Electrical Power and Energy Systems 104 (2019) 645–653

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

Detection of unidentified appliances in non-intrusive load monitoring using siamese neural networks

Leen De Baets*, Chris Develder, Tom Dhaene, Dirk Deschrijver

IDLab, Department of Information Technology, Ghent University – imec, Technologiepark-Zwijnaarde 15, 9052 Ghent, Belgium

ARTICLE INFO

Keywords:
Non-intrusive load monitoring
Appliance classification
Voltage-current trajectory
Siamese neural network

ABSTRACT

Non-intrusive load monitoring methods aim to disaggregate the total power consumption of a household into individual appliances by analyzing changes in the voltage and current measured at the grid connection point of the household. The goal is to identify the active appliances, based on their unique fingerprint. Most state-of-the-art classification algorithms rely on the assumption that all events in the data stream are triggered by known appliances, which is often not the case. This paper proposes a method capable of detecting previously unidentified appliances in an automated way. For this, appliances represented by their VI trajectory are mapped to a newly learned feature space created by a siamese neural network such that samples of the same appliance form tight clusters. Then, clustering is performed by DBSCAN allowing the method to assign appliance samples to clusters or label them as 'unidentified'. Benchmarking on PLAID and WHITED shows that an $F_{1,macro}$-measure of respectively 0.90 and 0.85 can be obtained for classifying the unidentified appliances as 'unidentified'.

1. Introduction

In October 2014, EU leaders agreed upon three key targets for the year 2030 [1]: (1) a reduction of at least 40% cuts in greenhouse gas emissions, (2) a save of at least 27% share for renewable energy, and (3) at least 27% improvement in energy efficiency. Energy monitoring proves a useful aid to reach these targets by providing an accurate, detailed view of energy consumption. It helps because: (1) if this information is given to households, studies have shown that they could save up to 12% of electrical energy and thereby reduce the emissions [2] (also useful for non-residential buildings [3]), (2) this information allows us to assess and exploit the flexibility of power consumption, which in turn is important for demand response systems that are responsible for an increased penetration of distributed renewable energy sources, (3) energy monitoring is one major prerequisite for energy efficiency measures [4].

In order to achieve the required energy monitoring cost-effectively, i.e., without relying on per-device monitoring equipment, non-intrusive load monitoring (NILM) provides an elegant solution [5]. NILM identifies the per-appliance energy consumption by first measuring the aggregated energy trace at a single, centralized point in the home using a sensor and then disaggregating this power consumption for individual devices, using machine learning techniques.

Several supervised and unsupervised methods have been developed to recognise the appliances and to compute the total power consumption [6,7,5]. However, to our knowledge, most classification algorithms described in the literature can not handle unidentified appliances. These will be assigned a label and power consumption that corresponds to the appliance having the most similar features. This paper suggest a method that is capable of detecting unidentified appliances, which are labeled as 'unidentified'. When such an appliance is detected, the user can be queried for information about the appliance (i.e., the class label). In this paper, appliances are characterised by their binary VI trajectory image [8,9], although other representations can also be considered.

The proposed method has a training and a test phase, as shown in Fig. 1. In the training phase, a new, lower dimensional feature space where samples of the same appliance are clustered, is computed from the VI trajectory images by training a siamese neural network. The VI trajectory images must be paired and labelled respectively as must- or cannot-links, depending if the images belong to the same class or not. The transformation does not depend on the appliance label. On the transformed input, DBSCAN is performed to group samples with similar feature vectors in the new space. DBSCAN is a state-of-the-art clustering method that does not require prior knowledge about the amount of clusters and that is capable of detecting outliers. In the test phase, a VI trajectory image is transformed to the new feature space. If this point does not belong to a cluster, it is labelled as 'unidentified'.

The outline of the paper is as follows: Section 2 describes the related

* Corresponding author.
E-mail address: leen.debaets@ugent.be (L. De Baets).

https://doi.org/10.1016/j.ijepes.2018.07.026
Received 4 December 2017; Received in revised form 26 May 2018; Accepted 16 July 2018

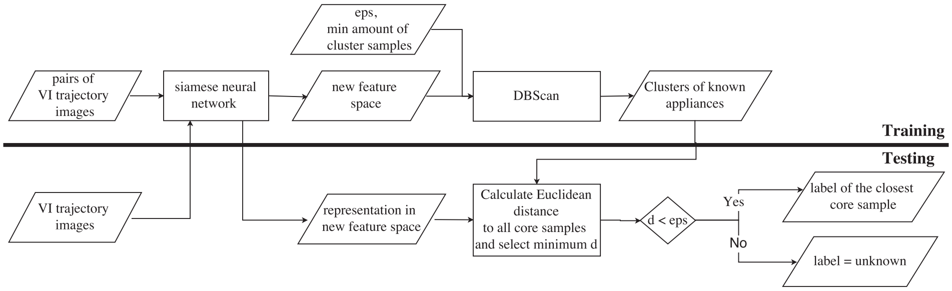


Fig. 1. The work flow of the proposed method that is able to detect unidentified appliances.

work concerning NILM classification algorithms. Section 3 explains the concept of siamese neural networks and how they can be used to learn a new feature space. Section 4 explains the DBSCAN clustering algorithm. Section 5 benchmarks the quality of the clustering, the capability of detecting unidentified appliances and the generalization property of the method. Additionally, it also discusses how this method can be used in a quasi real-time solution. Finally, Section 6 concludes this paper.

2. Related work

A specific application of NILM is appliance detection. Hart [10] was the first to describe the several steps in this process: (1) measuring the aggregated power consumption with a sensor attached to the main power cable, (2) detecting state-transitions of appliances (events) from the captured data using a robust statistical test [11], (3) describing transitions using a well-chosen feature vector, e.g., VI trajectories, (4) recognizing and monitoring each appliance using supervised and unsupervised methods. It must be noted that for some NILM algorithms, the event detection is a side effect of the approach, and not a separate module in the algorithm itself.

Feature definition. After detecting state-transitions of appliances, these must be described by a well-chosen feature vector. The type of features depends strongly on the sampling rate of the measurements. When using low frequency data (⩽1 Hz), the most common features are the power levels and the ON/OFF durations [12]. A drawback of this approach is that only energy-intensive appliances can be detected. When using higher frequency data, it is possible to calculate features like the harmonics [13] and the frequency components [14] from the steady-state and transient behaviour of the current and voltage signal, enabling the algorithm to also detect non energy-intensive appliances. More recently, the possibility to consider voltage-current (VI) trajectories has also been considered [8,9,15].

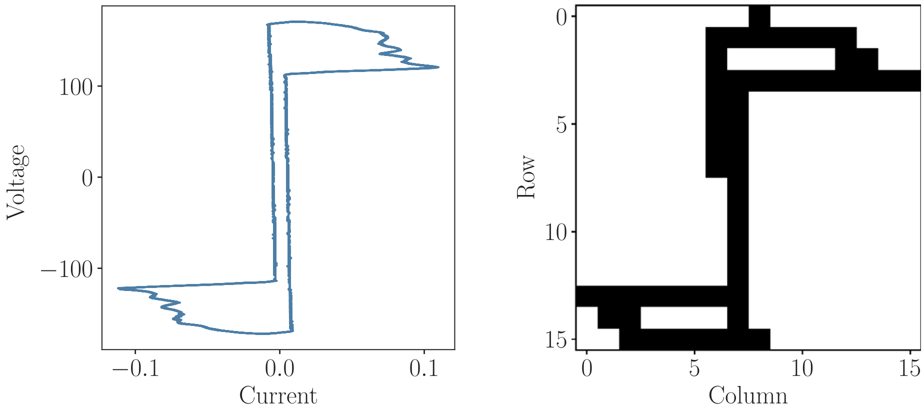
The VI trajectory of an appliance is obtained by plotting the voltage against the current for a defined time period when the appliance is turned on, see Fig. 2a. It is shown in [15] that manually extracting features from the VI trajectory can be informative to classify the appliances. Nevertheless, this is not straightforward. As an alternative, the VI trajectory can be converted into a binary VI image ($n \times n$ matrix) by meshing the VI trajectory, see Fig. 2b. In [8,9], each cell of the mesh is assigned a binary value that denotes whether or not it is traversed by the trajectory. Based on this binary VI image, several features can be extracted to classify different power loads [9]. Even the binary VI image itself can be used as input for a classifier [8], as will also be the case in this paper.

In order to distinguish appliances based on their VI trajectories, measurement devices must be used that are able to sample high frequency data.

Recognizing appliances and monitoring power consumption. Once the features are extracted, they can be fed into different classification methods, like support vector machines (SVM) [16,17], neural networks [18], decision trees [19], or nearest neighbours [20]. For these methods, labelled training data is necessary. If labels are not present, unsupervised methods can be used. An overview of these methods is given in [21].

The majority of the NILM approaches, supervised or unsupervised, are sensitive to appliance changes in the house, thus require regular retraining. In this paper, the focus lies on creating a classification algorithm that is able to detect unidentified appliances and is thus resilient against appliance changes in the house. If an unidentified appliance is detected, labeling and retraining is requested.

Clustering. In order to detect unidentified appliances, clustering must be performed. The idea is that samples originating from the same appliances will appear as clusters in the feature space and samples originating from unidentified appliances will appear as outliers indicating the need to create a new cluster. The use of clustering methods has previously been explored in non-intrusive load monitoring. In [10], a simple clustering algorithm was mentioned where the appliances are grouped using the active – reactive power (*P*-*Q*) plane as feature



(a) (b)

Fig. 2. The original VI trajectory (left) and the corresponding binary VI image (right).

representation space. Despite its simplicity, it is incapable of recognizing appliances with overlapping *P* and *Q* consumption. In [22], the *P*-*Q* plane is also used for genetic k-means and agglomerative clustering. This method has problems in distinguishing appliances with small *P* and *Q* consumptions as their steady-state changes tend to cluster together. In [23], mean-shift clustering is proposed on features that are extracted from the power signal. The resulting clusters are classified into different appliance classes.

None of these clustering algorithms exploit their capability to detect unidentified appliances, and none are capable of clustering on appliance-level with high accuracy. The proposed method uses a novel clustering work flow to cope with these two shortcomings. Section 3 explains how a higher accuracy can be obtained by learning a new feature space using siamese neural networks. Section 4 explains how unidentified appliances can be detected.

3. Siamese neural network

The ability of clustering algorithms to detect small power consuming appliances can be improved by adding more features. However, clustering is sensitive to the curse of dimensionality as it relies on the computation of a distance function like the Euclidean distance. In a high-dimensional case, the differences in distance become less apparent, making the clustering method unusable. For clustering to work, it is thus key to find a low dimensional feature space where the clusters are well separated. To this end, siamese neural networks can be used. These are a special kind of neural networks [24].

A siamese network consists of two identical NN, meaning that each of them has the same architecture, parameter values and weights. As input, two binary VI images must be given and as label, a binary value indicating whether or not the images belong to the same class. The output of the siamese network are two vectors, forming a lower-dimensional representation of the two input images. The idea is to learn the representation in such a way, that the distance between these representations will be smaller than a given threshold if the two belong to the same class and larger if not. This leads to the use of the so-called contrastive loss function:

$$L(y, d) = \frac{1}{2}(y \times d + (1-y) \times \max\{m-d, 0\}) \tag{1}$$

where *y* is the binary output, *d* is the distance between the two input feature vectors, and *m* is the margin determining when samples are dissimilar: dissimilar input vectors only contribute to the loss function if their distance is smaller than the margin.

Siamese networks are ideally suited to find a relationship between two comparable samples. This is the case in one-shot learning [25], where classification needs to be done with only one example of each class or signature verification [26], where the authenticity of a signature is checked. In this paper, the siamese neural network is used for dimensionality reduction, like in [27]. This method of dimension reduction is different from classical approaches, such as local linear embedding (LLE) and principal component analysis (PCA), as the siamese neural network learns a function that is capable of consistently mapping unseen samples to the learned feature space and as the siamese neural network is not constrained by a simple distance function like the Euclidean distance.

After the training phase, the siamese neural network can be used to calculate a n_{out}-dimensional representation of new VI binary images. These are obtained by using the output of just one cNN.

4. DBSCAN

After learning the feature space, unidentified appliances can be detected by performing clustering. Namely, if a new sample is too distant from present clusters (representing known appliances), then it is considered as unidentified.

Density-based spatial clustering of applications with noise (DBSCAN) is a density-based clustering algorithm: points forming a cluster will be close together, whereas the outliers will only have relatively far away neighbours [28]. The algorithm starts by picking one random sample out of the dataset. If there are not enough close by neighbours, then the point will be labeled as an outlier and the process continues by selecting a new sample. If there are sufficient close by neighbours, they are all added to the same cluster. The algorithm continues by iterating over all new added points, if these have sufficient close by neighbours, these are also added to the same cluster. This continues until no more samples are added to this cluster. Then a new unvisited random sample is selected and the process is iterated until all points belong to a cluster or are labeled as outliers (noise). Three elements needs to be defined for DBSCAN: (1) the amount of sufficient close by points, mintPts, (2) the distance function, and (3) the maximal distance to a close by sample, ϵ. These last two points define if a sample is close by or not.

The advantages of DBSCAN are that the number of clusters does not need to be specified by the user (unlike, e.g., for K-means clustering), clusters can be of any shape (not just the circular ones), and outliers are not forced to belong to a cluster but are identified as such. The algorithm is also robust against an imbalance in the occurrence of samples from different clusters. DBSCAN is one of the most common clustering algorithms and was awarded the test of time award at the leading data mining conference, KDD [29].

In this paper, the transformed input samples are clustered with DBSCAN. The used distance metric is the Euclidean distance. The parameters mintPts and ϵ are not trained but heuristically set to respectively 5 and 0.2.

To determine which cluster a test sample belongs to (if any), its feature vector is first transformed to the calculated lower-dimensional space. Next, the Euclidean distance is calculated to all core samples and the minimal distance is selected. If this distance is smaller than threshold ϵ, then the sample belongs to the same cluster as the closest core sample. Otherwise, it will be assigned the label ‘unidentified’.

5. Results

This section first describes the used datasets in Subsection 5.1 and specifies the input and architecture of the siamese neural network in Subsection 5.2. To benchmark the described method on the data, several checks must be performed. First, it must be examined if the learned feature space separates the different classes well, see Subsection 5.3. Second, the capability of detecting unidentified appliances is tested by using data of an unidentified appliance as test data, see Subsection 5.4. Lastly, the generalization property of the method is checked by using the data from other (unseen) houses as test data, see Subsection 5.5. To conclude this section, a discussion concerning the time usage of this algorithm is given in Subsection 5.6.

5.1. Benchmark dataset

The performance of the proposed algorithm is validated on the Plug-Load Appliance Identification Dataset (PLAID) [30] and the Worldwide Household and Industry Transient Energy Dataset (WHITED) [31].

- PLAID is a public dataset including sub-metered current and voltage measurements sampled at 30 kHz for 11 different appliance types. For each appliance type, at least 38 individual appliances are available, captured in 55 households. For each appliance, at least 5 start-up events are measured, resulting in a total of 1074 measurements.
- WHITED is a public dataset including sub-metered current and voltage measurements sampled at 44 kHz for 46 different appliance types. For each appliance type, 1 to 9 different appliances are available. For each appliance, 10 start-up events are measured,

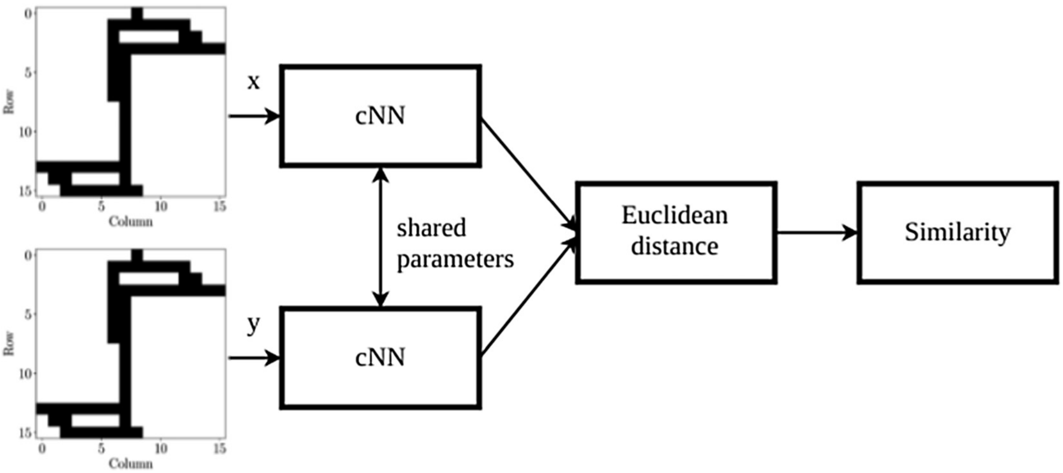


Fig. 3. The architecture of the siamese network.

resulting in a total of 1100 measurements. For training and testing purposes, it is required that at least two different appliances are measured per type. For 24 appliance types, only one appliance was available and these are thus omitted. The final number of used appliances in WHITED is 22, resulting in 860 measurements.

5.2. Input and architecture of siamese neural network

In this paper, the input of the siamese networks consists of pairs of binary VI trajectory images with size $n \times n$. In order to obtain the binary VI images for PLAID and WHITED, the voltage and current are measured over a time interval of 20 cycles when the appliances reach steady-state, resulting in respectively 10000 and 17600 samples. The voltage is plotted against the current and the methodology described in the related work is used to create binary VI images. Fig. 5 gives examples of binary VI images for the appliance types present in PLAID.

The architecture of the siamese network is shown in Fig. 3. For the siamese neural network, the proposed method uses two convolutional neural network (cNN). cNNs are a type of neural networks (NNs) that are often used in computer vision because they are highly suitable to classify images [32]. The used cNN is depicted in Fig. 4 and takes as input an $n \times n$ binary image. This is transformed by a convolutional layer which uses 20 filters, each considering regions of 5×5 pixels. After the convolutional layer, there is a maximal pooling layer with a sliding window of 2×2. This combination of a convolutional layer followed by a pooling layer is repeated, and finally, a dense layer is added with n_{out} nodes. In total, five layers are present. The trainable weights and biases of the network are initialized by sampling from a Gaussian distribution with zero mean and a standard deviation of 0.05. The margin m used in the loss function is set to 50. The results are fairly insensitive to the choice of m.

5.3. Scenario 1

To determine if the learned feature space from the siamese neural network separates the classes well, the rand index (RI) of the clusters found by the DBSCAN algorithm is calculated. The RI is a measure of similarity between two data clusterings X and Y:

$$RI = \frac{a+b}{a+b+c+d} \tag{2}$$

where a and b are respectively the number of pairs of elements that are in the same/different cluster(s) in both clusterings X and Y, and c and d are respectively the number of pairs that are in the same cluster for X/Y, but in a different one for Y/X. Higher values of RI (max. value 1) indicate a better match of clusterings.

Fig. 6a and b show the RI for respectively PLAID and WHITED for different parameter configurations of the siamese neural network that learns the representation from binary images of the VI trajectories. The input size of the VI image ($n \times n$) and the dimension n_{out} of the learned representation are altered. For this, all data samples of the PLAID dataset are fed into the siamese neural network to calculate the mapping function. Increasing n_{out} (for fixed values of n) has little impact on the RI values. In contrast, changes in the value of n (for fixed values of n_{out}) have a strong impact. The best RI values for both datasets are obtained for $n \geqslant 30$ with a maximum of 0.996 for PLAID and 0.879 for WHITED. The RI value for WHITED is lower than the one for PLAID. A possible explanation is that the number of appliances is much larger for WHITED, introducing more chance for confusion. The high RI values confirm the capability of the siamese neural network to learn a feature space where the clusters are well separated and confirm the ability of the DBSCAN algorithm to find these clusters. Fig. 7 shows an example of the learned three dimensional feature space ($n_{out} = 3$ for the ease of visualization) for PLAID and the corresponding clustering. Each axis of

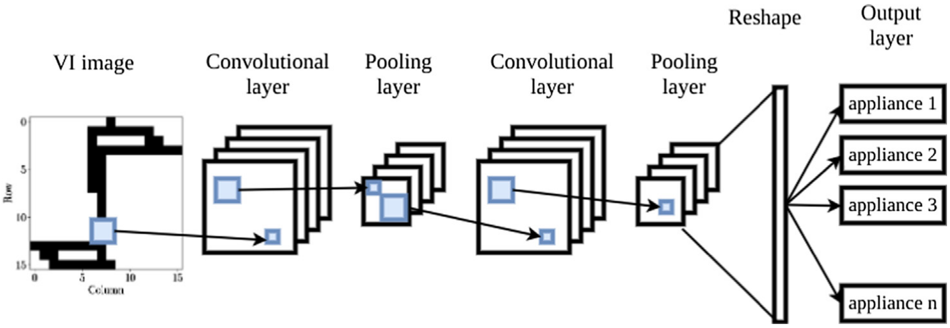


Fig. 4. The architecture of the cNN that is used in the siamese network.

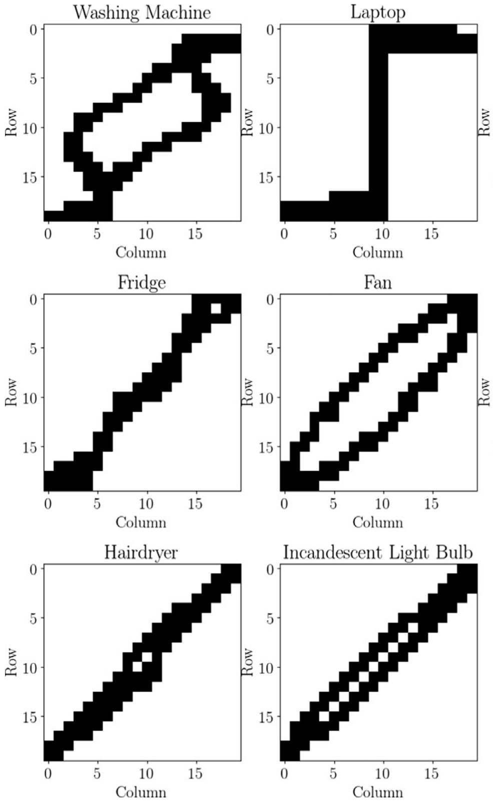


Fig. 5. Examples of binary VI images with $n = 20$ for the appliance types present in PLAID.

the plot represents one element of the neural network's output vector with length n_{out}. The label of each cluster is written next to it. A visualization for WHITED is not given as the number of appliances is too big to create a clear plot.

To indicate that the method also works for lower frequency data, the method is benchmarked on subsampled data from PLAID and WHITED. The original sampling frequency of PLAID can be reduced from 30 kHz to 15, 3, 1 and 0.5 kHz by selecting respectively 1/2, 1/10, 1/30, 1/60 of the samples. The original frequency of WHITED can be reduced from 44 kHz to 22, 4, 1, and 0.5 kHz by selecting respectively 1/2, 1/11, 1/44, 1/88 of the samples. Fig. 8 shows the VI trajectories for the fridge and hairdryer from PLAID for different sampling frequencies. These can still be separated, although this is more difficult for the human eye when the data is sampled at 0.5 and 1 kHz. The RI for when the feature space is learned using different sampling frequencies for PLAID and WHITED are given in Table 1 for when $n_{out} = 3$ and $n = 30$. The results indicate that the method remains robust if the sampling frequency is reduced. This means that for the CNN, the binary VI images of different appliance types are still sufficiently distinct. Note that, only for this scenario the sampling frequency is lowered. However, as the result for the different sampling frequencies is alike, similar conclusions can be drawn for the forthcoming scenarios.

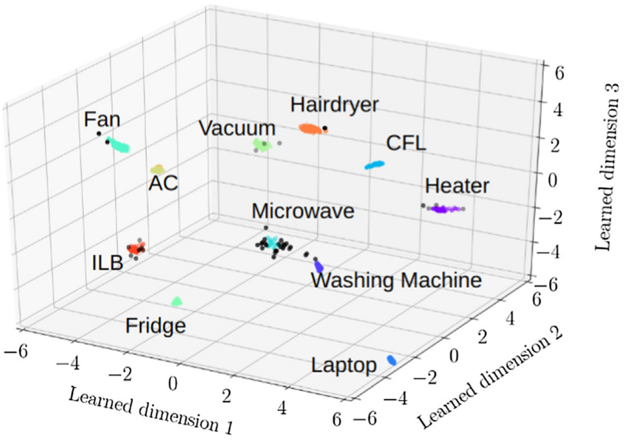
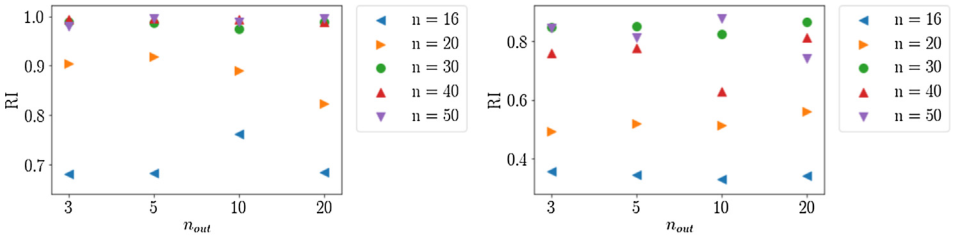


Fig. 7. The clusters found by DBSCAN in the learned feature space.

5.4. Scenario 2

To define how well the method can identify unidentified appliances, training is done on 10 appliances and testing on 1 hold out appliance. It



(a)

(b)

Fig. 6. The rand index for (a) PLAID and (b) WHITED when all data is used to learn the feature space with dimension n_{out}.

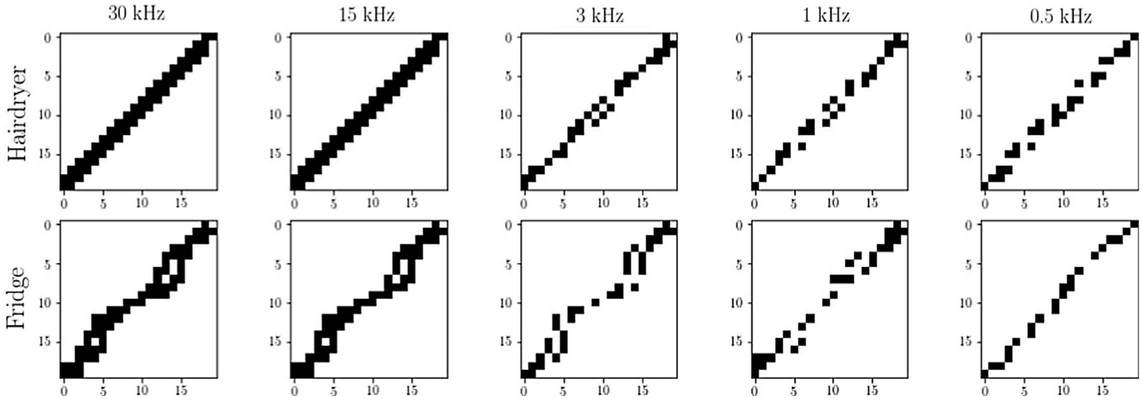


Fig. 8. The binary VI images for the hairdryer and fridge in PLAID when using different sampling frequencies.

Table 1
The rand index for PLAID and WHITED when all data is used to learn the feature space with dimension $n_{out} = 3$, the image size is $n_{out} = 30$, and sampling frequency is altered.

	PLAID		WHITED
30 kHz	0.99	44 kHz	0.85
15 kHz	1	22 kHz	0.85
3 kHz	0.99	4 kHz	0.81
1 kHz	1	1 kHz	0.87
0.5 kHz	0.99	0.5 kHz	0.85

is validated whether (1) the 10 selected appliances are properly separated in individual clusters, and (2) the 11th appliance has its data points classified as ‘unidentified’. The first criterion is validated by calculating the RI, like done above, and the second criterion by calculating, as recommended in [33], the F_1-measure of the hold out appliance whose samples must be labeled as ‘unidentified’. For this hold out appliance, the F_1-measure is defined as:

$$F_1 = 2\cdot\frac{precision\cdot recall}{precision + recall} \tag{3}$$

$$precision = \frac{TP}{TP + FP} \tag{4}$$

$$recall = \frac{TP}{TP + FN} \tag{5}$$

with *TP* the true positives (the appliance’s samples labelled as ‘unidentified’), *FP* the false negatives (the appliance’s samples assigned to a cluster), and *FN* the false negatives (other samples labelled as ‘unidentified’). This procedure is performed in a leave-one-appliance-out cross-validation. As PLAID and WHITED contain respectively 11 and 22 appliances, the training and testing is repeated 11 and 22 times, resulting in 11 and 22 F_1-measures. To obtain the final test result, the macro-average is taken:

$$F_{macro} = \sum_{i=1}^{a} F_{1,i} \tag{6}$$

where a is the amount of different appliances, and $F_{1,i}$ is the F_1-measure when appliance i is used as hold out appliance.

Fig. 9a and c display the RI values, and Fig. 9b and d the F_{macro} for respectively PLAID and WHITED for different n_{out} and n values. Increasing n_{out} and keeping n fixed has little impact on the values. Setting $n_{out} = 3$ is already sufficient to represent the VI image. However, changing n and keeping n_{out} fixed has a bigger influence. The best RI values and F_{macro} for both datasets are obtained for $n \geqslant 30$, the maximum is respectively 0.994 and 0.899 for PLAID, and respectively 0.856 and 0.847 for WHITED. It is thus important that the VI images have a sufficient fine resolution. The high RI values confirm the capability of the siamese neural network to learn a feature space where the 10 clusters are well separated and the high F_{macro}-values confirm the capability to detect new (unidentified) appliances.

Fig. 10 shows an example of the learned three dimensional feature space ($n_{out} = 3$ for the sake of easy plotting) for PLAID when using ten appliances for learning. The hold out appliance is the microwave. Each axis of the plot represents one element of the neural network’s output vector with size n_{out}. The ten appliances used for learning form ten clusters, the label of each of them is written next to it. The samples belonging to the microwave, i.e., the hold out appliance, do not belong to any cluster and neither form a cluster as the siamese neural network is not trained on them. They are spread around, and get the label ‘unknown’. Note, that the learned feature space in Fig. 10 is different from the one in Fig. 7, because the samples used to learn the space are different (in Fig. 10, the samples for the microwave are not used). To know which appliances are mixed up, a confusion matrix is created: Fig. 11 reports for each appliance type (row index) the number of labels that were correctly predicted or confused with other appliances (column index). The values in the matrix are absolute and the colors represent the relative value per row (thus per appliance). It can be seen that if the laptop is not used for training, it is put in the cluster containing the CFL examples and the other way around. A visualization for WHITED is not given as the number of appliances is too big to create a clear plot.

5.5. Scenario 3

To test whether the method generalizes well, training and testing is performed on different houses. As recommended in [8], leave-one-house-out cross-validation is used. Note, no appliances are left out. For PLAID, this means that 54 houses are used for the training set and 1 for test set. For WHITED, the annotation of measurement locations is not available. Houses are created artificially by assigning each appliance of each appliance type randomly to one house. The total number of houses is set at 9, which corresponds to the maximum number of appliances per appliance type.

It will now be validated whether (1) the learned feature space using the 54 houses separates the appliance clusters sufficiently, and (2) the appliances of the test set are projected on the correct cluster. The first criterion is validated by calculating the RI and the second by counting the amount of test samples that are assigned to the correct cluster, to a wrong cluster or get the label unidentified. Three accuracy measures are defined: (1) the positive rate defined as the percentage of samples assigned to the correct cluster, (2) the negative rate defined as the percentage of samples assigned to the wrong cluster, and (3) the unidentified rate defined as the percentage of samples labelled as ‘unidentified’. These quantities add up to 100%.

To be able to calculate the first and second rates, the appliances labeling provided in the PLAID dataset is used. As leave-one-house-out

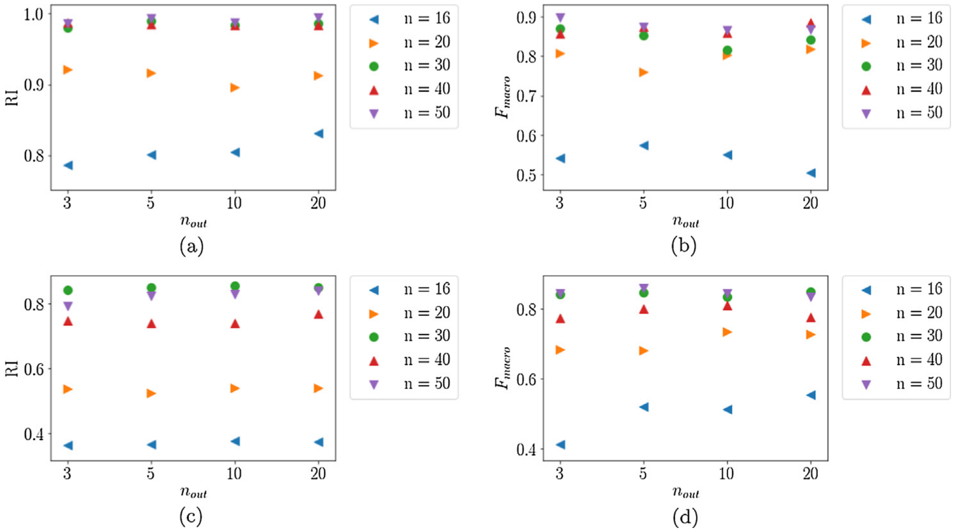


Fig. 9. The RI values and F_{macro} for different n_{out} and n values when leave-one-appliance-out cross-validation is used for PLAID (a)–(b), and WHITED (c)–(d).

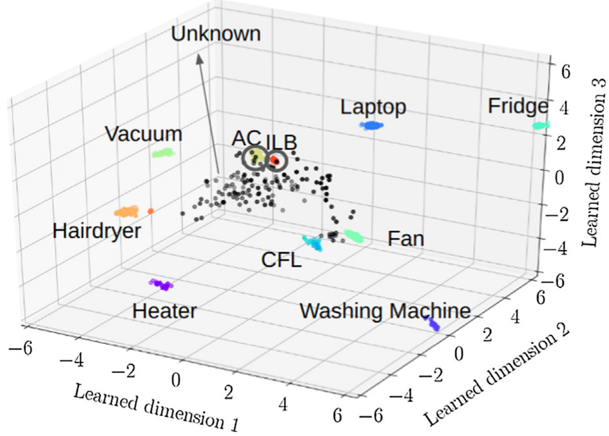
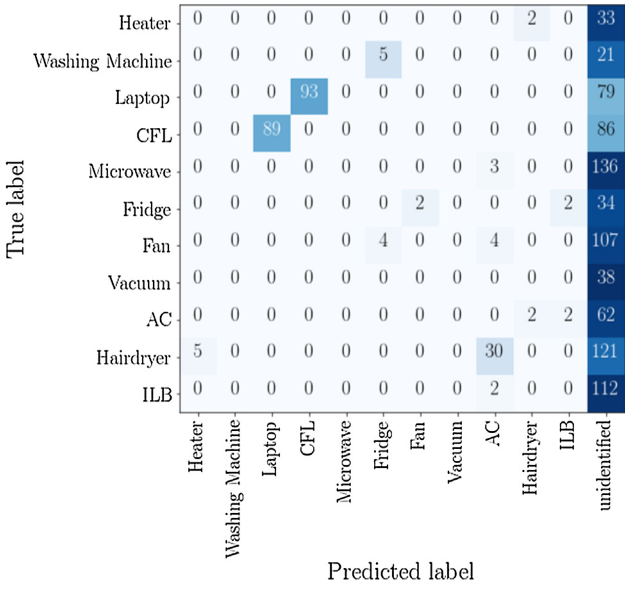


Fig. 10. The 10 clusters and unidentified points found by DBSCAN in the learned feature space representing respectively the 10 seen appliances and the 1 hold out appliance.



	Heater	Washing Machine	Laptop	CFL	Microwave	Fridge	Fan	Vacuum	AC	Hairdryer	ILB	unidentified
Heater	0	0	0	0	0	0	0	0	0	2	0	33
Washing Machine	0	0	0	0	0	5	0	0	0	0	0	21
Laptop	0	0	0	93	0	0	0	0	0	0	0	79
CFL	0	0	89	0	0	0	0	0	0	0	0	86
Microwave	0	0	0	0	0	0	0	0	3	0	0	136
Fridge	0	0	0	0	0	0	2	0	0	0	2	34
Fan	0	0	0	0	0	4	0	0	4	0	0	107
Vacuum	0	0	0	0	0	0	0	0	0	0	0	38
AC	0	0	0	0	0	0	0	0	0	2	2	62
Hairdryer	5	0	0	0	0	0	0	0	30	0	0	121
ILB	0	0	0	0	0	0	0	0	2	0	0	112

Fig. 11. The confusion matrix when leave-one-appliance-out cross-validation is used.

cross-validation is performed on 55 houses, there are 55 test scores (one for each house). To obtain the final test result, these values are averaged.

Figs. 12a and 13a show the RI values for different n_{out} and n values for respectively PLAID and WHITED. Again, for both datasets, it can be concluded that the clusters are separated sufficiently for higher n values. Changing n_{out} has little impact on the result.

Fig. 12b, c and d show the three accuracies defining how much test samples are respectively assigned to the correct, incorrect or no cluster for PLAID. Again, changing n_{out} does not change the results significantly. When using $n = 16$, a large part of the test samples (> 50%) are assigned to the correct cluster, but also a significant part of them (∽ 22.5%) are classified incorrectly. When using a larger $n = 50$, the amount of correctly assigned test samples is much smaller, namely around 38.7%, but so are the incorrectly assigned samples (~1.3%). As a consequence, the amount of samples labeled as unidentified is larger (around 60%).

Fig. 13b, c and d show the three accuracies defining how much test samples are respectively assigned to the correct, incorrect or no cluster for WHITED. Again, changing n_{out} does not change the results significantly. For WHITED, the highest obtained positive rate is 73% for when $n = 50$. The corresponding negative rate and unidentified rate are 6%, and 20%. So the parameter setting leading the best positive rate, also leads to the best negative rate and a good unidentified rate. This in contrast to PLAID.

Although the RI values for WHITED are lower than those for PLAID, the third scenario obtains the best result for WHITED. An observation that can explain this, is that the different appliances for each appliance type are more alike in WHITED than in PLAID. Thus, an appliance of a known appliance type is more likely to fall in the center of the cluster. While for PLAID, these appliances of a known appliance type will drift around the cluster. This can be checked by calculating the rank of the correct cluster for each test sample labeled as unidentified. First, all clusters are ranked by calculating the distance d_{ij} from the given sample i to each of the clusters j, and sorting these distances in ascending order. Ideally, the test sample's appliance cluster should be on the first position. Fig. 14a shows the rank that is averaged over the 55 folds. For all the different parameter combinations, this result shows that the correct cluster is the closest or second closest cluster. Fig. 14b shows the box plot of the rank of the correct cluster for all the test samples labeled as 'unidentified' in all the test houses (1074 ranks) when $n = 50$. As these box plots show, their is very little variance on the rank of the correct cluster, making the average a valid measure. This result is important because it implies that if the method labels a sample as unidentified, it

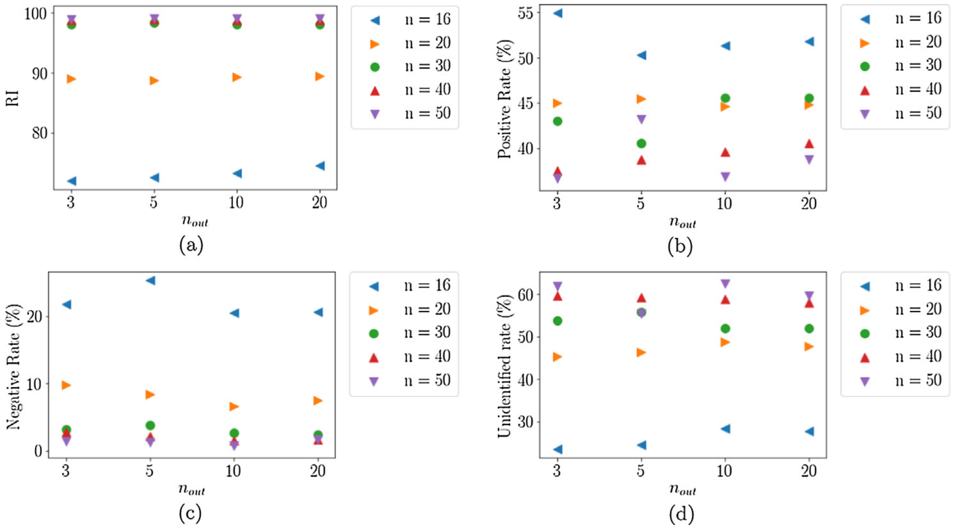


Fig. 12. The RI values, and the positive, negative and unidentified rate for PLAID for different n_{out} and n values when leave-one-house-out cross-validation is used.

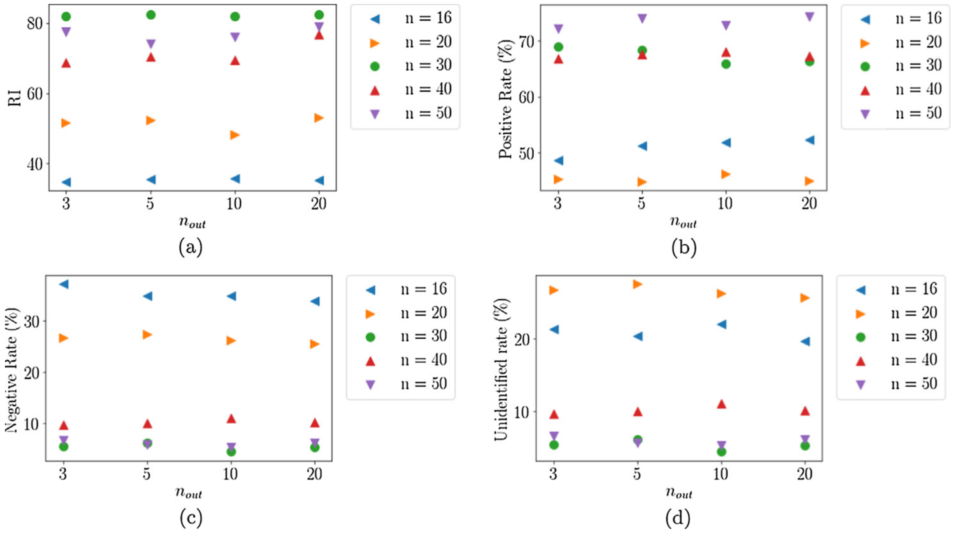


Fig. 13. The RI values, and the positive, negative and unidentified rate for WHITED for different n_{out} and n values when leave-one-house-out cross-validation is used.

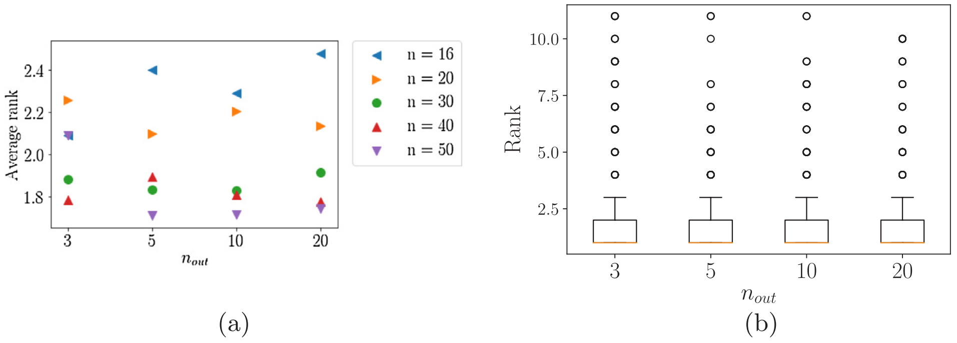


Fig. 14. (a) The average rank of the correct cluster for the samples of PLAID labeled as 'unidentified' for different n_{out} and n values when using leave-one-house-out cross-validation. (b) The box plots of the rank of the correct cluster for all samples of PLAID labeled as 'unidentified' for $n = 50$ and different n_{out} values.

can query the user to confirm if it is a new appliances or not, while immediately suggesting a label (e.g., listing the top-3 of the ranked list). If the sample originates from an existing appliance, the correct appliance (cluster) will be in this top.

5.6. Quasi real-time solution

The presented results were obtained in an offline manner. However, in practice, this method needs to be used in an online manner (quasi real-time). For an operational deployment, we start from a trained siamese neural network. This implies a learned feature space, and

Table 2
The required time expressed in seconds to train the method, and to execute the different steps to evaluate one sample in respectively the PLAID and WHITED dataset.

		PLAID	WHITED
Training	Siamese NN (one epoch)	1022.44	300.16
	DBSCAN	0.0088	0.0067
Testing	Creating VI image	0.0018	0.002
	Transform VI image	0.0013	0.0014
	Labeling	0.0001	$9.9\ 10^{-5}$

available appliance clusters (for the already known appliances). To apply this, a system needs to perform the following steps:

1. real-time event detection [34] to identify the activation or deactivation of an appliance,
2. extraction of the appliance specific current and voltage signal when it reached steady-state behaviour, as discussed in [35],
3. creation of the binary VI image, as described in [8],
4. transformation of the image to the learned feature space by feeding the image as input to the convolutional neural network part of the siamese neural network,
5. in this feature space, calculation of the distance of the detected device to all known appliances.
   (a) If the distance to all of the clusters is bigger than a threshold, then mark the device as 'unidentified', else
   (b) label the device as the type of the closest cluster.
6. saving the VI image for (re) training the siamese neural network later,
7. (optional) if the detected appliance is classified as 'unidentified', asking feedback to the user in order to obtain a correct label (although this label is not required for the continuation of the algorithm).Table 2 shows that every step takes less than a second for one sample, and can thus be realized in quasi real-time. The experiments were executed on an Intel(R) Xeon(R) CPU with 20 cores and 128 GB ram. It is also seen from Table 2 that retraining the siamese neural network is more time consuming and cannot be performed real-time. Therefore, it is advisable to retrain the siamese network daily, e.g., every night. To bootstrap the system, we can start from a dataset obtained from other households. If this is not possible, the user will have to label each event first.

6. Conclusion

This paper presents a novel method for appliance classification and detection of unidentified appliances in non-intrusive load monitoring. Both rely on a learned vector representation function (a trained cNN), which takes as input an image representation of a VI trajectory. To learn this representation, training data in the form of such VI images, in labeled pairs of the same/different appliance instances is needed. A siamese neural network is trained on these paired instances outputting a pair of lower-dimensional vector representations, such that the distance between these two vectors is lower or higher than a threshold for respectively same/different appliance. In this newly learned feature space, DBSCAN is performed, allowing us to assign test samples to clusters or label them as unidentified. Benchmarking on PLAID and WHITED shows that an F_{macro}-measure of 0.90 and 0.85 is obtained when detecting unknown appliances. Furthermore for PLAID, if unseen instances of known appliance types are given as input: 39% is classified correctly, 1% incorrectly and 60% as unidentified. However for the appliances classified as unidentified, a ranking of good suggestions can be made concerning the cluster it belongs to, as the expected rank of the correct cluster is 1.75. For WHITED, if unseen instances of known appliance types are given as input: 73% is classified correctly, 6% incorrectly and 20% as unidentified. Additionally, we benchmarked the method on subsampled data of PLAID and WHITED, and the results pointed out that the performance remained unchanged for frequencies as low as 0.5 kHz.

References

[1] EU climate strategies & targets. https://ec.europa.eu/clima/policies/strategies/2030_en, accessed: 2017-04-12.
[2] Ehrhardt-Martinez K, Donnelly KA, Laitner S, et al. Advanced metering initiatives and residential feedback programs: a meta-review for household electricity-saving opportunities. Washington, DC: American Council for an Energy-Efficient Economy; 2010.
[3] Iyer SR, Sankar M, Ramakrishna PV, Sarangan V, Vasan A, Sivasubramaniam A. Energy disaggregation analysis of a supermarket chain using a facility-model. Energy Build 2015;97:65–76.
[4] Armel KC, Gupta A, Shrimali G, Albert A. Is disaggregation the holy grail of energy efficiency? The case of electricity. Energy Policy 2013;52:213–34.
[5] Faustine A, Mvungi NH, Kaijage S, Michael K. A survey on non-intrusive load monitoring methodies and techniques for energy disaggregation problem. arXiv preprint 1703.00785.
[6] Zeifman M, Roth K. Nonintrusive appliance load monitoring: review and outlook vol. 57. IEEE; 2011.
[7] Zoha A, Gluhak A, Imran MA, Rajasegarar S. Non-intrusive load monitoring approaches for disaggregated energy sensing: a survey. Sensors 2012;12(12):16838–66.
[8] Gao J, Kara EC, Giri S, Bergés M. A feasibility study of automated plug-load identification from high-frequency measurements. 2015 IEEE global conference on signal and information processing (GlobalSIP). IEEE; 2015. p. 220–4.
[9] Du L, He D, Harley RG, Habetler TG. Electric load classification by binary voltage–current trajectory mapping. IEEE Trans Smart Grid 2016;7(1):358–65.
[10] Hart GW. Nonintrusive appliance load monitoring. Proc IEEE 1992;80(12):1870–91.
[11] De Baets L, Ruyssinck J, Develder C, Dhaene T, Deschrijver D. On the bayesian optimization and robustness of event detection methods in nilm. Energy Build 2017;145:57–66.
[12] Henao N, Agbossou K, Kelouwani S, Dubé Y, Fournier M. Approach in nonintrusive Type I load monitoring using subtractive clustering. IEEE; 2015.
[13] Wichakool W, Remscrim Z, Orji UA, Leeb SB. Smart metering of variable power loads. IEEE Trans Smart Grid 2015;6(1):189–98.
[14] Chang H-H, Chen K-L, Tsai Y-P, Lee W-J. A new measurement method for power signatures of nonintrusive demand monitoring and load identification. IEEE Trans Ind Appl 2012;48(2):764–71.
[15] Hassan T, Javed F, Arshad N. An empirical investigation of VI trajectory based load signatures for non-intrusive load monitoring. IEEE Trans Smart Grid 2014;5(2):870–8.
[16] Altrabalsi H, Stankovic V, Liao J, Stankovic L. Low-complexity energy disaggregation using appliance load modelling. AIMS Energy 2016;4(1):884–905.
[17] Altrabalsi H, Liao J, Stankovic L, Stankovic V. A low-complexity energy disaggregation method: performance and robustness. 2014 IEEE symposium on computational intelligence applications in smart grid (CIASG). IEEE; 2014. p. 1–8.
[18] Zhang C, Zhong M, Wang Z, Goddard N, Sutton C. Sequence-to-point learning with neural networks for nonintrusive load monitoring. arXiv preprint 1612.09106.
[19] Nguyen M, Alshareef S, Gilani A, Morsi WG. A novel feature extraction and classification algorithm based on power components using single-point monitoring for NILM. 2015 IEEE 28th canadian conference on electrical and computer engineering (CCECE). IEEE; 2015. p. 37–40.
[20] Basu K, Hably A, Debusschere V, Bacha S, Driven GJ, Ovalle A. A comparative study of low sampling non intrusive load dis-aggregation. IECON 2016-42nd annual conference of the IEEE industrial electronics society. IEEE; 2016. p. 5137–42.
[21] Zhao B, Stankovic L, Stankovic V. On a training-less solution for non-intrusive appliance load monitoring using graph signal processing. IEEE Access 2016;4:1784–99.
[22] Goncalves H, Ocneanu A, Berges M, Fan R. Unsupervised disaggregation of appliances using aggregated consumption data. In: The 1st KDD workshop on data mining applications in sustainability (SustKDD); 2011.
[23] Wang Z, Zheng G. Residential appliances identification and monitoring by a nonintrusive method. IEEE Trans Smart Grid 2012;3(1):80–92.
[24] Nielsen MA. Neural networks and deep learning. Determination Press; 2015.
[25] Koch G. Siamese neural networks for one-shot image recognition [Ph.D. thesis]. University of Toronto; 2015.
[26] Bromley J, Bentz JW, Bottou L, Guyon I, LeCun Y, Moore C, Säckinger E, Shah R. Signature verification using a 'siamese' time delay neural network. IJPRAI 1993;7(4):669–88.
[27] Hadsell R, Chopra S, LeCun Y. Dimensionality reduction by learning an invariant mapping. 2006 IEEE computer society conference on computer vision and pattern recognition, vol. 2. IEEE; 2006. p. 1735–42.
[28] Ester M, Kriegel H-P, Sander J, Xu X, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In: KDD, vol. 96; 1996. p. 226–31.
[29] 2014 SIGKDD test of time award. http://www.kdd.org/News/view/2014-sigkdd-test-of-time-award, accessed: 2017-04-10.
[30] Gao J, Giri S, Kara EC, Bergés M. PLAID: a public dataset of high-resoultion electrical appliance measurements for load identification research: demo abstract. Proceedings of the 1st ACM conference on embedded systems for energy-efficient buildings. ACM; 2014. p. 198–9.
[31] Kahl M, Haq AU, Kriechbaumer T, Jacobsen H-A. WHITED-a worldwide household and industry transient energy data set. In: 2016 Proceedings of the 3rd international workshop on non-intrusive load monitoring (NILM); 2016.
[32] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. Imagenet large scale visual recognition challenge. Int J Comput Vision 2015;115(3):211–52.
[33] Makonin S, Popowich F. Nonintrusive load monitoring (nilm) performance evaluation. Energy Effi 2015;8(4):809–14.
[34] Nguyen TK, Dekneuvel E, Jacquemod G, Nicolle B, Zammit O, Nguyen VC. Development of a real-time non-intrusive appliance load monitoring system: an application level model. Int J Electric Power Energy Syst 2017;90:168–80.
[35] Wang AL, Chen BX, Wang CG, Hua D. Non-intrusive load monitoring algorithm based on features of v–i trajectory. Electric Power Syst Res 2018;157:134–44.