of active, reactive, and apparent power in each semi cycle are calculated from the voltage and current samples. Then, by analyzing their variations, these waveforms are partitioned into stable and transient segments. In every transient segment, a power envelope is extracted from the difference to the previous stable segment. This envelope is then classified into five types (Figure 9):

(i) Negative Exponential; (ii) Spike; (iii) Step; (iv) Slow Increase; (v) Other. The first two present a high inrush current that decays exponentially for the Negative Exponential or very rapidly (up to four semi cycles) for the Spike. The Step envelope has none or a small (up to 20% of the steady-state power level) overshoot; thus, it has a short transient period. Finally, the Slow Increase envelope is characterized by an increasing power over several tens of cycles, during which the load current slowly increases. Whenever a waveform's envelope characteristics do not fit any of these four types (i to iv) then it is classified as "Other". For each of these envelope types, particular features are extracted, including: (i) the duration of the transient periods; (ii) the overshoot (peak power versus stable power); (iii) the power difference between the start and the end of the transient period, and (iv) the number of semi cycles from power-on to the peak inrush power. Waveforms classified as "Other" have all these features extracted.

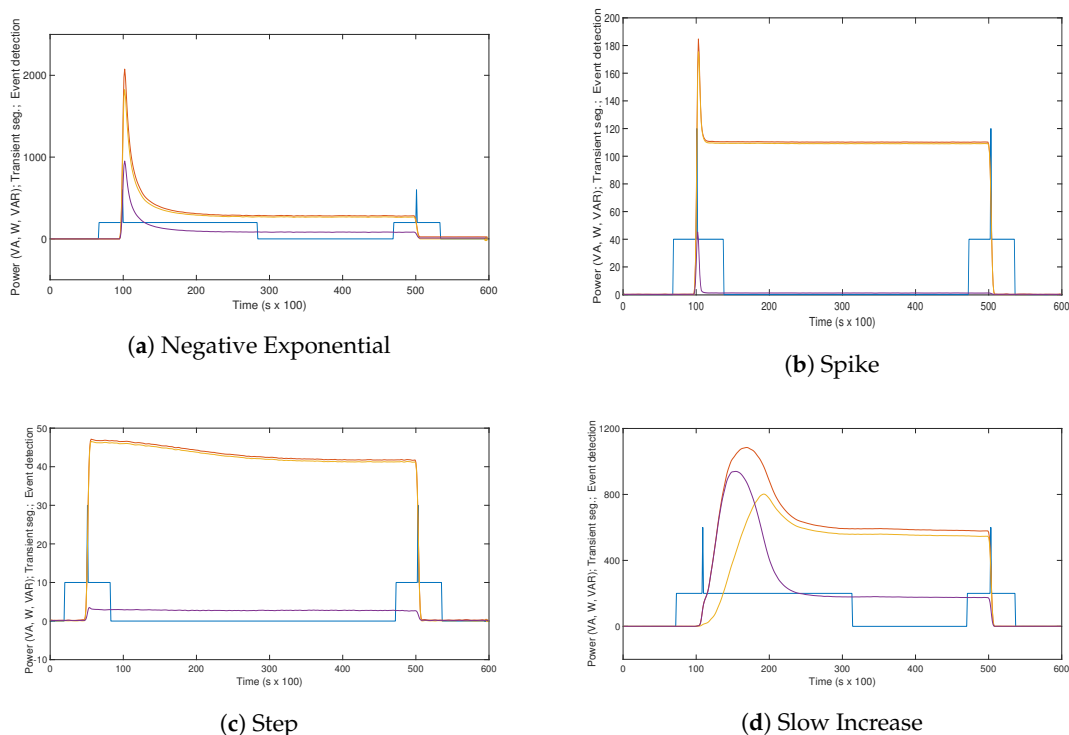


Figure 9. Examples of the four types of power envelopes. Apparent power (VA) in red, active power (W) in yellow, and reactive power (VAR) in purple. The light blue line shows stable power periods, power transient periods, and load events.

3.3.5. Vectorization

The Vectorization method (Section 3.1.5) generates a vector of complex numbers that represent a segment of a waveform: $[n_s[0] + jv_{ref}[0], n_s[1] + jv_{ref}[1], \dots, n_s[k-1] + jv_{ref}[k-1], n_s[k] + jv_{ref}[k]]$. These values, generated during the state identification, can be used as features for subsequent classification. However, different signals may have different complex vector lengths. Hence, as feature vectors must have the same length for the selected classifiers, the larger vectors have their size reduced using nearest-neighbor interpolation. This procedure produces vectors of equal sizes (20) for all loads.

3.4. Load Classification Agents

This Section presents the description of classification agents, emphasizing theoretical aspects and parameters of each selected machine learning classifier: k-NN, DT, SVM, ENS, and LDA used in the proposed architecture.

3.4.1. k-Nearest-Neighbors (k-NN)

The k-Nearest-Neighbors classifier classifies the features vector by associating it to the previously classified features vectors at the closest Euclidean distance [45]. As the classification is simply based on distances related to a training set, this method may be considered one of the simplest machine learning algorithms [45], as it is fairly easy to be implemented. On the other hand, this method requires the storage of all the training data, which may impose problems on embedding it into platforms with limited resources [7].

3.4.2. Decision Tree (DT)

The classification using Decision Tree (DT) consists of performing different binary classifications and concatenating them in a tree structure, in which each node concerns each variable and parameter. In the end, this combination of different evaluations allows obtaining the final class label. The trees and their branches are defined during training based on the Information Gain [45]. DTs may be considered one of the most common methods for fault classification presented in the literature [45].

3.4.3. Support Vector Machine (SVM)

Support Vector Machines were originally developed to solve classification problems using the concept of an optimum separation hyperplane, which maximizes the separation margin ρ between classes [46]. The maximization of ρ is based on a complexity measurement known as the Vapnik–Chervonenkis (VC) dimension [46], whose upper limit is inversely proportional to ρ . This maximization can be formulated as a convex optimization problem, which takes into account the model complexity and goodness of fit to the training data. Additionally, the nonlinear input mapping, implicit to the SVM formulation, can be replaced by a dot product kernel in feature space [45]. There are several types of kernel, which must abide by the conditions of Mercer’s theorem—in this work, we use the Gaussian kernel.

3.4.4. Ensemble Method (ENS)

Using an Ensemble Method implies combining different classifiers (usually by averaging) to improve the accuracy that those individual classifiers can provide [45]. Ensemble classification combines a set of trained weak learner models. It can predict ensemble responses for new data by aggregating predictions from its weak learners. This method can use different algorithms for sequential learning (weaker learning models), such as *AdaBoostM1*, *AdaBoostM2*, *Bag*, *GentleBoost*, *LogitBoost*, *LPBoost*, *LSBoost*, *RobustBoost*, *RUSBoost*, *Subspace*, and *TotalBoost* [45].

3.4.5. Linear Discriminant Analysis

The objective of this method is to reduce the features to a lower-dimensional space, maximizing the separation between classes, so its complexity and required computational resources are reduced, as well as the possibility of overfitting. Thus, the classifier avoids adapting to noise instead of relevant data [45].

3.5. Combination of Classification Agents

The Signature Analysis Composition (SAC) block combines the individual extraction from FEAs and classification from CAs to produce a single output that reports the most likely load type. Similarly to EDC, the principle of operation of the SAC is a weighted voting among the individual FEAs and CAs, detailed as follows:

$$pc_{ji} = \alpha_1 \left(\frac{1}{n_{agc}} \sum_{l=1}^{n_{agc}} I \left(Cout_{jl} = class_i \right) \right) + \alpha_2 \left(\frac{1}{n_{cc}} \sum_{l=1}^{n_{agc}} I \left(Cout_{jl} = class_i \right) acc_{il} \right) + \alpha_3 \left(\frac{1}{n_{agc}} \sum_{l=1}^{n_{agc}} I \left(Cout_{jl} = class_i \right) sc_{il} \right), \quad (6)$$

in which $Cout$ is the output of each agent for each load event to be classified (segment of a waveform) $\{Cout \in \mathbb{N}^{n_{examples} \times n_{classes}} \mid Aout_{jl} \in \{1, 2, 3, \dots, n_{classes}\}\}$, sc is the event classification likelihood (degree of confidence) of each agent in each class $\{sc \in \mathbb{R}^{n_{classes} \times n_{agc}} \mid 0 \leq sc_{il} \leq 1\}$, and acc is the performance of each agent for each one of the classes ($n_{classes}$). Where $\{acc \in \mathbb{R}^{n_{classes} \times n_{agc}} \mid 0 \leq acc_{il} \leq 1\}$; $I(x)$ is an operator that returns 1 if the condition in x is true and zero otherwise, while $n_{examples}$ is the total number of load events to be classified, n_{agc} is the total number of classifier agents, $n_{cc} = \sum_{i=1}^{n_{agc}} I(Cout_{jl} = class_i)$, and $class = [1, 2, 3, \dots, n_{classes}]$.

Finally, pc is a matrix with the number of rows equals to the number of examples to be classified, and the number of columns equals to the number of classes (Unlike the combination of detection agents, the classes in this case represent the loads to be classified. Also, it is noteworthy that the number of classes is different for each dataset), i.e., $\{pd \in \mathbb{R}^{n_{examples} \times n_{class}} \mid 0 \leq pd_{ij} \leq 1\}$. The selected final class corresponds to the column with the highest pd_{ji} value.

Each portion of the equation takes into account a weight (α_1, α_2 , and α_3 subject to $\alpha_1 + \alpha_2 + \alpha_3 = 1$), defined a priori and selected to maximize the final classification accuracy. Once these parameters are defined, Equation (6) is applied for $i = [1, 2, 3, \dots, n_{classes}]$ and $j = [1, 2, 3, \dots, n_{examples}]$ to fully calculate pd .

4. Methods

To compare the performance of the detection and classification agents, as well as their combined results, the datasets used, the detection and classification metrics, the selection of configuration parameters of the agents, and the training procedures were used. The following sections detail these topics.

4.1. Datasets

To evaluate the performance of the detection and classification agents, two high-frequency datasets (High-frequency datasets are those that register voltage, current, or power at a rate greater than 1 Sps [13]) were used: COOLL (Available at <https://coolldataset.github.io/>) and LIT (Available at http://dainf.ct.utfpr.edu.br/~douglas/LIT_Dataset/index.html). Two subsets of the LIT-Dataset were used: the Synthetic (LIT-SYN) and the Simulated (LIT-SIM). The main characteristics of these datasets are presented in Table 2 and detailed as follows.

Table 2. COOLL and LIT datasets characteristics.

Characteristic	COOLL	LIT-SYN	LIT-SIM
Date	2016	2019	2019
Residential	Yes	Yes	Yes
Commercial	No	Yes	Yes
Industrial	No	Yes	Yes
High-Frequency sampling (kHz)	100 kHz	15.6 kHz	15.6 kHz
Precise indication of load events	No	Yes	Yes
Number of Waveforms	840	1664	4824
Acquisitions Time (s)	6 s	30 to 40 s	2.5 to 16 s
Number of classes	12	16	7
Number of appliances	42	20	28
Multiple simultaneous loads	No	Yes	Yes

4.1.1. COOLL Dataset

The COOLL dataset [47] is composed of voltage and current waveforms from 42 home appliances organized into 12 load classes. Waveforms were acquired at a sampling frequency of 100 kHz during 6 s each. For each load, 20 waveforms were recorded at different turn-on trigger angles.

Table 3 presents the 12 home appliances classes and the number of appliances used for each class, resulting in a total of 840 waveforms, with 20 waveforms per appliance. In this work, the agent classifiers are used to distinguish among the 42 appliances that compose the COOLL dataset.

Table 3. Appliance classes in the COOLL dataset (based on [47]).

Set of Classes	Number of Appliances
Saw	8
Vacuum cleaner	7
Drill	6
Hairdryer	4
Lamp	4
Hedge trimmer	3
Sander	3
Fan	2
Grinder	2
Paint stripper	1
Planer	1
Router	1
Total	42

Note: There are 20 waveforms recorded for each of the 42 appliances, totaling 840 waveforms.

4.1.2. LIT-Dataset

The LIT-Dataset is composed of three different subsets: Synthetic [31], Simulated [32], and Natural [33]. In the Synthetic subset data is collected using a 1 kW jig to switch up to eight loads on and off, in a programmable pattern. In the Simulated subset, different types of loads are simulated using MATLAB Models and circuits, enabling to collect data in different scenarios and to control parameters that otherwise wouldn't be possible. In the Natural subset, data is collected in an uncontrolled, real-world environment; however, power sensors monitor each energy outlet, identifying and recording the time that a load event occurs. The Synthetic and Simulated subsets were used to test the detection and classification methods.

The Synthetic subset comprises residential, commercial, and industrial loads, with a sample frequency of 15,360 Hz and precise indications of load events (<5 ms). The AC mains voltage and aggregated current are monitored for up to 40 s. A total of 16 load classes, linear and non-linear, as shown in Table 4, were acquired individually (LIT-SYN-1), and in combinations of two loads (LIT-SYN-2), three loads (LIT-SYN-3), and eight loads (LIT-SYN-8). The loads' power range from 4 W up to 1 kW, in a total of 1664 waveform acquisitions. For every load or loads' combinations, 16 turn-on trigger angles in respect to the AC mains were used, as the turn-on angle directly affects the loads' inrush current.

Table 4. Characteristics of the synthetic subset of the LIT-dataset.

Class	Loads Types	Num. of Appliances	Num. of Waveforms in LIT-SYN-1
1	Microwave Oven (standby and on)	1	32
2	Hair Dryer (two fan speed levels)	2	64
3	Hair Dryer (two power levels)	1	32
4	Led Lamp	2	32
5	Incandescent Lamp	1	16
6	CRT Monitor	1	16
7	Led Monitor	1	16
8	Fume Extractor	1	16
9	Phone Charger	2	32
10	Laptop Charger	2	32
11	Drill (two speed levels)	1	32
12	Resistor	1	16
13	Fan	1	16
14	Oil Heater (two power levels)	1	32
15	Soldering Station	1	16
16	Resistive Heater	1	16

The Simulated subset has seven different classes, each one with four different power levels (simulating different loads), as shown in Table 5, ranging from 15.5 W up to 12.2 kW. The load switching events are precisely controlled at the sample level. The simulation calculation step is 1 μ s, resulting in 16,667 points in one AC mains cycle. The samples were decimated from 1 MHz to 15,360 Hz, the same sampling frequency used in the LIT Synthetic and Natural subsets. The simulation comprises single, twofold, threefold, fourfold, and fivefold load combinations, with trigger angles of 0°, 45°, and 90°. Also, six different scenarios of the AC mains were simulated, as shown below:

1. ideal (SLIT1);
2. with a stray inductance (SLIT2);
3. with a stray inductance and a total harmonic distortion in the AC mains of 6.8% (SLIT3);
4. with a stray inductance, a total harmonic distortion in the AC mains of 6.8%, and an Additive White Gaussian Noise (AWGN) with a Signal-to-Noise Ratio (SNR) of 10 dB (SLIT4);
5. a stray inductance, a total harmonic distortion in the AC mains of 6.8%, and an AWGN with an SNR of 30 dB (SLIT5);
6. with a stray inductance, a total harmonic distortion in the AC mains of 6.8%, and an AWGN with an SNR of 60 dB (SLIT6).

For the performance evaluation of agents, the SLIT5 scenario was used, with individual, 2, 3, 4, and 5 loads combinations, each one with three to nine turn-on trigger angles, resulting in a total of 804 acquisitions, for 28 loads. This subset was used in this work because it represents the most similar characteristics to actual acquired signals in our laboratory [36].

Table 5. Characteristics of the Simulated Subset of the LIT-Dataset—Scenario SLIT5—Number of single load acquisitions.

Class	Classes	Power Levels (Loads)	Num. of Waveforms
1	Resistor (R)	4	3
2	Resistor and Inductor (RL)	4	3
3	Half Wave Diode Rectifier	4	3
4	Full Bridge Diode Rectifier	4	3
5	Half Wave Thyristor Rectifier with R load	4	3
6	Half Wave Thyristor Rectifier with RL load	4	3
7	Universal Motor	4	3

4.2. Detection Metrics

To quantitatively evaluate each agent's performance, the following metrics were used.

4.2.1. Power-ON Event Detection

To evaluate the correct detection of power-ON events, the PC_{on} (Percentage of Correct detection of an ON event for a given method) was used. This metric is defined as the ratio between the number of correct ON detections (A_{on}) and the number of power-ON events registered in the waveforms (N_{ON}):

$$PC_{on} = \frac{A_{on}}{N_{ON}}. \quad (7)$$

A detection tolerance window of ± 10 semi cycles (in relation to real ON event) was used; hence, if a detection agent indicates a power-ON event within ± 10 semi cycles of the actual event, this detection is counted as a valid one.

Considering the COOLL dataset, where each of the 840 waveforms has a single power-ON event, a PC_{on} of 100% indicates that all the actual power-ON events were correctly detected by this agent.

4.2.2. Power-OFF Event Detection

In the same way as for power-ON events, the Percentage of Correct detection of a power-OFF event (PC_{off}) is defined as the ratio between the number of correct Off detections (A_{off}) and the number of OFF events registered in the waveforms (N_{OFF}):

$$PC_{off} = \frac{A_{off}}{N_{OFF}}. \quad (8)$$

Again, a tolerance window of ± 10 semi cycles was used.

4.2.3. Average Number of False Positives and False Negatives

The False Positives (FP) are "ON" or "OFF" event detections that are outside the tolerance window (± 10 semi cycles). Hence, a false positive occurs when a detection agent indicates an event that did not occur or was reported outside the tolerance window. As the FP evaluation considers a whole dataset, its values are defined as the mean value over all the waveforms within that dataset. The Average number of False Positives (AFP) is defined as (9).

$$AFP = \frac{1}{N} \sum_{i=1}^N FP_i, \quad (9)$$

in which N stands for the number of waveforms in the dataset. In case of multiple detections within the tolerance window, only the first detection is considered as valid, and the other detections inside the tolerance window are not considered as false positives. Then, the Average number of False Positives (AFPs) is the average of all the false positives per waveform, within a dataset.

False negatives, on the other hand, correspond to actual "ON" or "OFF" events that have not been detected. The Average number of False Positives (AFNs) is also calculated as the average of all the False Negatives (FN) per waveform, within a dataset, as presented in (10).

$$AFN = \frac{1}{N} \sum_{i=1}^N FN_i \quad (10)$$

An AFP of 1 means that, on average, in each waveform, a single false positive (either power-ON or power-OFF) occurred. Likewise, an AFN of 1 means that, on average, in each waveform, a single false negative occurred.

4.2.4. Average Distance between Actual “ON” and Its Detection

The average distance between an actual power-ON and its detection, as reported by a detection agent, D_{on} , as the ratio between the sum of the absolute distances (in semi cycles) between the detection and the actual power-ON event and the total number of correct detections, A_{on} :

$$D_{on} = \frac{\sum_{i=1}^{A_{on}} |d_{on}(i) - ev_{on}(i)|}{A_{on}}. \quad (11)$$

in which $d_{on}(i)$ represents the semi cycle in which the first “ON” detection occurred (within the tolerance window), and $ev_{on}(i)$ is the semi cycle in which the actual event has occurred. Only the detections that have incremented A_{on} in (7) are considered.

A D_{on} of 4 means that, on average, this detection agent identified a power-ON event within 4 semi cycles of the time where the actual event occurred.

4.2.5. Average Distance between Real “OFF” and Its Detection

It is also possible to establish a distance metric for power-OFF events: D_{off} . It is defined as a ratio between the sum of the absolute values of the distances in the detection of “OFF” events, and the total number of correct detection, A_{off} :

$$D_{off} = \frac{\sum_{i=1}^{A_{off}} |d_{off}(i) - ev_{off}(i)|}{A_{off}}. \quad (12)$$

being $d_{off}(i)$ the semi-cycle in which the first “OFF” detection has occurred, in the tolerance window, and $ev_{off}(i)$ is the semi-cycle labeled with the actual event.

4.3. Classification Metric

For the overall performance assessment of the classification agents, a confusion matrix with Y classes can be constructed, being Y the number of loads in each dataset or subset. Using this confusion matrix, the performance for all load classes (ACC) in terms of classification accuracy can be calculated, using individual performances (ACC_i) per class i , as follows:

$$ACC = \frac{1}{Y} \sum_{i=1}^Y ACC_i. \quad (13)$$

The individual performance is computed by the number of corrected classified load events (CCE_i) divided by the total number of load events (TNE_i) per load class:

$$ACC_i = \frac{CCE_i}{TNE_i}. \quad (14)$$

Then, the average of individual agents’ performances is used to reduce the impact of load class imbalance on the final result, allowing an adequate comparison among all agents, as well as their combination.

4.4. Parameter Selection

In each detection, extraction, and classification agent, several parameters must be tuned, to enhance each agent’s performance and provide a fair comparison among agents. To do so, we selected the COOLL dataset as a reference for parameter selection. With that, all experiments were carried out aiming at maximizing the detection accuracy off power-ON and power-OFF events, as well as the classification performance, using all the 840 waveforms from COOLL dataset.

The first detection agent, Kalman filter, had its internal parameters adjusted to maximize the variation of the innovation in the decoupling moments between the model and the measurements (transient in the network): number of harmonics varied from 8 to 10, based on our previous works with Kalman filter detection [16]; size of the input window fixed in the number of samples needed to represent one cycle of the power network; and minimum step of the input window displacement was set at 25% of the total window size (one cycle). Empirically, we reached $R = 0.001$ and $Q = 0.1$.

Different wavelet functions were evaluated for the DWT, such as Daubechies (db) $\{db4, db8\}$, Symmlet (sym) $\{sym2, sym8\}$, and Coiflets (coif): $\{coif2, coif4\}$. The number of levels was varied in the set $\{1, 2, 3, 4, 5\}$. The HCApP method used 8 half-cycles for integration, with the effect of an averaging filter, as well as an adaptive threshold to reduce the effect of noise and power variations of the loads. The DWT and Kalman detection methods were used in combination with the adaptive threshold previously presented. The thresholds were varied from γ standard deviations, relative to the mean, with γ ranging from 1 to 15. Regarding the HAND method, the parameters were set according to the authors' original suggestion [11].

For the Vectorization method, the tuning parameters imply different representations of vectors: (i) a higher value of n implies a lower number of elements in the vector; (ii) a lower value of l_v implies in more restrict characterization of valid states, increasing the number of non-valid states. In this sense, these parameters were searched in the set $\{0.01, 0.02, 0.05, 0.1, 0.2, 0.3\}$ and set $\{10, 20, 30, 40, 50\}$, respectively. The same set was used for feature extraction as the detection and extraction of features are coupled in the Vectorization method.

For all other feature extraction methods, different parameters were also experimented. With respect to the DWT, the energy was calculated using the same levels and wavelet functions as presented in the detection stage. For V-I trajectories, the transient and steady-state feature parameters were based on the recommendation presented in [27]. Prony's parameters and solving procedures are based on the comparison discussed in [28]. For the power envelope technique, first, the envelope is analyzed, and its type is determined. Then, based on envelope type, particular features are extracted, including: (i) the duration of the transient periods; (ii) the overshoot (peak power versus stable power); (iii) the power difference between the start and the end of the transient period, and (iv) the number of semi cycles from power-on to the peak inrush power.

The classification parameters for each classification method must be defined as follows: the number of neighbors (k) for k-NN; the regularization and kernel parameters for SVM; the weaker learning models for Ensemble; the depth of the trees for Decision Trees; the linear coefficient threshold and regularization parameter for LDA.

For EDC and SAC, the parameters γ_1 , γ_2 , γ_3 , α_1 , α_2 , and α_3 were searched in the set $\{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$, respecting the constraints: $\gamma_1 + \gamma_2 + \gamma_3 = 1$ and $\alpha_1 + \alpha_2 + \alpha_3 = 1$. In this manner, it is possible to evaluate different contributions of each term in Equations (4) and (6).

4.5. Training Procedure

In terms of training of the classification agents, the procedure was divided into two stages: (i) training; and (ii) test steps. During the training, the feature extraction is performed for each agent, and part of the available examples (training set) is used in a Bayesian Optimization to select the parameters of each classifier. At the end of this step, different classifiers—classification models—are defined for later evaluation.

During the test stage, models built in the training stage are evaluated using a different subset of the dataset than the one used in the training step (test set). The overall performance evaluation is then performed, which contemplates the computation of the classification accuracy for each load class and the corresponding average accuracy among all load classes. This procedure was performed using the Matlab's Statistics and Machine Learning Toolbox [48]. Such an approach is

equivalent to the standard cross-validation procedure [45] to select classifier parameters and, in general, less computationally intense, as presented in [48].

To perform the training and test procedures, 80% of the available examples were used as the training set, and the remaining 20% were used as a test set. Additionally, we organized ten different training and test subsets, randomly selecting a percentage of examples per load class, including all the features, aiming to obtain an average of the classification accuracy for different training and test set configurations. It is noteworthy that the indices used to organize each subset were fixed for all the feature extraction methods, i.e., the 10 subsets are the same for all comparisons. Figure 10 summarizes the main steps of the classification procedure. The same procedure was used for the combination of agents.

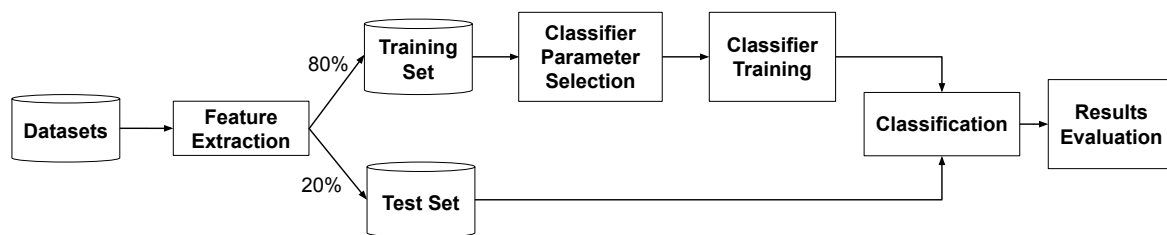


Figure 10. Overview of the classification procedure.

Before being fed to the classifiers, all features are normalized in the range of $[-1, 1]$ for all models. This normalization is only applied in the feature domain, i.e., it has no relation to the voltage and current waveforms.

Finally, we also evaluated the classification performance of a model trained with a set of individual or multiple loads when classifying waveforms with multiple loads. The LIT-SYN dataset was used as it provides individual and multiple loads. The experiments were divided into the following sets:

- *LIT-SYN-1 to LIT-SYN-2*: Models are trained with individual loads from LIT-SYN-1 and evaluated (tested) with data from LIT-SYN-2;
- *LIT-SYN-1 to LIT-SYN-3*: Models are trained with individual loads from LIT-SYN-1 and evaluated (tested) with data from LIT-SYN-3
- *LIT-SYN-1 to LIT-SYN-8*: Models are trained with individual loads from LIT-SYN-1 and evaluated (tested) with data from LIT-SYN-8;
- *LIT-SYN-3 to LIT-SYN-8*: Models are trained with individual loads from LIT-SYN-3 and evaluated (tested) with data from LIT-SYN-8.

With the first three sets, it is possible to verify the performance of the model trained with individual loads on waveforms with multiple loads, as the number of multiple loads increases (two, three, and eight). The last set comprises the performance evaluation of a model trained with three concurrent loads when classifying up to eight concurrent loads. The purpose of these analyzes is to verify, in terms of classification accuracy, how much the typical signature of an electric load, both in transient and steady-state, is maintained as multiple loads are added. To the extent of our knowledge, such analysis is unexplored in the literature.

5. Results and Discussion

In the next sections, the different evaluation experiments that we conducted are described:

- Individual assessment of each detection agent. The objective of this experiment is to evaluate how each agent, according to the proposed metrics, performs the detection of power-ON and power-OFF events in different databases, including multiple loads.

- Performance evaluation with detection agents combination. To evaluate how different agents' combinations, according to the same metrics, performs the detection of power-ON and power-OFF events in different databases, including a comparative analysis with individual agents.
- Evaluation of the performance of each extraction and classification agent. The objective is to evaluate how each extraction and classification agent behaves in terms of classification accuracy for different databases, including multiple loads. The generalization ability of classifiers trained with simple loads and evaluated with multiple loads is also discussed.
- Evaluation of the computational cost of each detection, extraction, and classification agent, providing an initial assessment of the computational resources demand, aiming at the future use of this multi-agent architecture in an embedded system.
- Performance evaluation with extraction and classification agents' combination. To evaluate how different combinations can improve the classification accuracy for different databases, including multiple loads.

5.1. Detection

The results for all detection agents using the proposed metrics (Section 4.2) are presented in Table 6. In terms of power-ON accuracy, DWT presented the best results for the COOLL dataset, detecting 98.9% of the power-ON events in that dataset. Similar results are observed for HAND and HCApP. For SLIT5, LIT-SYN-1, LIT-SYN-2, and LIT-SYN-8, HAND presented the best accuracy, detecting around 94% of the events in those datasets. For LIT-SYN-3, HCApP presented the best results, with 81.5% of accuracy. On average, the best methods for ON events are HCApP and HAND, with 84.3% and 85.1% of accuracy, respectively. DWT, KF, and VEC showed a relatively lower performance, mainly for datasets with multiple loads, such as SLIT5, LIT-SYN-2, LIT-SYN-3, and LIT-SYN-8.

In terms of OFF accuracy, the best method is HCApP (85.6%), but similar results are obtained for VEC and HAND—around 82% on average, for all datasets. DWT and KF do not capture transient from OFF events [16], making the OFF detection completely compromised for these methods. As observed for ON detection, OFF accuracies decrease as the number of multiple loads increases in LIT-SYN and SLIT5. This can be explained by the combination of loads with a high difference in electrical power. For instance, in LIT-SYN-2, LIT-SYN-3, and LIT-SYN-8, there are loads of 10–20 W that are switched-ON and OFF when loads of 800–1000 W are ON. So, in those cases, the detection of load switching transients is hindered. An example of a two load acquisition from LIT-SYN is presented in Figure 11. An oil heater and then a LED lamp are triggered at different instants. As the oil heater has a higher power (520 W), the turn-on event of the led lamp (6 W) may be very difficult to detect, as detailed in Figure 12.

Table 6. Results for Detection Agents.

Method	Metric	COOLL	SLIT5	LIT-SYN-1	LIT-SYN-2	LIT-SYN-3	LIT-SYN-8	Avg.
DWT	PC_{on}	<u>98.9</u>	67.9	92.3	78.7	56.8	40.7	72.8
	PC_{off}	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	AFPs	3.4	14.3	3.9	4.8	37.3	46.9	18.3
	AFNs	1.0	4.1	1.0	2.4	4.8	13.9	4.4
	D_{on}	<u>0.4</u>	0.5	<u>0.1</u>	<u>0.2</u>	<u>0.3</u>	0.7	<u>0.3</u>
	D_{off}	–	–	–	–	–	–	–
HAND	PC_{on}	94.4	<u>96.4</u>	<u>95.7</u>	<u>94.4</u>	76.3	<u>61.2</u>	<u>85.1</u>
	PC_{off}	98.3	87.9	91.8	87.1	78.6	46.4	82.3
	AFPs	1.6	1.9	2.2	4.6	1.5	3.5	2.6
	AFNs	<u>0.1</u>	<u>0.5</u>	<u>0.1</u>	<u>0.4</u>	1.3	7.9	1.7
	D_{on}	1.1	<u>0.7</u>	1.3	1.2	1.1	1.1	1.1
	D_{off}	<u>0.6</u>	<u>0.7</u>	<u>0.5</u>	0.9	0.7	1.1	<u>0.7</u>
KF	PC_{on}	85.0	35.9	57.9	44.8	64.8	60.5	60.3
	PC_{off}	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	AFPs	69.8	28.7	0.5	<u>0.2</u>	1.8	<u>2.3</u>	6.8
	AFNs	0.8	3.2	1.2	2.9	2.8	11.2	3.6
	D_{on}	5.4	5.9	2.1	1.6	4.8	2.8	3.9
	D_{off}	–	–	–	–	–	–	–
VEC	PC_{on}	64.6	67.9	47.2	44.4	51.5	42.1	52.6
	PC_{off}	<u>100.0</u>	72.7	<u>96.2</u>	<u>92.8</u>	72.4	53.6	81.6
	AFPs	9.9	5.4	17.5	29.4	10.2	13.7	14.4
	AFNs	0.1	1.7	0.1	0.2	1.7	8.3	1.9
	D_{on}	1.9	3.8	1.7	4.6	4.9	4.8	3.6
	D_{off}	2.7	3.1	3.3	3.5	3.2	2.6	3.1
HCApP	PC_{on}	95.2	92.4	94.7	88.7	<u>82.8</u>	51.9	84.3
	PC_{off}	99.6	<u>88.1</u>	94.9	85.3	<u>89.6</u>	<u>56.1</u>	<u>85.6</u>
	AFPs	<u>0.1</u>	<u>0.1</u>	<u>0.5</u>	0.5	<u>0.5</u>	<u>0.5</u>	<u>0.3</u>
	AFNs	<u>0.1</u>	0.7	<u>0.1</u>	0.5	<u>0.8</u>	8.2	<u>1.7</u>
	D_{on}	0.7	0.8	0.5	0.5	0.6	<u>0.6</u>	0.6
	D_{off}	0.9	0.9	0.6	<u>0.5</u>	<u>0.5</u>	<u>0.6</u>	<u>0.7</u>

Note: The best results for each dataset and each metric are highlighted in bold and underlined.

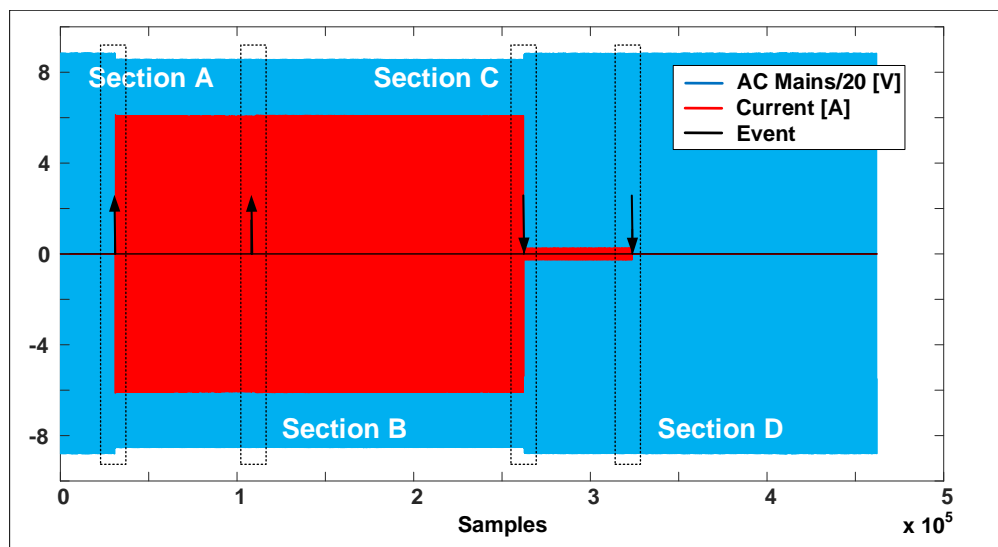


Figure 11. Jig acquisition of two loads: oil heater and led lamp. Section A and B are turn-on events (oil heater and lamp, respectively), while Section C and D are turn-off (oil heater and lamp, respectively). Such events are indicated by an arrow.

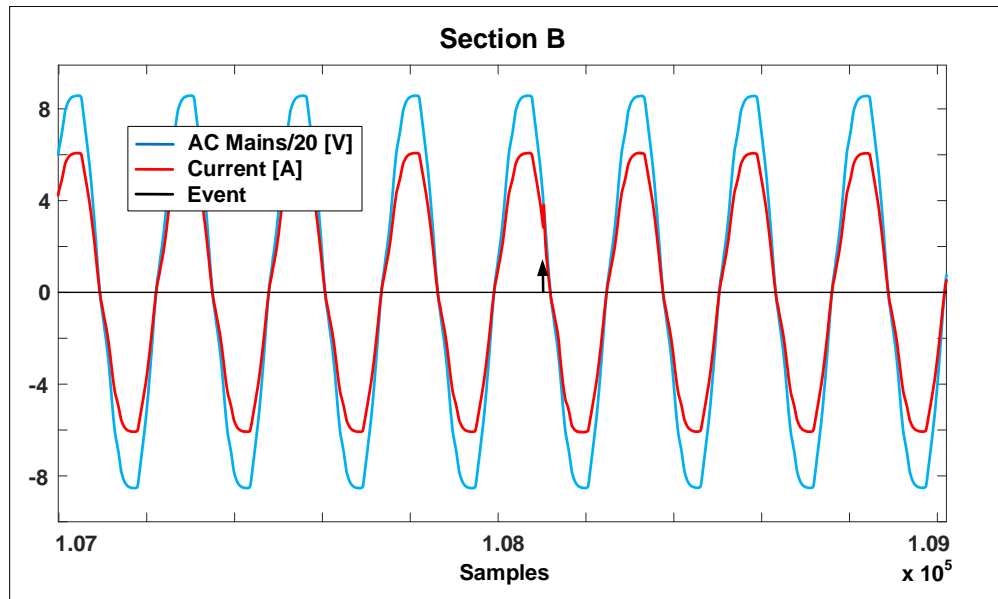


Figure 12. Jig acquisition of oil heater and led lamp—turn-on event Section B.

The average number of FPs and FNs per waveform are the lowest for HCApP and HAND for all datasets. For instance, in the HCApP method, there is less than one false positive per waveform on average for all datasets. DWT, KF, and VEC present a relatively high number of FPs and FNs, achieving more than three FPs per waveform. This is justified by the fact that these methods are more sensitive to small waveform variations since they are more appropriate for detecting transients [8].

Similar results are observed for average distances between actual events and their detections. The mean distances in semi cycles in terms of ON events, on average, is the lowest for DWT, but similar results are observed for HAND and HCApP. For OFF events, HAND and HCApP presented the lowest values, indicating that those two agents, besides presenting higher accuracy for detecting ON and OFF events, also have higher accuracy in the location of those events. The average distances in semi cycles for OFF events for DWT and KF are not presented, since these methods do not detect OFF events.

Regarding computational time, the results are presented in Table 7. Such results were generated using a start-stopwatch timer in MATLAB running on a Windows-PC with a 2.3 GHz Intel Core i7 processor, 8 GB RAM, running only MATLAB during this measurement. Additionally, to generate these results, the first load of the COOLL database was used, since the entire base is composed of acquisitions of only one load with waveforms of a six-seconds duration. For the SLIT database, load 417 was used, as this acquisition consists of five different loads with a 16 s duration. For the LIT-SYN database, load 1681 was used; this acquisition has nine ON events, nine off events, and a duration of 40 s; hence, the extraction and classification must occur nine times in the same acquisition.

Table 7. Results for Detection Agents (time).

Method	COOLL (s)	SLIT5 (s)	LIT-SYN (s)
DWT	0.22	0.08	0.28
HAND	0.11	0.09	0.24
HCApP	0.15	0.13	0.27
KF	0.26	0.09	1.21
VEC	0.05	0.04	0.07

In general, VEC presents the lowest times to process entire waveforms for all datasets. HAND and HCApP present similar performances, with results in the order of 150–270 ms. As KF involves several matrix multiplications, which significantly increases the computational time, its time results are

relatively higher than other methods. In some cases, particularly for LIT-SYN, the use of such method may be impracticable due to times superior than one second to process one waveform. DWT also presents higher time results, superior than HAND and HCApP methods.

5.2. Combination of Detection Agents

The two methods of combining the individual results from Event Detection Agents, as described in Section 3.2, were implemented and evaluated using the same metrics that were used to evaluate individual agents (Section 5.1). The results are presented in Table 8, below.

Table 8. Combination of detection agents.

Method	Metric	COOLL	SLIT5	LIT-SYN-1	LIT-SYN-2	LIT-SYN-3	LIT-SYN-8
HCApP	PC_{on}	95.2	92.4	94.7	88.7	82.8	51.9
	PC_{off}	99.6	88.1	94.9	85.3	89.6	56.1
	AFPs	0.1	0.0	0.5	0.5	0.5	0.5
	AFNs	0.1	0.7	0.1	0.5	0.8	8.2
	D_{on}	0.7	0.8	0.5	0.5	0.6	0.6
	D_{off}	0.9	0.9	0.6	0.5	0.5	0.6
Weighted Combination	PC_{on}	99.5	97.8	97.8	96.3	91.5	64.9
	PC_{off}	96.1	90.5	96.2	92.5	84.5	60.5
	AFPs	1.6	1.9	2.7	5.1	1.9	6.1
	AFNs	0.1	0.4	0.1	0.1	0.7	3.9
	D_{on}	0.9	0.7	1.3	1.2	1.0	1.0
	D_{off}	1.1	0.6	0.5	0.7	0.6	0.9
Dilation Erosion	PC_{on}	95.4	94.1	99.5	95.1	84.4	57.9
	PC_{off}	99.6	89.3	95.4	90.2	91.3	58.3
	AFPs	0.6	0.1	1.4	2.9	1.4	1.6
	AFNs	0.1	0.6	0.1	0.3	0.7	7.5
	D_{on}	0.7	0.8	0.6	0.6	0.6	0.7
	D_{off}	0.9	0.9	0.6	0.7	0.7	0.7

Note: The best results for each dataset, PC_{on} and PC_{off} metrics are highlighted in bold and underlined.

Both methods presented comparable results and improved the results over any single method when observing the metrics as a whole. PC_{on} and PC_{off} had improvements of up to 13% and 7.2%, respectively. A different perspective is to observe that the inaccuracies of ON and OFF detections were reduced from 17.2% (100–82.8%) to 8.5% (100–91.5%) when observing PC_{on} from LIT-SYN-3 for the results of HCApP and Weighted Combination. This indicates that the inaccuracies were reduced to less than a half.

5.3. Feature Extraction and Classification

In terms of feature extraction and classification methods, the average results (using the accuracy discussed in Section 4.3) are presented in Table 9, for the ten different training and test sets combinations presented in Section 4.5.

Table 9. Accuracy of Extraction and Classification Agents.

Ext.	Clas.	COOLL (%)	SLIT5 (%)	LIT-SYN -1 (%)	LIT-SYN -2 (%)	LIT-SYN -3 (%)	LIT-SYN -8 (%)	LIT-SYN -All (%)	Avg. (%)
V-I	LDA	98.3	87.8	99.3	98.9	89.3	85.7	72.7	90.3
	k-NN	96.4	75.5	99.0	99.7	96.7	93.9	94.6	93.7
	SVM	96.3	91.1	98.1	98.4	97.5	92.5	93.3	95.3
	DT	98.4	92.3	98.6	99.4	95.6	91.0	95.8	95.9
	ENS	99.6	94.8	99.5	99.3	99.5	92.8	95.7	97.3
Prony	LDA	88.8	72.1	89.9	95.7	89.2	81.0	83.7	85.8
	k-NN	84.6	53.9	88.2	94.6	86.3	83.3	82.8	82.0
	SVM	90.1	73.9	70.4	67.6	60.8	62.2	80.4	72.2
	DT	87.8	89.2	88.8	96.5	89.0	81.0	83.6	88.0
	ENS	98.4	93.7	96.1	98.5	95.2	90.8	94.4	95.3
DWT	LDA	91.5	86.3	82.8	70.0	61.1	60.6	60.4	73.2
	k-NN	97.2	92.5	95.3	87.8	85.5	75.2	94.2	89.7
	SVM	93.1	90.4	93.2	88.7	83.6	77.8	95.1	88.8
	DT	96.2	93.5	92.2	90.1	90.4	90.2	90.3	91.8
	ENS	100.0	97.6	99.3	97.2	94.5	95.2	95.1	97.0
VEC	LDA	85.2	81.1	85.4	79.9	69.8	70.0	70.1	77.4
	k-NN	90.2	85.3	91.1	60.2	49.8	55.3	90.1	74.6
	SVM	90.2	84.3	91.1	80.3	85.2	80.4	90.8	86.0
	DT	94.2	89.5	85.6	90.7	91.2	92.0	85.8	89.9
	ENS	98.2	91.9	93.5	90.3	90.2	88.4	90.7	91.9
ENV	LDA	95.3	89.2	91.4	99.3	79.5	81.2	70.1	86.5
	k-NN	97.2	89.2	95.6	99.2	96.4	96.1	93.2	95.2
	SVM	98.1	92.2	96.1	99.4	94.5	95.3	91.5	95.3
	DT	97.3	96.2	95.2	99.6	94.2	91.1	93.4	95.2
	ENS	97.5	96.1	97.5	99.4	95.2	95.3	95.6	96.5

Note: The best average results are highlighted in bold and underlined.

As one can observe, the best classifier, on average, is ENS for all feature extractors and all datasets (in bold), with results superior to 95% for most extractors, whilst LDA presented the lowest average accuracies. DWT and V-I, when combined with ENS, presented the best average results. However, ENV and Prony have similar performances.

As observed for detection methods, the best results are obtained for datasets with single loads (LIT-SYN-1 and COOLL)—above 99% for DWT and V-I combined with ENS. Additionally, since LIT-SYN provides combinations of single, two, three, and eight loads, it is possible to observe that the performance of feature extraction and classification methods decreases, as the number of loads increases, being the lowest performances observed for LIT-SYN-8. Furthermore, there are loads with a high difference in rated power, which comprises the proper characterization of features. LED Panel, for instance, presents an average individual class performance around 50% for LIT-SYN subsets.

LIT-SYN-All shows the results when all the loads (single, two, three, and eight) are used to train and test the models. Although the performance is superior to the cases of only multiple loads (around 95% for DWT and V-I combined with ENS), it is still inferior to the case of single loads (around 99% for DWT and V-I combined with ENS). These results emphasize the importance of including different single and multiple loads to build the signature extractor and classifier of a NILM approach.

Classification performance on SLIT5 presents relatively inferior results, this is explained by the combination of loads with different rated power, but similar signatures. For instance, a circuit with a resistor and an inductor and another with only a resistor have similar signatures, making it difficult to correctly discriminate between these two loads. The same is observed for other loads, such as a full-bridge rectifier with R and RL loads.

As LIT-SYN provides combinations of single, two, three, and eight loads, we also evaluate the generalization of training procedures, as discussed in Section 4.5. We train models with single and

three loads and evaluate the classification performance of the model with two, three, and eight loads for each feature extractor and classifier, as presented in Table 10.

Table 10. Accuracy of Extraction and Classification Agents (Generalization).

Ext.	Datasets	LDA (%)	k-NN (%)	SVM (%)	DT (%)	ENS (%)	Avg. (%)
V-I	LIT-SYN-1 to LIT-SYN-2	76.6	53.6	51.5	87.5	86.6	71.2
	LIT-SYN-1 to LIT-SYN-3	64.1	46.9	46.6	61.8	62.2	56.3
	LIT-SYN-1 to LIT-SYN-8	40.5	28.3	26.5	44.9	42.1	36.4
	LIT-SYN-3 to LIT-SYN-8	76.2	67.4	78.1	74.1	76.9	74.5
Prony	LIT-SYN-1 to LIT-SYN-2	71.1	65.3	62.7	50.2	83.4	67.1
	LIT-SYN-1 to LIT-SYN-3	51.8	53.2	44.8	35.5	59.4	48.9
	LIT-SYN-1 to LIT-SYN-8	31.4	40.2	34.8	25.5	45.3	35.4
	LIT-SYN-3 to LIT-SYN-8	51.3	55.3	63.6	50.2	64.2	56.9
DWT	LIT-SYN-1 to LIT-SYN-2	87.4	93.1	90.2	75.4	93.5	87.9
	LIT-SYN-1 to LIT-SYN-3	61.9	76.6	58.7	39.8	73.2	62.1
	LIT-SYN-1 to LIT-SYN-8	34.3	62.1	61.8	18.4	61.7	47.7
	LIT-SYN-3 to LIT-SYN-8	52.2	73.4	70.6	78.3	90.3	72.9
VEC	LIT-SYN-1 to LIT-SYN-2	68.6	78.2	80.1	77.5	79.9	76.9
	LIT-SYN-1 to LIT-SYN-3	56.5	70.0	70.1	51.5	65.7	62.7
	LIT-SYN-1 to LIT-SYN-8	49.7	65.2	50.7	42.2	52.1	51.9
	LIT-SYN-3 to LIT-SYN-8	70.4	80.8	60.2	78.4	81.6	74.3
ENV	LIT-SYN-1 to LIT-SYN-2	74.1	80.1	80.6	77.5	80.1	78.5
	LIT-SYN-1 to LIT-SYN-3	65.4	69.1	60.3	54.3	61.5	62.1
	LIT-SYN-1 to LIT-SYN-8	53.4	66.4	62.2	39.6	59.0	56.1
	LIT-SYN-3 to LIT-SYN-8	83.2	82.9	86.5	80.4	84.9	83.6

Note: The best results are highlighted in bold and underlined.

When single loads are used to train the model, and multiple loads (two, three, and eight) are used to test, the performance decreases as the number of multiple loads increases. However, it decreases more severely compared to the previous case, Table 9, where data from multiple loads were also included in the training. In this case, accuracy declines reaching values below 60%, particularly in the scenario LIT-SYN-1 to LIT-SYN-8. When multiple loads are included in the training, LIT-SYN-3 to LIT-SYN-8, the performance increases, but it is still considerably below the results from Table 9. Such a result emphasizes the importance of training and testing with data including multiple loads since the signature of single loads is highly modified when multiple loads are simultaneously turned-on, as in an electrical bar.

By analyzing each feature extractor and classifier agent, DWT presents the best average results for different classifiers, with the best results for ENS (in bold). ENV presents better average results for different classifiers. Prony and V-I features are the most influenced by multiple loads in the test, achieving performances below 40% on average. This is justified by the deformations caused in the form of V-I trajectory, resulting from the subtraction of the current. DT is the most affected classifier with multiple loads in the test, with an average performance below 35% for different feature extractors in the LIT-SYN-1 to LIT-SYN-8 case.

Results for extraction and classification agents in terms of computational complexity are presented in Tables 11 and 12, respectively.

Table 11. Execution times for Extraction Agents.

Method	COOLL (s)	SLIT5 (s)	LIT-SYN (s)	Average (s)
DWT	0.11	0.57	0.81	0.49
V-I	0.05	0.13	0.23	0.14
ENV	0.09	0.60	0.66	0.45
Prony	0.03	0.24	0.18	0.15
VEC	0.004	0.50	0.01	0.17

Table 12. Execution times for Classification Agents.

Extractor	Classifier	COOLL (ms)	SLIT5 (ms)	LIT-SYN (ms)
V-I	ENS	19.0	268.2	12.5
	LDA	4.7	3.6	4.1
	k-NN	4.9	4.9	5.4
	DT	2.2	2.4	2.7
	SVM	229.0	106.1	91.8
Prony	ENS	252.5	259.7	298.3
	LDA	4.4	4.6	3.8
	k-NN	4.1	4.8	6.1
	DT	2.0	2.4	3.1
	SVM	227.8	108.1	90.4
DWT	ENS	9.8	119.7	35.5
	LDA	4.4	3.8	4.4
	k-NN	3.9	4.0	5.1
	DT	2.3	2.3	2.1
	SVM	227.7	110.8	92.8
VEC	ENS	251.8	32.4	286.0
	LDA	4.8	3.8	3.9
	k-NN	3.9	4.4	5.3
	DT	1.8	2.3	2.6
	SVM	224.4	102.5	9.6
ENV	ENS	8.4	241.5	10.8
	LDA	5.4	3.9	4.2
	k-NN	7.4	4.3	5.6
	DT	4.0	2.6	1.3
	SVM	251.4	103.3	92.4

VEC presents the lowest time results for COOLL and LIT-SYN, while V-I presents the lowest time results for SLIT5. On average, Prony and V-I present the lowest time results, and DWT is the most computationally costly extraction method. In terms of classification, DT presents the lowest execution times for all extraction agents, followed by k-NN and LDA. SVM is the most computationally costly classification method, and ENS presents relatively higher times for V-I, Prony, and VEC. Considering that the classifier that presented the best accuracy results was EVM (Table 9), the best execution times for ENS were obtained by the ENV features extractor.

5.4. Combination of Extraction and Classification Agents

Results for the combination of Extraction and Classification Agents are presented in Table 13. In this table, the results for the best average individual agent are also presented for each dataset for a comparison reference. In the combination, all the feature extraction agents were selected, combined with ENS—the best average classifier for all datasets and extraction methods.

Table 13. Combination of Extraction and Classification Agents.

Datasets	Best Single Agent (%)	Combined Agents (%)
COOLL	100.0 (DWT)	100.0
SLIT5	97.6 (DWT)	99.7
LIT-SYN-1	99.5 (V-I)	99.8
LIT-SYN-2	99.4 (ENV)	100.0
LIT-SYN-3	99.5 (V-I)	99.8
LIT-SYN-8	95.3 (ENV)	99.8
LIT-SYN-T	95.7 (V-I)	99.8
LIT-SYN-1 to LIT-SYN-2	93.5 (DWT)	99.1
LIT-SYN-1 to LIT-SYN-3	73.2 (DWT)	79.3
LIT-SYN-1 to LIT-SYN-8	61.7 (DWT)	71.3
LIT-SYN-3 to LIT-SYN-8	90.3 (DWT)	90.7

The accuracy improves for most datasets or, at least, remains the same (in COOLL dataset, for instance). As the number of combined loads increases, the accuracy of combined agents increases more significantly. The same is observed for generalization, increasing 9.4% for *LIT-SYN-1 to LIT-SYN-8*, which was the worst of the cases previously analyzed.

5.5. Discussion

The main contributions that were presented are discussed and summarized in this section.

The proposed Half Cycle Apparent Power (HCApP) considers the Apparent Power, Active Power, and Reactive Power, calculated on every mains semi cycle, to identify load events. Its performance is similar to HAND and generally supersedes the remaining EDAs. Among its strengths is the low rate of False Positives.

The proposed Vectorization method aims at an efficient representation of a waveform as a vector of complex numbers. Such representation is then used both for event detection as well as the features later required for load classification. As a classification method, it achieved comparable results to the remaining approaches but with an efficient use of the computational resources, thus, indicating its adequacy for an embedded solution.

The Power Envelope classification method is based on an initial classification of the power envelope waveform into five classes, followed by the extraction of features that are specific for each class. The best results of individual classification agents in the scenario of eighth simultaneous loads (LIT-SYN8) were achieved by the Power Envelope combined with k-NN, SVM, and Ensemble, resulting in 95.3% to 96.1% accuracy.

The Weighted Combination is used both to combine the individual results of event detection agents as well as from classification agents. The results attained from this method provided improvements, in both situations, over any individual result. Likewise, the Dilation Erosion method for combining event detection agents produced improvements over individual agents.

Finally, we experimented with a technique where the feature extraction, and the classification agents are trained in a single load scenario, and the trained system has to classify the loads in a multiple concurrent loads scenario. It is our understanding that this is a more realistic situation, in the end-user premises than training in a multiple load scenario. Such results indicate that this is a challenging scenario that must be further examined.

6. Conclusions

The relevance of NILM in the context of Electrical Energy Efficiency was characterized as a diagnostic and management support tool, hence, supporting the current worldwide research effort on energy efficiency. The authors' contribution to this research effort is sevenfold: a multi-agent architecture, event detection HCApP method, event detection and load classification by the Vectorization method, Power Envelope method for load classification, the Weighted Combination and

the Dilation-Erosion methods for combining the results of individual agents, and the technique of M.L. training with a single load to classify multiple loads. These contributions were summarized in Section 5.5, above.

Concerning the multi-agent architecture, its rationale is to combine the expertise of several event detection agents into a single result; likewise, the expertise of several load classification agents is combined into a single result. This approach resulted in an event detection performance improvement of up to 13% and 9.4% performance improvement for load classification. Furthermore, five event detection techniques were individually evaluated as well as five feature extraction techniques and five classification methods. The individual results were compared to the combined results, thus, providing the baseline for quantification of the performance improvement of the proposed architecture, as well as to support the characterization of the strengths and weaknesses of each individual agent.

A further contribution of this research includes the comparison of the technologies and methods used by detection agents, feature extraction agents, and classification agents. Although the results indicate that an actual implementation of a multiagent NILM system would not require 15 agents (five for detection, five for feature extraction, and five for classification), we opted for reporting all these results in Section 5 to document our results.

Concerning the interpretation of the improvements reported by a multiagent architecture, an important consideration is that an improvement in accuracy from 95.3% to 99.8% could be interpreted merely as a 4.5% improvement and, hence, not being substantial. Such an improvement was reported in Table 13 line LIT-SYN-8. However, this result indicates that the errors in classification were reduced from 4.7% to 0.2%. This represents a classification error rate reduction over twenty-fold, which is a significant result of the proposed architecture.

Additionally, as part of our future work, we intend to evaluate the proposed multiagent architecture in other realistic situations, including prolonged tests (days or weeks) in different residential and work environments, as well as premises that make use of distributed generation.

Author Contributions: Conceptualization, A.E.L., D.P.B.R., C.R.E.L., and F.P.; methodology, B.M.M., E.O., H.C.A., C.R.E.L., and R.R.L.; software, B.M.M., H.C.A., D.P.B.R., F.P., L.d.S.N., L.T.L., and J.S.O.; validation, D.P.B.R., F.P., C.R.E.L., E.O., and A.E.L.; formal analysis, A.E.L., D.P.B.R., B.M.M., H.C.A., and R.R.L.; investigation, B.M.M., H.C.A., D.P.B.R., L.d.S.N., and L.T.L.; resources, A.E.L. and D.P.B.R.; data organization, R.R.L., L.d.S.N., and L.T.L.; writing—original draft preparation, A.E.L., D.P.B.R., F.P., and E.O.; writing—review and editing, D.P.B.R., J.S.O., R.B.d.S., F.P., B.M.M., H.C.A., C.R.E.L., and R.R.L.; supervision, D.P.B.R., J.S.O., R.B.d.S., and A.E.L.; project administration, D.P.B.R., J.S.O., R.B.d.S., and A.E.L.; funding acquisition, D.P.B.R. and A.E.L. All authors have read and agreed to the published version of the manuscript.

Funding: This study was fully financed by Agência Nacional de Energia Elétrica (ANEEL) and Companhia Paranaense de Energia Elétrica (COPEL) under the research and development program (project PD2866-0464/2017).

Acknowledgments: The authors would like to thank COPEL and ANEEL for the support and promotion in the research project PD2866-0464/2017.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; the authors affiliated to the funder company had the role of manuscript revision and evaluating the request for publication.

References

1. Zoha, A.; Gluhak, A.; Imran, M.A.; Rajasegarar, S. Non-intrusive Load Monitoring approaches for disaggregated energy sensing: A survey. *Sensors* **2012**, *12*, 16838–16866. [CrossRef] [PubMed]
2. Stone, P.; Veloso, M. Multiagent Systems: A Survey from a Machine Learning Perspective. *Auton. Robot.* **2000**, *8*, 345–383. [CrossRef]
3. IEA. *World Energy Outlook 2019*; Technical Report; International Energy Agency: Paris, France, 2019. Available online: <https://www.oecd-ilibrary.org/content/publication/caf32f3b-en> (accessed on 15 June 2020).
4. Efficiency Valuation Organization. *International Performance Measurement and Verification Protocol. In Concepts and Options for Determining Energy and Water Savings*; U.S. Department of Commerce: Oak Ridge, TN, USA, 2012.

5. ISO. *ISO Standard 50001: 2018 Energy Management Systems—Requirements with Guidance for Use*; ISO: Geneva, Switzerland, 2018.
6. Liang, J.; Ng, S.K.K.; Kendall, G.; Cheng, J.W.M. Load Signature Study—Part I: Basic Concept, Structure and Methodology. *IEEE Trans. Power Deliv.* **2010**, *25*, 870–878. [[CrossRef](#)]
7. Figueiredo, M.; Almeida, A.; Ribeiro, B. Home Electrical Signal Disaggregation for Non-intrusive Load Monitoring Systems. *Neurocomputing* **2012**, *96*, 66–73. [[CrossRef](#)]
8. Bollen, M.; Gu, I. *Signal Processing of Power Quality Disturbances*; John Wiley & Sons: Hoboken, NJ, USA, 2006.
9. Renaux, D.P.B.; Linhares, R.R.; Pottker, F.; Lazzaretti, A.E.; de Lima, C.E.E.; Coelho-Neto, A.; Campaner, M.H. Designing a Novel Dataset for Non-Intrusive Load Monitoring. In Proceedings of the VIII Brazilian Symposium on Computing Systems Engineering, Salvador, Brazil, 6–9 November 2018.
10. Nait-Meziane, M.; Ravier, P.; Abed-Meraim, K.; Lamarque, G.; Bunetel, J.C.; Raingeaud, Y. Electrical transient modeling for appliance characterization. *EURASIP J. Adv. Signal Process.* **2019**, *55*, 1–19. [[CrossRef](#)]
11. Nait-Meziane, M.; Ravier, P.; Lamarque, G.; Le Bunetel, J.C.; Raingeaud, Y. High accuracy event detection for Non-Intrusive Load Monitoring. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, New Orleans, LA, USA, 5–9 March 2017.
12. Dantas, P.; Sabino, W.; Batalha, M. Energy Disaggregation via Data Mining. In Proceedings of the 4th Brazilian Technology Symposium, Campinas, Brazil, 28 October 2018; pp. 541–546.
13. Ruano, A.; Hernandez, A.; Ureña, J.; Ruano, M.; García, J. NILM Techniques for Intelligent Home Energy Management and Ambient Assisted Living: A Review. *Energies* **2019**, *12*, 2203. [[CrossRef](#)]
14. Ukil, A.; Živanović, R. Adjusted Haar wavelet for application in the power systems disturbance analysis. *Digit. Signal Process.* **2008**, *18*, 103–115. [[CrossRef](#)]
15. Chang, H.H. Non-Intrusive Demand Monitoring and Load Identification for Energy Management Systems Based on Transient Feature Analyses. *Energies* **2012**, *5*, 4569–4589. [[CrossRef](#)]
16. Pottker, F.; Lazzaretti, A.; Renaux, D.; Linhares, R.; Erig Lima, C.; Ancelmo, H.; Machado Mulinari, B. Non-Intrusive Load Monitoring: A Multi-Agent Architecture and Results. In Proceedings of the 2nd European Conference on Electrical Engineering and Computer Science (EECS), Bern, Switzerland, 20–22 December 2018; pp. 328–334.
17. Iksan, N.; Sembiring, J.; Haryanto, N.; Supangkat, S.H. Appliances identification method of non-intrusive load monitoring based on load signature of V-I trajectory. In Proceedings of the International Conference on Information Technology Systems and Innovation, Bandung, Indonesia, 16–19 November 2015.
18. Lee, K.D.; Leeb, S.B.; Norford, L.K.; Armstrong, P.R.; Holloway, J.; Shaw, S.R. Estimation of variable-speed-drive power consumption from harmonic content. *IEEE Trans. Energy Convers.* **2005**, *20*, 566–574. [[CrossRef](#)]
19. Srinivasan, D.; Ng, W.S.; Liew, A.C. Neural-network-based signature recognition for harmonic source identification. *IEEE Trans. Power Deliv.* **2006**, *21*, 398–405. [[CrossRef](#)]
20. Lima, A.C.S.; Alves da Silva, A.P.; Nascimento, D. Noninvasive monitoring of residential loads. In Proceedings of the 2012 3rd IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe), Berlin, Germany, 14–17 October 2012; pp. 1–5.
21. Wang, A.E.L.; Chen, B.X.; Wang, C.G.; Hua, D.D. Non-intrusive load monitoring algorithm based on features of V-I trajectory. *Electr. Power Syst. Res.* **2018**, *157*, 134–144. [[CrossRef](#)]
22. Kim, J.; Le, T.T.H.; Kim, H. Nonintrusive Load Monitoring Based on Advanced Deep Learning and Novel Signature. *Comput. Intell. Neurosci.* **2017**, *2017*, 1–22. [[CrossRef](#)] [[PubMed](#)]
23. Chang, H.; Lian, K.; Su, Y.; Lee, W. Power-Spectrum-Based Wavelet Transform for Nonintrusive Demand Monitoring and Load Identification. *IEEE Trans. Ind. Appl.* **2014**, *50*, 2081–2089. [[CrossRef](#)]
24. Martins, J.F.; Lopes, R.; Lima, C.; Romero-Cadaval, E.; Vinnikov, D. A novel nonintrusive load monitoring system based on the S-Transform. In Proceedings of the 13th International Conference on Optimization of Electrical and Electronic Equipment, Brasov, Romania, 24–26 May 2012; pp. 973–978.
25. Gillis, J.M.; Alshareef, S.; Morsi, W.G. Nonintrusive Load Monitoring Using Wavelet Design and Machine Learning. *IEEE Trans. Smart Grid* **2016**, *7*, 320–328. [[CrossRef](#)]
26. Tabatabaei, S.; Dick, S.; Xu, W. Toward Non-Intrusive Load Monitoring via Multi-Label Classification. *IEEE Trans. Smart Grid* **2016**, *8*, 1–8. [[CrossRef](#)]
27. Machado Mulinari, B.; Linhares, R.; Campos, D.; Costa, C.; Ancelmo, H.; Lazzaretti, A.; Oroski, E.; Erig Lima, C.; Renaux, D.; Pottker, F. A New Set of Steady-State and Transient Features for Power Signature

- Analysis Based on V-I Trajectory. In Proceedings of the IEEE PES Innovative Smart Grid Technologies Conference, Gramado, Brazil, 15–18 September 2019; pp. 1–6.
28. Ancelmo, H.; Grando, F.; Machado Mulinari, B.; Costa, C.; Lazzaretti, A.; Oroski, E.; Renaux, D.; Pottker, F.; Erig Lima, C.; Linhares, R. A Transient and Steady-State Power Signature Feature Extraction Using Different Prony's Methods. In Proceedings of the 20th International Conference on Intelligent Systems Applications to Power Systems, New Delhi, India, 10–14 December 2019; pp. 1–7.
 29. Lin, Y.H.; Tsai, M.S. Development of an Improved Time-Frequency Analysis-Based Nonintrusive Load Monitor for Load Demand Identification. *IEEE Trans. Instrum. Meas.* **2014**, *63*, 1470–1483. [[CrossRef](#)]
 30. Zhukov, A.; Sidorov, D.; Foley, A. Random Forest Based Approach for Concept Drift Handling. *Communications in Computer and Information Science*, Yekaterinburg, Russia, 7–9 April 2016; pp. 69–77.
 31. Sidorov, D.; Tao, Q.; Muftahov, I.; Zhukov, A.; Karamov, D.; Dreglea, A.; Liu, F. Energy balancing using charge/discharge storages control and load forecasts in a renewable-energy-based grids. Chinese Control Conference, Guangzhou, China, 27–30 July 2019; pp. 6865–6870.
 32. Zhukov, A.; Tomin, N.; Kurbatsky, V.; Sidorov, D.; Panasetsky, D.; Foley, A. Ensemble methods of classification for power systems security assessment. *Appl. Comput. Inform.* **2019**, *15*, 45–53.
 33. Renaux, D.P.B.; Lima, C.R.E.; Pottker, F.; Oroski, E.; Lazzaretti, A.E.; Linhares, R.R.; Almeida, A.R.; Coelho, A.O.; Hercules, M.C. Non-Intrusive Load Monitoring: An Architecture and its evaluation for Power Electronics loads. In Proceedings of the IEEE International Power Electronics and Application Conference and Exposition (PEAC), Shenzhen, China, 4–7 November 2018; pp. 1–6.
 34. Ancelmo, H.C.; Grando, F.L.; Costa, C.H.; Mulinari, B.M.M.; Oroski, E.; Lazzaretti, A.E.; Pottker, F.; Renaux, D.P.B. Automatic Power Signature Analysis using Prony Method and Machine Learning-Based Classifiers. In Proceedings of the 2nd European Conference on Electrical Engineering and Computer Science, Bern, Switzerland, 20–22 December 2018.
 35. Renaux, D.P.B.; Linhares, R.R.; Pottker, F.; Lazzaretti, A.E.; de Lima, C.E.E.; Coelho-Neto, A.; Campaner, M.H. Designing a Novel Dataset for Non-Intrusive Load Monitoring. In Proceedings of the VIII Brazilian Symposium on Computing Systems Engineering, Salvador, Brazil, 6–9 November 2018.
 36. Ancelmo, H.; Machado Mulinari, B.; Pottker, F.; Lazzaretti, A.; de Paula Machado Bazzo, T.; Oroski, E.; Renaux, D.; Erig Lima, C.; Linhares, R. A New Simulated Database for Classification Comparison in Power Signature Analysis. In Proceedings of the 20th International Conference on Intelligent Systems Applications to Power Systems, New Delhi, India, 10–14 December 2019; pp. 1–7.
 37. Linhares, R.R.; Lima, C.R.E.; Renaux, D.P.B.; Pottker, F.; Oroski, E.; Lazzaretti, A.E.; Mulinari, B.M.M.; Ancelmo, H.C.; Gamba, A.; Bernardi, L.A.; et al. One-millisecond low-cost synchronization of wireless sensor network. In Proceedings of the VIII Brazilian Symposium on Computing Systems Engineering, Salvador, Brazil, 6–9 November 2018.
 38. Mallat, S. *A Wavelet Tour of Signal Processing*, 3rd ed.; Academic Press: Orlando, FL, USA, 2008.
 39. Lam, H.Y.; Fung, G.S.K.; Lee, W.K. A Novel Method to Construct Taxonomy of Electrical Appliances Based on Load Signatures. *IEEE Trans. Consum. Electron.* **2007**, *53*, 653–660. [[CrossRef](#)]
 40. Hassan, T.; Javed, F.; Arshad, N. An Empirical Investigation of V-I Trajectory Based Load Signatures for Non-Intrusive Load Monitoring. *IEEE Trans. Smart Grid* **2014**, *5*, 870–878. [[CrossRef](#)]
 41. Lam, H.; Ting, K.; Lee, W.K.; Fung, G. An analytical understanding on voltage-current curve of electrical load. In Proceedings of the Conference on Electrical Engineering, Bialystok, Poland, 13–17 September 2006.
 42. Hudgins, B.; Parker, P.; Scott, R.N. A new strategy for multifunction myoelectric control. *IEEE Trans. Biomed. Eng.* **1993**, *40*, 82–94. [[CrossRef](#)] [[PubMed](#)]
 43. Meziane, M.N.; Hacine-Gharbi, A.; Ravier, P.; Lamarque, G.; Bunetel, J.L.; Raingeaud, Y. Electrical Appliances Identification and Clustering using Novel Turn-on Transient Features. In Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods, Porto, Portugal, 24–26 February 2017; pp. 647–654.
 44. Marple, S.L. *Digital Spectral Analysis: With Applications*; Prentice-Hall: Upper Saddle River, NJ, USA, 1986.
 45. Cherkassky, V.S.; Mulier, F. *Learning from Data: Concepts, Theory, and Methods*, 1st ed.; John Wiley & Sons: New York, NY, USA, 1998.
 46. Vapnik, V.N. *Statistical Learning Theory*; Wiley-Interscience: Hoboken, NJ, USA, 1998.

47. Picon, T.; Nait Meziane, M.; Ravier, P.; Lamarque, G.; Novello, C.; Le Bunetel, J.C.; Raingeaud, Y. COOLL: Controlled On/Off Loads Library, a Public Dataset of High-Sampled Electrical Signals for Appliance Identification. *arXiv* **2016**, arXiv:1611.05803.
48. Snoek, J.; Larochelle, H.; Adams, R.P. Practical Bayesian Optimization of Machine Learning Algorithms. In Proceedings of the 25th International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 2951–2959.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).