SP4LC: A Method for Recognizing Power Consumers in a Smart Plug

Dániel István Németh[a] and Kálmán Tornai[b]

Pázmány Péter Catholic University, F. of Information Technology and Bionics, 50/a Práter str, 1083 Budapest, Hungary

Keywords: Smart Homes, Consumer Recognition, Load Classification, Smart Plugs, Smart Grid, Edge Computing, Machine Learning.

Abstract: Electrical load classification is a crucial task related to balance management in smart electrical grids. The classification algorithms and methods enable the smart system to schedule and adjust the grid load to meet the production capabilities. Fast decision-making is key to creating a responsive grid, especially when grid operators utilize renewable energy sources such as wind or solar power. This paper proposes new approach Smart Plug for Load Classification, an active load classification system to recognize the connected devices based on their load with less than 10 seconds of measurement data. Also, we propose an IoT-capable measurement device and show the collected data's classification results with multiple methods suited for both Edge Computing and Cloud computation.

1 INTRODUCTION

With the rise of renewable resources in electrical grids, load balancing became a more complex task. Unlike traditional power plants, renewable power production levels cannot be controlled in most cases. One solution to this challenge of balancing electricity production and consumption levels is controlling the demand side. This, however, requires knowledge of the load and the ability to control them. As both electricity production and consumption levels can change rapidly, fast decision-making is required to create a responsive grid. This paper presents the Smart Plug for Load Classification (SP4LC), an active load classification system capable of recognizing the connected load based on its characteristic response to manipulating its power signal. The data collected in less than 10 seconds is enough to identify the connected load accurately. We show multiple approaches to classify the data measured by our prototype device. The classification method depends on the use case of the system. To enable on-device classification for rapid response, less data is better and a method that requires less computational power. In edge computing situations, fewer restrictions apply. With Cloud-based solutions, there are virtually no restrictions in terms of computational power.

The rest of the paper is structured as follows. Section 2 shows a summary of related publications. In Section 3, we present the hardware prototype and measurement methodology. Section 4 shows the Support Vector Machines classification results. In Section 5, we introduce measurement profiles for optimizing the data collection depending on the requirements, followed by Section 6 containing the Fully Connected and Convolutional Neural Network classification results. The conclusions are presented in Section 7.

2 RELATED WORK

Electrical load classification is an essential part of the operation of smart grids. With the adoption of renewable energy sources, load balancing has become a critical part of the operation of the grid (Jaradat et al., 2014). In order to actively balance the system by controlling the load, knowledge is required about the types of loads connected to the grid. In (Jaradat et al., 2014), a Demand-Side Management system is shown as a linear programming problem. The goal was to maximize the utilization of renewable energy sources and minimize the price of the purchased electricity from the grid.

Electrical load classification can be done intrusively, and non-intrusively (Ridi et al., 2014). Non-Intrusive Load Monitoring can be achieved using a Smart Meter. The Smart Meter can communicate with the grid provider to help the operation of the Smart

[a] https://orcid.org/0000-0002-5740-284X
[b] https://orcid.org/0000-0003-1852-0816

Németh, D. and Tornai, K.
SP4LC: A Method for Recognizing Power Consumers in a Smart Plug.
DOI: 10.5220/0010982800003203
In *Proceedings of the 11th International Conference on Smart Cities and Green ICT Systems (SMARTGREENS 2022)*, pages 69-77
ISBN: 978-989-758-572-2; ISSN: 2184-4968
Copyright © 2022 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

Grid. With Smart Meters, only the sum of all loads in the household is measured, so disaggregation of the load curve is necessary to learn about the individual loads. In Intrusive Load Monitoring, metering is done either for every load or each zone within the building.

Current smart plugs available on the market are not capable of load identification (Gomes et al., 2018). The user has to set up the basic properties and scheduling of the connector. The proposed system in (Gomes et al., 2018) uses environmental sensors to help determine if an electric load is needed. In (Gomes et al., 2019) a case study is shown how EnAPlugs can provide energy savings by using sensors to enable environmental awareness.

In (da S. Veloso et al., 2019), a system is shown which uses Electric Load Signature (ELS) to differentiate between loads. Measurements were done every second for one hour to collect the ELS data. Another possibility for faster data collection is to use the Voltage-Current curve of the load to determine the type of electric load connected (Du et al., 2016).

In (Petrović and Morikawa, 2017) load classification is achieved by using a bidirectional triode thyristor to manipulate the voltage supply of the load. An Arduino microcontroller was used to collect the measurement data and control the TRIAC. The microcontroller masked the voltage signal of the load between ratios of 10% and 95% with 5% steps. The other parameter used was the number of consecutive masking cycles between 1 and 20. The load current, voltage, and power were measured for each cycle of the AC signal. The measured power data was put into a matrix, and this matrix was the input of a Fully Connected Neural Network used for load classification. The classification accuracy was 96.5%, and each measurement took 45 seconds.

This paper presents a similar approach to (Petrović and Morikawa, 2017), but with several improvements in the prototype device, measurement speed, data collection, and classification methods.

3 NEW MEASUREMENT PROTOCOL AND PROTOTYPE

To measure the response of an electric load to the manipulation of the AC input voltage, a custom measurement device prototype was built. The prototype device is capable of cutting off the AC supply of the load, measuring the power characteristics of the device during the experiment, processing the data and sending the processed data to the connected computer. This section describes the measurement device prototype as well as the measurement method used for collecting data about the devices' characteristic response.

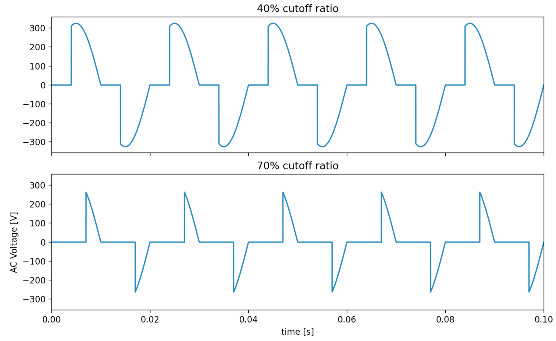

Figure 1: Voltage cutoff method with different cutoff ratios.

3.1 Hardware Configuration

The prototype device uses the ESP32 microcontroller. An off-the-shelf AC dimmer module is used to control the masking of the AC signal. A transformer and a current transformer are used to measure the voltage and current of the load. The off-the-shelf dimmer had zero-crossing detection capabilities so the measurement could be precisely synchronized to the AC voltage curve. The main advantages of the ESP32 over the Arduino microcontroller used in (Petrović and Morikawa, 2017) are the faster CPU frequency, the 12-bit ADC, and the dual cores so that one core can measure while the other core processes and sends the data to the computer. In each period of the 230V 50Hz AC signal, the ESP32 measures 279-280 ADC values from the transformer and the current transformer. The period of the 50Hz AC signal is 20ms. This includes two zero-crossing events.

3.2 Measurement Method

Using the dimmer, the ESP32 cuts the voltage supply of the load after a zero-crossing event for a specific time period. This time period is given as the ratio of cutoff time and the time between two zero-crossing events (10ms) as demonstrated by Figure 1. The device uses cutoff ratios between 10% and 75% with a 5% step. For each cutoff ratio, the device measures 20 AC periods. Data is calculated for each period. After a measurement with a cutoff ratio is completed, the device waits 16 AC periods before proceeding to measure with the following cutoff ratio. This procedure allows the load to receive uninterrupted power. The measurement starts with a 10% cutoff ratio, and the cutoff ratio is increased by 5% until 75%. The time of the entire measurement is 488 AC cycles which are 9.76s.

Current

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
10%	0.086	0.093	0.083	0.087	0.086	0.09	0.076	0.074	0.069	0.082	0.07	0.087	0.082	0.076	0.083	0.077	0.077	0.083	0.078	0.077
15%	0.073	0.085	0.08	0.079	0.081	0.089	0.086	0.084	0.077	0.078	0.084	0.078	0.079	0.08	0.077	0.079	0.078	0.089	0.076	0.08
20%	0.072	0.08	0.081	0.086	0.082	0.08	0.079	0.077	0.082	0.086	0.087	0.084	0.065	0.088	0.081	0.068	0.081	0.077	0.075	0.076
25%	0.074	0.078	0.08	0.068	0.076	0.079	0.078	0.078	0.073	0.076	0.083	0.083	0.08	0.082	0.086	0.079	0.086	0.086	0.081	0.079
30%	0.066	0.075	0.078	0.082	0.076	0.077	0.067	0.075	0.082	0.08	0.075	0.084	0.077	0.073	0.089	0.075	0.077	0.075	0.081	0.091
35%	0.077	0.069	0.077	0.076	0.085	0.066	0.073	0.083	0.069	0.078	0.079	0.077	0.084	0.072	0.069	0.083	0.079	0.082	0.08	0.067
40%	0.073	0.086	0.077	0.087	0.074	0.074	0.069	0.07	0.075	0.079	0.071	0.071	0.079	0.084	0.075	0.084	0.068	0.077	0.071	0.074
45%	0.082	0.082	0.077	0.075	0.084	0.077	0.08	0.08	0.082	0.086	0.077	0.081	0.078	0.076	0.075	0.083	0.087	0.084	0.074	0.089
50%	0.107	0.103	0.097	0.098	0.101	0.105	0.099	0.105	0.099	0.104	0.098	0.101	0.1	0.105	0.108	0.102	0.101	0.105	0.102	0.102
55%	0.126	0.119	0.118	0.115	0.116	0.116	0.121	0.118	0.12	0.124	0.122	0.117	0.118	0.109	0.122	0.112	0.119	0.118	0.116	0.12
60%	0.18	0.143	0.14	0.132	0.135	0.131	0.134	0.136	0.133	0.132	0.132	0.132	0.135	0.133	0.14	0.132	0.135	0.133	0.133	0.135
65%	0.149	0.189	0.179	0.181	0.183	0.182	0.174	0.184	0.183	0.187	0.177	0.181	0.18	0.184	0.177	0.188	0.181	0.181	0.177	0.188
70%	0.061	0.172	0.194	0.194	0.2	0.19	0.198	0.195	0.197	0.196	0.198	0.201	0.192	0.195	0.2	0.199	0.195	0.195	0.198	0.199
75%	0.026	0.101	0.244	0.264	0.249	0.268	0.244	0.268	0.255	0.26	0.267	0.256	0.262	0.265	0.254	0.255	0.266	0.268	0.255	0.263

Voltage

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
10%	216.2	221.2	220.7	220.7	220.9	221	220.9	221	221.1	221	220.5	220.8	220.9	220.5	220.5	220.6	220.8	220.8	221.2	221.3
15%	213.6	211.5	211.3	211.8	210.8	211.9	211.4	211.2	211.6	211.2	211.6	211.6	211.9	211.7	211.2	211.6	211.6	212	211.2	211.6
20%	199.9	202.7	202.5	202.1	201.9	202.7	202.2	202.3	202.8	202.6	202.4	202.9	203	202.1	202.4	202	202.6	202.7	202.8	202.9
25%	202.1	197.1	197.1	196.7	196.6	196.9	196.7	197.2	196.9	197	196.7	197	196.5	196.9	196.7	196.6	196.5	196.4	197.1	196.8
30%	194.8	193.8	193.8	194.3	193.9	193.8	193.9	193.6	193.8	194.2	193.7	193.8	194.2	193.8	193.6	193.9	194.1	193.8	193.8	194.2
35%	191.6	190.1	190.1	190.3	190.6	190.5	190.7	190.7	190.6	190.9	190.5	190.3	190.5	191.2	190.6	190.5	190.4	190.5	190.9	190.7
40%	186.4	188.7	188.5	188.3	188	187.7	188.8	187.9	188.2	188.4	187.9	188.1	188	188	188.2	187.9	188	188	187.9	187.7
45%	180.6	183.8	183.8	183.6	183.9	183.6	183.7	182.3	182.6	182.6	183.8	183.8	183.6	183.5	183.3	183.3	183.1	182.3	182.3	183.7
50%	169.7	174.6	174.8	174.6	175.1	173.8	174.6	174.3	174.7	174.7	174.3	174.2	174.1	174.3	173.8	174.4	174.3	174.6	174.2	173.9
55%	152.2	162.1	161.9	162	162.4	162	162.3	161.1	161.8	161.9	162.6	161.9	161.3	162.2	161.3	161.1	160.5	161	161.1	161.6
60%	129.5	145.8	145.2	144.8	146	145	144.5	144.2	146	145.2	144.5	145.9	144.9	144.5	145.8	144.7	143.9	145.4	144.6	144
65%	108	128	125.7	126.6	127.5	126.8	125.9	126.6	127.3	126.5	126	126.5	127.1	126.8	126	126.5	127.2	126.5	126.2	127.2
70%	101.5	102.7	102.5	101.7	102.6	101.6	102.7	102.3	102	102	101.7	102.6	102.4	102.2	102.7	102.6	102.7	102.6	102.2	101.8
75%	74.57	77.11	76.96	78.29	77.45	77.8	77.06	77.36	77.23	76.4	77.44	77.27	77.23	78.52	77.18	77.31	77.07	78.5	78.09	77.71

Power

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
10%	6.006	9.493	7.875	8.076	8.429	9.156	7.46	6.368	6.109	6.658	6.381	6.842	6.2	6.814	8.355	7.406	7.269	7.556	6.199	7.591
15%	6.488	7.498	7.252	7.333	4.748	8.259	9.662	8.148	6.878	6.99	7.522	6.402	7.309	7.254	7.464	6.513	7.046	8.293	7.011	5.898
20%	5.495	6.475	7.233	7.417	7.094	7.757	7.096	7.021	7.08	6.284	8.1	7.656	4.837	6.678	7.499	5.133	6.424	6.855	7.242	7.097
25%	6.95	6.558	7.369	5.489	6.065	7.426	5.827	7.394	6.887	7.641	8.269	6.464	5.558	7.562	8.104	6.431	8.54	6.839	7.32	6.044
30%	6.793	6.765	6.65	7.929	7.169	6.172	6.249	6.225	7.59	8.122	7.257	7.343	6.941	5.852	8.017	7.314	6.218	6.201	8.459	7.331
35%	7.726	7.022	6.766	6.848	8.795	6.795	7.651	7.024	4.382	7.502	7.7	7.241	7.129	6.596	6.3	7.646	6.512	7.619	8.418	6.777
40%	7.326	8.39	8.597	8.676	6.074	5.98	6.317	6.658	8.086	6.882	6.492	7.598	6.466	7.929	6.965	8.225	4.953	7.891	6.654	6.977
45%	7.041	7.46	7.126	5.856	6.82	6.777	6.183	7.462	7.198	8.036	6.117	7.618	6.016	6.361	7.355	8.229	7.813	6.773	6.606	7.79
50%	7.64	7.8	6.641	7.267	8.351	7.805	7.828	7.316	7.746	8.149	7.896	7.643	7.252	7.962	7.585	7.89	7.303	7.897	8.135	7.913
55%	7.465	7.962	7.683	7.707	7.344	7.69	7.433	7.783	7.562	8.52	8.434	7.41	8.441	7.078	8.805	7.205	7.756	7.999	7.332	8.038
60%	7.306	8.868	8.364	7.387	7.799	6.465	7.804	8.202	8.176	7.995	6.889	7.444	7.965	6.973	9.249	6.963	7.48	7.156	8.459	7.418
65%	5.025	9.032	7.669	8.423	8.445	8.258	7.399	8.587	8.334	8.432	8.285	8.149	8.169	8.316	7.874	8.525	8.801	7.761	7.994	8.867
70%	1.318	6.626	8.686	8.054	8.127	8.021	7.914	8.306	8.237	8.332	8.037	8.68	8.115	8.115	8.483	8.111	8.252	8.236	8.149	7.253
75%	0.034	2.526	8.353	9.252	8.39	9.19	8.336	9.248	8.856	8.512	9.018	8.946	8.915	9.15	8.796	8.69	9.243	9.735	9.043	9.24

Figure 2: Measurement matrices for a USB charger. The vertical axis shows the cutoff ratio, and the horizontal shows the measurements for that cutoff ratio in sequence.

For each AC period, the device measures Voltage($U[k]$) and Current($I[k]$) values as fast as the ESP32 ADC allows. From this, for each AC period, three values are calculated. The RMS Voltage and Current:

$$V_{RMS} = \sqrt{\frac{1}{n} \cdot \sum_{k=1}^{n} U[k]^2} \;, I_{RMS} = \sqrt{\frac{1}{n} \cdot \sum_{k=1}^{n} I[k]^2} \quad (1)$$

And the Real Power:

$$P = \frac{1}{n} \cdot \sum_{k=1}^{n} U[k]I[k] \quad (2)$$

The calculations are done on the ESP32. The data is sent to the computer, where a matrix is constructed for the Voltage, Current, and Power measurements. An example of this can be seen in Figure 2.

3.3 Measured Devices

Common household devices were measured with the prototype device. The following list contains the labels used in the paper and the device description.

- ipad10W - A 10W Apple USB adapter for iPad
- usbapple5V1A - A 5W Apple USB adapter
- usb5V1A - A 5W generic USB adapter
- batterycharger4A - A four ampere "smart" lead-acid battery charger
- batterycharger800mA - An 800mA traditional lead-acid battery charger
- fan - A fan
- hairdryer - A hairdryer
- incandescentbulb - An incandescent light bulb
- irlamp - An infrared heat lamp
- laptop - A laptop charger charging the laptop
- monitor - An LCD screen
- solderingiron - A soldering iron

At least 250 measurements were taken with every device. For all classification methods, only the Power matrix was used. Only those measurements were used, where the average of the Power matrix was greater than 1.5W.

4 PERFORMANCE OF SVM

Support Vector Machine classification requires feature extraction for fast computation and accurate results. Choosing these features is crucial in order to separate the different loads. The following ten features were selected to be used for the SVM classification:

- AVG: mean of the matrix elements
- STDEV: standard deviation of the matrix elements
- ROWAVG: mean of the standard deviations of matrix rows
- ROWSTD: standard deviation of the standard deviations of matrix rows
- COLUMNAVG: mean of the standard deviations of matrix columns
- COLUMNSTD: standard deviation of the standard deviations of matrix columns
- TOPLEFT: mean of the top left 2x2 submatrix divided by AVG
- BOTTOMLEFT: mean of the bottom left 2x2 submatrix divided by AVG
- TOPRIGHT: mean of the top right 2x2 submatrix divided by AVG
- BOTTOMRIGHT: mean of the bottom right 2x2 submatrix divided by AVG

Five of the feature values for the measured matrices can be seen in Figure 3. One can observe that the USB adapters have similar characteristics, and some devices can be separated from some of the other devices based on a single feature. These features change in time, as can be seen in Figure 4.

For the SVM classification, 30 samples from each class were enough to produce accurate predictions. A linear kernel was used. The average confusion matrix from 100 runs can be seen in Figure 5. It can be seen that most of the error comes from wrongly classifying a USB charger device. In most cases, differentiating between USB chargers is indifferent to the task of load classification, so in Figure 6, only one USB class was used.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

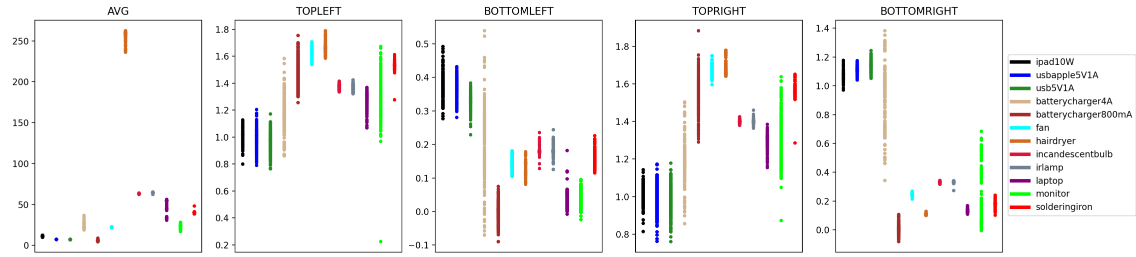

Figure 3: Five of the feature values plotted for the first 250 measurements.

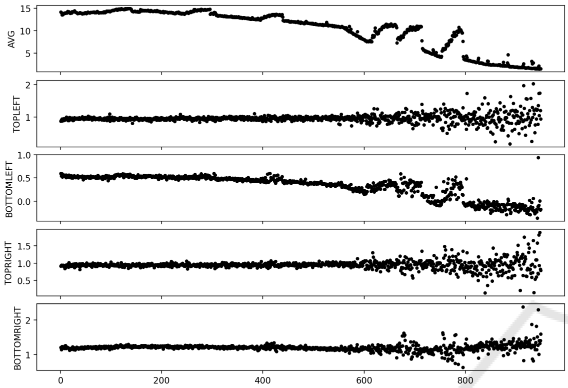

Figure 4: Five characteristics plotted for measurements taken during the charging of the iPad.

Dimensions: <20x14> Accuracy: 0.9648768472906404 WORST Accuracy: 0.9480863963622584	ipad10W	usbapple5V1A	usb5V1A	batterycharger4A	batterycharger 800mA	fan	hairdryer	incandescentbulb	irlamp	laptop	monitor	solderingiron
ipad10W	219.9	0.08	0	0	0	0	0	0	0	0	0	0
usbapple5V1A	0	183.5	36.52	0	0	0	0	0	0	0	0	0
usb5V1A	0	46.06	173.9	0	0	0	0	0	0	0	0	0
batterycharger4A	0	0	0	215.7	0	0.01	0	0	0	1.17	3.14	0
batterycharger800mA	0	0	0	0	220	0	0	0	0	0	0	0
fan	0	0	0	0	0	219	0	0	0	0	0	0
hairdryer	0	0	0	0	0	0	220	0	0	0	0	0
incandescentbulb	0	0	0	0	0	0	0	220	0	0	0	0
irlamp	0	0	0	0	0	0	0	0.07	219.9	0	0	0
laptop	0	0	0	0	0	0.32	0	0.81	0	218.6	0	0.25
monitor	0	0	0	1.78	0	1.57	0	0	0	0	216.7	0
solderingiron	0	0	0	0	0	0	0	0.03	0.88	0	0	219.1

Figure 5: Confusion matrix (average of 100 runs) of the SVM classification results. 30 samples from each class were used for training.

5 MEASUREMENT PROFILES

The previous section showed that the SVM method is accurate for classifying the measurement data collected. The question is whether similar results can be achieved with fewer data and if so, it also reduces computational complexity. Less computational complexity allows Edge Computing methods to be used and may also make it possible to run the classification on the ESP32 microcontroller in the future.

The definition of measurement profiles is introduced to modify the measurement parameters and enable the search for possible optimal choices. The measurement profile defines the parameters of the

Dimensions: <20x14> Accuracy: 0.9956025466120964 WORST Accuracy: 0.9881764438381082	ipad10W	batterycharger4A	batterycharger 800mA	fan	hairdryer	incandescentbulb	irlamp	laptop	monitor	solderingiron
ipad10W	220	0	0	0	0	0	0	0	0	0
batterycharger4A	0	215.7	0	0	0	0	0	1.22	3.05	0
batterycharger800mA	0	0	220	0	0	0	0	0	0	0
fan	0	0	0	219	0	0	0	0	0	0
hairdryer	0	0	0	0	220	0	0	0	0	0
incandescentbulb	0	0	0	0	0	220	0	0	0	0
irlamp	0	0	0	0	0	0.06	219.9	0	0	0
laptop	0	0	0	0.25	0	0.6	0	218.9	0	0.3
monitor	0	1.55	0	1.76	0	0	0	0	216.7	0
solderingiron	0	0	0	0	0	0.01	0.86	0.01	0	219.1

Figure 6: Confusion matrix (average of 100 runs) of the SVM classification results. 30 samples from each class were used for training. Only one USB class was used.

measurement. The measurement profile consists of the following:

- r - the number of different cutoff ratios
- $percentage\_min$ - the minimal cutoff ratio
- $percentage\_max$ - the maximum cutoff ratio
- h - the number of cycles the AC signal is cut for each cutoff ratio
- d - the number of cycles where the AC signal is not modified between measuring with two cutoff ratios

The cutoff ratios are evenly spaced between $percentage\_min$ and $percentage\_max$. The measurement profiles will be shown in the following form:

$$\{< r, percentage\_min - percentage\_max >, h, d\}$$

The number of cycles (one full period of the AC voltage signal - 20ms) required for one full measurement with a measurement profile can be calculated using the following formula:

$$N_{cycles} = h \cdot r + d \cdot (r - 1) \tag{3}$$

Multiple submatrices can be extracted from original measurements and used for classification. These submatrices extract the data that the measurement profile would have collected. (E.g.: if $h = 6$, then only the first six columns of the original matrices would be considered.) Running the simulations for multiple parameters shows an estimate of how accuracy would change using different measurement profiles. Using

RESULTS for WORST classification accuracy:										
	2	4	6	8	10	12	14	16	18	20
<14,10%-75%>	96.25%	96.56%	97.43%	98.00%	96.95%	97.90%	97.71%	97.62%	98.29%	97.71%
<7,10%-70%>	95.14%	96.08%	97.05%	97.05%	96.57%	96.76%	97.52%	96.00%	97.24%	97.14%
<5,10%-70%>	93.48%	96.39%	97.28%	97.20%	97.61%	96.86%	97.43%	97.14%	96.85%	96.85%
<4,10%-70%>	93.76%	95.34%	96.53%	96.73%	96.65%	97.50%	96.64%	96.56%	96.95%	97.52%
<3,10%-70%>	93.65%	95.19%	96.82%	96.86%	97.19%	97.29%	97.41%	97.42%	97.52%	97.52%
<2,10%-75%>	92.31%	94.84%	96.59%	97.02%	97.44%	97.96%	97.97%	98.07%	97.87%	97.67%
<3,15%-75%>	91.93%	95.50%	96.17%	97.11%	96.93%	97.25%	97.75%	97.08%	98.05%	97.67%
<4,15%-75%>	93.44%	96.04%	96.47%	97.18%	97.58%	97.49%	97.01%	96.63%	97.32%	97.79%
<5,15%-75%>	95.19%	96.56%	97.38%	97.30%	97.31%	97.51%	98.09%	97.51%	97.80%	98.09%
<7,15%-75%>	94.08%	96.60%	97.03%	97.14%	97.61%	97.04%	97.24%	96.37%	97.90%	98.09%

RESULTS for AVERAGE classification accuracy:										
	2	4	6	8	10	12	14	16	18	20
<14,10%-75%>	97.89%	98.23%	98.84%	99.04%	98.99%	98.99%	98.97%	98.92%	98.99%	99.00%
<7,10%-70%>	96.65%	97.73%	98.34%	98.62%	98.58%	98.52%	98.62%	98.51%	98.61%	98.63%
<5,10%-70%>	96.01%	97.79%	98.39%	98.62%	98.57%	98.43%	98.65%	98.45%	98.46%	98.43%
<4,10%-70%>	95.43%	97.30%	98.16%	98.52%	98.45%	98.76%	98.62%	98.52%	98.49%	98.79%
<3,10%-70%>	95.85%	97.22%	98.21%	98.40%	98.32%	98.70%	98.73%	98.76%	98.78%	98.84%
<2,10%-75%>	94.42%	96.54%	98.32%	98.48%	98.83%	99.22%	99.07%	99.19%	99.11%	99.01%
<3,15%-75%>	93.65%	96.75%	97.84%	98.37%	98.49%	98.60%	98.82%	98.84%	99.02%	99.03%
<4,15%-75%>	95.23%	97.40%	97.84%	98.59%	98.60%	98.43%	98.52%	98.56%	98.65%	98.82%
<5,15%-75%>	96.58%	97.68%	98.42%	98.91%	98.94%	98.86%	99.00%	98.97%	98.95%	99.03%
<7,15%-75%>	96.40%	97.85%	98.24%	98.74%	98.80%	98.71%	98.72%	98.70%	98.65%	98.88%

Figure 7: Simulations ran on an early version of the created dataset, 100 samples from each class, 30 used for training. The vertical axis shows the cutoff ratios, and the horizontal shows the number of AC cycles for each cutoff ratio. Each simulation was run 100 times, and the worst and average accuracy values were shown.

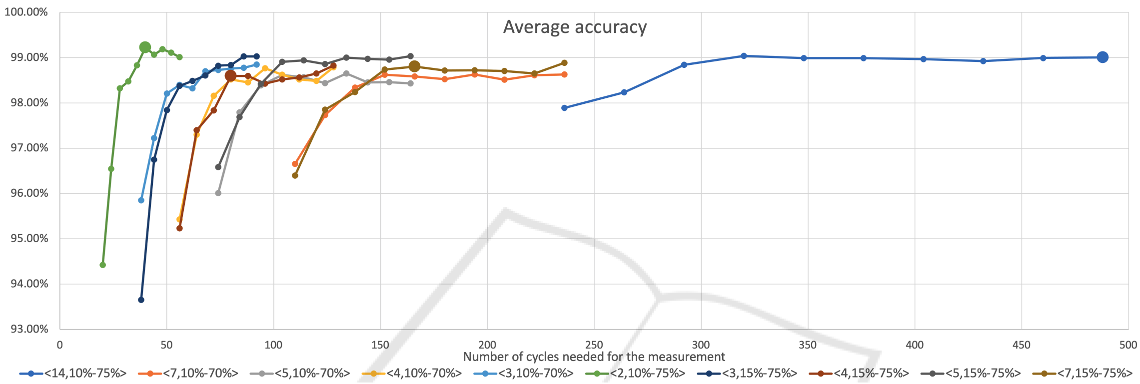

Figure 8: Average simulation results plotted for each cutoff ratio set. The bigger markers show the measurement profiles chosen. The horizontal axis shows the number of cycles each measurement would take assuming $d = 16$.

different cutoff ratio numbers between 2 and 14 and different h values between 2 and 20, the accuracy results can be seen in Figure 7.

Then we can choose measurement profiles to use for actual measurement collection. In the plots of the results for each measurement ratio set (Figure 8), it can be seen that by increasing h, the change in accuracy slows down, and only the measurement time increases. Based on this data, the following measurement profiles were selected:

- TEST_ORIG : $\{< 14, 10\% - 75\% >, h = 20, d = 16\}$ Measurement time: 488 AC cycles (9.76s)
- TEST_HALVED : $\{< 7, 15\% - 75\% >, h = 10, d = 8\}$ Measurement time: 118 AC cycles (2.36s)
- TEST_TINY : $\{< 2, 10\% - 75\% >, h = 12, d = 4\}$ Measurement time: 28 AC cycles (0.56s)
- TEST_FOUR : $\{< 4, 15\% - 75\% >, h = 8, d = 4\}$ Measurement time: 44 AC cycles (0.88s)

In Figure8, the selected measurement profiles are shown with a bigger maker.

The software of the microcontroller was also modified to allow measurements with measurement profiles. Data was collected for the same devices listed in Section 3.3. For each measurement profile, at least 250 measurements were taken per class.

Table 1: SVM Classification results for each measurement profile. Each classification was run 100 times, the average accuracy values are shown.

Measurement profile	All USB classes	One USB class (iPad)
TEST_ORIG	96.49%	99.56%
TEST_HALVED	93.36%	98.74%
TEST_TINY	91.89%	97.40%
TEST_FOUR	94.42%	99.35%

The results of the SVM classification with measurement profiles can be seen in Figures 9 (with separate USB classes) and 10 (one usb class - iPad). It can be seen that with the reduced amount of data collected, the classification accuracy decreases, but the average accuracy values are still over 91%. Table 1 summarizes the classification accuracy results for all measurement profiles.

A small training sample size (30) is enough to achieve above 99% accuracy with the SVM classification method. This means that only a few minutes are required to collect the necessary measurements and profile a device.

Dimensions: <10x7>
Accuracy: 0.933579545454546
WORST Accuracy: 0.9098484848484848

TEST_HALVED

ipad10W	219.8	0.17	0.02	0	0	0	0	0	0	0	0	0
usbapple5V1A	0	174.5	45.52	0	0	0	0	0	0	0	0	0
usb5V1A	0	103.5	116.6	0	0	0	0	0	0	0	0	0
batterycharger4A	0	0	0	216.9	0	0	0	0	0	0.31	2.82	0
batterycharger800mA	0	0	0	0	220	0	0	0	0	0	0	0
fan	0	0	0	0	0	220	0	0	0	0	0	0
hairdryer	0	0	0	0	0	0	220	0	0	0	0	0
incandescentbulb	0	0	0	0	0	0	0	219.9	0.1	0	0	0
irlamp	0	0	0	0	0	0	0	0.87	219.1	0	0	0
laptop	0	0	0	0	0	1.56	0	4.64	0	210.6	0	3.17
monitor	0	0	0	1.93	0	4.53	0	0	0	0	213.5	0
solderingiron	0	0	0	0	0	0	0	0	0	6.26	0	213.7

Dimensions: <12x2>
Accuracy: 0.9189427813565745
WORST Accuracy: 0.8976885183781735

TEST_TINY

ipad10W	217.5	1.07	1.42	0	0	0	0	0	0	0	0	0
usbapple5V1A	0.01	142.7	77.3	0	0	0	0	0	0	0	0	0
usb5V1A	0.55	76.89	142.6	0	0	0	0	0	0	0	0	0
batterycharger4A	0.21	0	0	219.1	0	0.08	0	0	0	0.08	0.57	0
batterycharger800mA	0	0	0	0	220	0	0	0	0	0	0	0
fan	0	0	0	0	0	220	0	0	0	0.03	0	0
hairdryer	0	0	0	0	0	0	220	0	0	0	0	0
incandescentbulb	0	0	0	0	0	0	0	219.2	0.84	0	0	0
irlamp	0	0	0	0	0	0	0	1.26	218.7	0	0	0
laptop	0	0	0	0	0	2.55	0	0	0	203.2	0	14.23
monitor	0	0	0	1.28	0	0	0	0	0	0	217.7	0
solderingiron	0	0	0	0	0	0	0	0	0	35.54	0	184.5

Dimensions: <8x4>
Accuracy: 0.9441818181818187
WORST Accuracy: 0.9166666666666666

TEST_FOUR

ipad10W	218.3	1.54	0.17	0	0	0	0	0	0	0	0	0
usbapple5V1A	0	159.1	60.91	0	0	0	0	0	0	0	0	0
usb5V1A	0	70.32	149.7	0	0	0	0	0	0	0	0	0
batterycharger4A	0.05	0	0	218.2	0	0	0	0	0	0.72	1.04	0
batterycharger800mA	0	0	0	0	220	0	0	0	0	0	0	0
fan	0	0	0	0	0	220	0	0	0	0	0.01	0
hairdryer	0	0	0	0	0	0	220	0	0	0	0	0
incandescentbulb	0	0	0	0	0	0	0	219.7	0.35	0	0	0
irlamp	0	0	0	0	0	0	0	3.95	216.1	0	0	0
laptop	0	0	0	0	0	0.08	0	1.98	0	217.7	0	0.2
monitor	0	0	0	3.17	0.86	1.86	0	0	0	0	214.1	0
solderingiron	0	0	0	0	0	0	0	0	0	0.15	0	219.9

Figure 9: Confusion matrix (average of 100 runs) of the SVM classification results for the new measurement profiles. 30 samples from each class were used for training. The column class labels are the same as in Figure 5.

Table 2: FC NN Classification results for each measurement profile. Each classification was run 100 times, one USB class was used.

Measurement profile	Average accuracy	Worst accuracy
TEST_ORIG	99.51%	98.53%
TEST_HALVED	98.52%	94.07%
TEST_TINY	97.88%	96.20%
TEST_FOUR	98.50%	97.20%

6 DATA CLASSIFICATION WITH NEURAL NETWORKS

The data were also classified with a simple, Fully Connected Neural Network. The input layer used the ten features chosen in Section 4, and two hidden layers of sizes 10 and 6 were used. The activation function was ReLU. The result can be seen in Figure 11, and the results are summarized in Table 2.

6.1 Classification with CNN

Convolutional Neural Networks are popular solutions in image processing tasks. As in the cases of the TEST_ORIG, TEST_HALVED, and TEST_FOUR

Dimensions: <10x7>
Accuracy: 0.9873545454545452
WORST Accuracy: 0.96

TEST_HALVED

ipad10W	220	0	0	0	0	0	0	0	0	0
batterycharger4A	0	216.5	0	0	0	0	0	0.43	3.06	0
batterycharger800mA	0	0	220	0	0	0	0	0	0	0
fan	0	0	0	220	0	0	0	0	0.01	0
hairdryer	0	0	0	0	220	0	0	0	0	0
incandescentbulb	0	0	0	0	0	219.9	0.09	0	0	0
irlamp	0	0	0	0	0	0.64	219.4	0	0	0
laptop	0	0	0	1.58	0	4.36	0	210.6	0	3.43
monitor	0	2.41	0	5.23	0	0	0	0	212.4	0
solderingiron	0	0	0	0	0	0	0	6.58	0	213.4

Dimensions: <12x2>
Accuracy: 0.9739745338790358
WORST Accuracy: 0.9404274670304684

TEST_TINY

ipad10W	220	0	0	0	0	0	0	0	0	0
batterycharger4A	0.12	219.2	0	0.06	0	0	0	0.1	0.5	0
batterycharger800mA	0	0	220	0	0	0	0	0	0	0
fan	0	0	0	220	0	0	0	0.02	0	0
hairdryer	0	0	0	0	220	0	0	0	0	0
incandescentbulb	0	0	0	0	0	218.9	1.12	0	0	0
irlamp	0	0	0	0	0	1.2	218.8	0	0	0
laptop	0	0	0	2.76	0	0	0	202.1	0	15.15
monitor	0	1.27	0	0	0	0	0	0	217.7	0
solderingiron	0	0	0	0	0	0	0	34.93	0	185.1

Dimensions: <8x4>
Accuracy: 0.9934727272727276
WORST Accuracy: 0.9877272727272727

TEST_FOUR

ipad10W	220	0	0	0	0	0	0	0	0	0
batterycharger4A	0.12	218.4	0	0	0	0	0	0.78	0.69	0
batterycharger800mA	0	0	220	0	0	0	0	0	0	0
fan	0	0	0	220	0	0	0	0	0	0
hairdryer	0	0	0	0	220	0	0	0	0	0
incandescentbulb	0	0	0	0	0	219.5	0.49	0	0	0
irlamp	0	0	0	0	0	4.14	215.9	0	0	0
laptop	0	0	0	0.08	0	2.16	0	217.4	0	0.35
monitor	0	2.75	0.88	1.81	0	0	0	0	214.6	0
solderingiron	0	0	0	0	0	0	0	0.11	0	219.9

Figure 10: Confusion matrix (average of 100 runs) of the SVM classification results for the new measurement profiles. 30 samples from each class were used for training. Only one USB class was used. The column class labels are the same as in Figure 6.

measurement profile matrices, we can interpret the task at hand as an image processing task with a low-resolution input image. Using only the power matrix, the network could not distinguish between the incandescent light bulb and the infrared lamp. As it turns out, the infrared lamp used for the measurements is also an incandescent bulb emitting infrared radiation, so we expect them to have similar characteristics. This inability to distinguish between the same kind of devices shows the CNN's capability to extract generalized features and shows the network's deeper understanding of the connected load.

The CNN consisted of two convolutional layers with (3×3) kernels. The first used ReLU and padding, while the second did not use padding and used softmax as the activation function. We were using softmax provided normalization before the FC layers. The first convolutional layer extracted 48 features, while the second extracted 64 features. After flattening the layers, two hidden, fully connected layers with ReLU activation function were used (48 and 64 neurons) before the final layer with softmax activation. The results of the classification can be seen in Figure 12. By using the Power, the Voltage (divided by 230), and the Current matrices, 50 samples for each class in the training set are enough to achieve

Table 3: CNN Classification results for the TEST_ORIG, TEST_HALVED and TEST_FOUR profiles.

Used data — Training samples per class	Power — 150		Power, Voltage, Current — 50	
Accuracy	avg	worst	avg	worst
TEST_ORIG	99.92%	99.56%	99.84%	99.50%
TEST_HALVED	99.90%	99.56%	99.86%	99.44%
TEST_FOUR	99.35%	75.44%	98.81%	82.17%

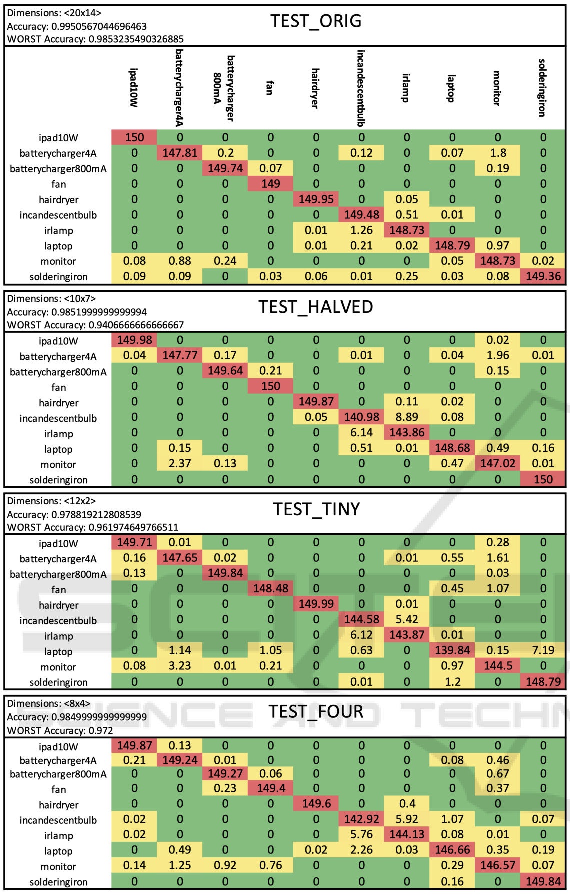

Figure 11: Confusion matrix (average of 100 runs) of the FC NN classification results. 100 samples from each class were used for training, 150 for testing. One USB class was used.

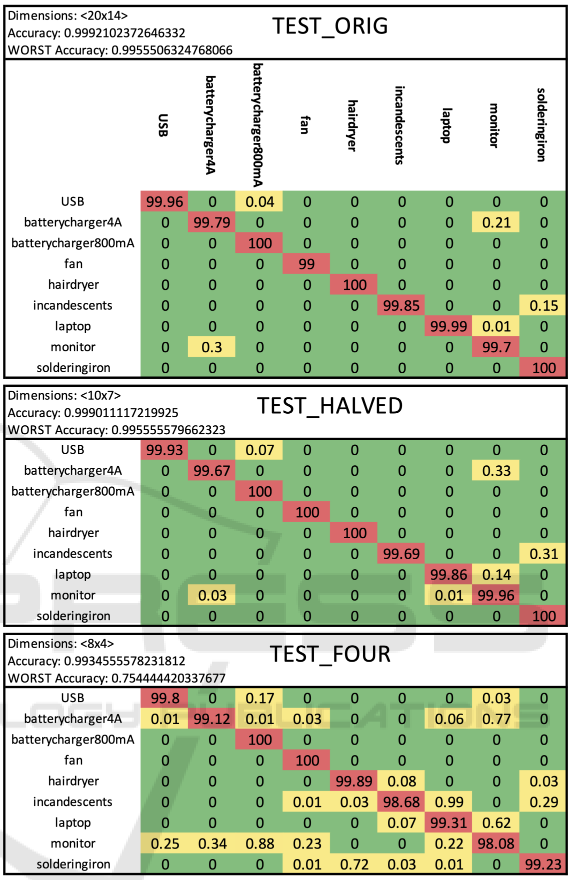

Figure 12: Confusion matrix (average of 100 runs) of the CNN classification results. A common USB class was used for the three USB adapters, and the incandescent light bulb and the infrared lamp were merged to one class (incandescents). 150 samples from each class were used for training the model, 100 were used for testing.

the same accurate results. The results are shown in Figure 13. Table 3 summarizes the results.

7 CONCLUSION

We have presented a solution to the fast classification of electric loads. The measurement time can be as low as 0.56s, while a complete measurement takes less than 10 seconds. We proposed different classification methods suited for different applications. While deep networks such as CNN can provide high (99.92% average) accuracy rate and better generalization, the computational requirements are much higher. For edge computing solutions, traditional FC NN and SVM provide a better solution to achieve similar results with less computational resources. If the data collection is the bottleneck, then SVM is the best option as a small dataset is enough thanks to the carefully selected features used for the input of the SVM classification. As SVM requires the least amount of computational power from the methods presented, it is ideal for on-device classification. Classification on the microcontroller of the measurement prototype device is one area considered for future research related

Dimensions: <20x14>
Accuracy: 0.9984380155801773
WORST Accuracy: 0.9949972033500671

TEST_ORIG

	USB	batterycharger4A	batterycharger800mA	fan	hairdryer	incandescents	laptop	monitor	solderingiron
USB	199.1	0	0.9	0	0	0	0	0	0
batterycharger4A	0	199.22	0	0	0	0	0	0.78	0
batterycharger800mA	0	0	200	0	0	0	0	0	0
fan	0	0	0	199	0	0	0	0	0
hairdryer	0	0	0	0	200	0	0	0	0
incandescents	0	0	0	0	0	199.84	0	0	0.16
laptop	0	0	0	0	0	0	199.98	0.01	0.01
monitor	0	0.61	0	0.34	0	0	0	199.05	0
solderingiron	0	0	0	0	0	0	0	0	200

Dimensions: <10x7>
Accuracy: 0.9986055541038513
WORST Accuracy: 0.9944444298744202

TEST_HALVED

USB	199.16	0	0.84	0	0	0	0	0	0
batterycharger4A	0.03	199.19	0	0	0	0	0	0.78	0
batterycharger800mA	0	0	200	0	0	0	0	0	0
fan	0	0	0	200	0	0	0	0	0
hairdryer	0	0	0	0	200	0	0	0	0
incandescents	0	0	0	0	0	199.5	0	0	0.5
laptop	0	0	0	0	0	0	199.87	0.13	0
monitor	0	0.1	0.02	0.09	0	0	0	199.79	0
solderingiron	0	0	0	0.02	0	0	0	0	199.98

Dimensions: <8x4>
Accuracy: 0.9881722241640091
WORST Accuracy: 0.8216666579246521

TEST_FOUR

USB	198.32	0.02	1.66	0	0	0	0	0	0
batterycharger4A	0.23	197.55	0.02	0.38	0	0.01	0.11	1.7	0
batterycharger800mA	0	0	200	0	0	0	0	0	0
fan	0	0	0	198.82	0	0	0	0	1.18
hairdryer	0	0	0	0	195.97	1.11	0.02	0	2.9
incandescents	0	0	0	0.89	0.91	197.09	0	0	1.11
laptop	0	0	0	0	0.07	0.4	198.13	1.4	0
monitor	0.64	0.67	2.01	0.98	0	0.01	1.45	194.24	0
solderingiron	0	0	0	1.26	0.07	0.08	0	0	198.59

Figure 13: Confusion matrix (average of 100 runs) of the CNN classification results using the Power, Current, and Voltage(divided by 230) matrices. A common USB class was used for the three USB adapters, and the incandescent light bulb and the infrared lamp were merged into one class (incandescents). Fifty samples from each class were used for training the model, 200 were used for testing.

to this topic.

We have also introduced measurement profiles that show that even less data is enough to classify the connected load accurately. A reduction of the amount of collected data also reduces the computational requirements of the classification. Based on the requirements of the classification system, the data collection can be optimized with the help of measurement profiles to achieve faster device labeling and data processing while decreasing accuracy only by a small amount.

7.1 Future Work

With the method presented, we have shown that with only 30 training samples, SVM classification could achieve an average of 99.56% accuracy rate. This means that even with the longest test profile, the training data collection requires less than 6 minutes of measurement per electric load. The CNN approach shows that the network can understand the type of features and can generalize, so similar types of devices (like USB chargers) will be accurately classified; however, the current system cannot detect new types of electric loads that were not measured previously. Detecting a previously unknown device as unknown is a complex task. In future work, we intend to examine Open Set classification methods for detecting previously unseen devices. A smart plug system with Open Set classification methods could automatically trigger the training data collection for a previously unseen load. User interaction would only be needed for providing a label for the device.

The other area considered for future work is the classification on the microcontroller. The methods presented may enable the classification of the connected load on the ESP32 microcontroller inside the prototype device. With the WiFi capabilities of the microcontroller, a Wireless Sensor Network could be built. As the dimmer used in the prototype device can cut the connected load's power supply, the prototype is capable of not only measuring but also controlling the load, so no hardware modifications would be required for a smart plug system.

ACKNOWLEDGEMENTS

Supported by the ÚNKP-20-1 and ÚNKP-21-1 new national excellence program of the ministry for innovation and technology from the source of the national research, development and innovation fund.

SCIENCE AND TECHNOLOGY PUBLICATIONS

REFERENCES

da S. Veloso, A. F., de Oliveira, R. G., Rodrigues, A. A., Rabelo, R. A. L., and Rodrigues, J. J. P. C. (2019). Cognitive smart plugs for signature identification of residential home appliance load using machine learning: From theory to practice. In *2019 IEEE International Conference on Communications Workshops (ICC Workshops)*, pages 1–6.

Du, L., He, D., Harley, R. G., and Habetler, T. G. (2016). Electric load classification by binary voltage–current trajectory mapping. *IEEE Transactions on Smart Grid*, 7(1):358–365.

Gomes, L., Sousa, F., Pinto, T., and Vale, Z. (2019). A residential house comparative case study using market available smart plugs and enaplugs with shared knowledge. *Energies*, 12:1647.

Gomes, L., Sousa, F., and Vale, Z. (2018). An intelligent smart plug with shared knowledge capabilities. *Sensors*, 18(11):3961.

Jaradat, M., Jarrah, M., Jararweh, Y., Al-Ayyoub, M., and Bousselham, A. (2014). Integration of renewable energy in demand-side management for home appliances. In *2014 International Renewable and Sustainable Energy Conference (IRSEC)*, pages 571–576.

Petrović, T. and Morikawa, H. (2017). Active sensing approach to electrical load classification by smart plug. In *2017 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pages 1–5.

Ridi, A., Gisler, C., and Hennebert, J. (2014). A survey on intrusive load monitoring for appliance recognition. In *2014 22nd International Conference on Pattern Recognition*, pages 3702–3707.

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS