



Image-based mains signal disaggregation and load recognition

Liston Matindife¹ · Yanxia Sun¹ · Zenghui Wang²

Received: 23 July 2020 / Accepted: 7 December 2020 / Published online: 5 January 2021
© The Author(s) 2021

Abstract

The mains signal is a complex fusion of various electrical equipment load signals in a building. In the non-intrusive load monitoring recognition, our main aim is to be able to extract as much load features as possible from the complex aggregate mains signal in a simpler way through a computer vision-based approach as opposed to the powers series signal approach. Power series methods, which are one dimensional in nature, suffer from poor aggregate and load signal feature localization necessitating a larger training dataset spanning very long time periods and normally require signal formatting and pre-processing. We use Gramian angular summation fields to transform the power series into a reduced image dataset that contains a rich set of localized signal features. A computer vision approach allows us to capture as much information as possible, and then propose an image-based mains load recognition system with high performance. In this paper for the entire recognition system, we use convolutional neural networks that very well adapted to vision recognition. The load signal image disaggregation is achieved through the powerful stacked denoising autoencoder noise extraction network. To test the proposed system, some simulations and comparisons are carried out and the results show that our easier to handle method can achieve acceptable performance.

Keywords Image disaggregation · Mains supply load recognition · Convolutional neural network · Stacked denoising autoencoder · Image recognition

Introduction

The proliferation of using power systems loads in buildings has resulted in high energy demand within the buildings. With more and more users and more and more loads there is a need to manage the energy within the buildings. The main focus point of mains disaggregation and load recognition is to achieve an automatic energy management in mainly residential and commercial buildings as these are the high consumers of electrical energy. However, there are various other electrical load equipment usage expectations by different users. The electrical mains supply signal on which the energy management system can be developed is a fusion into a complex form of the various electrical equipment load

signals within a building. Through the non-intrusive-load-monitoring (NILM) [1–4] method we are able to extract each load signal from this composite thereby establishing the equipment ‘s’ exact operational status. The contemporary NILM mains power series signal disaggregation and load recognition approach focuses on deep learning (DL) algorithms that are modeled on speech recognition and natural language processing recognition systems. Some examples of NILM power series based recognition systems include: (1) the “sequence-to-point learning” where the output is made up of one point of the target appliance and input is made up of a window of the aggregate signal as raw data, (2) one-dimensional convolutional differential input systems, and (3) stacked denoising autoencoders (sdAEs) with the ability to reconstruct a good signal from a composite of noise and signal [4–9]. The NILM method has traditionally been based on the power series format of the equipment signal [6, 7, 10] in labeled or unlabeled form, often with a detailed incorporation of event detection mechanism [11, 12]. The appliance features that are used in NILM systems broadly fall in the following categories of steady state (power change, time and frequency domain voltage-current ($V-I$), $V-I$ trajectory),

✉ Zenghui Wang
wangzengh@gmail.com

¹ Department of Electrical and Electronic Engineering Science, University of Johannesburg, Auckland Park 2006, South Africa

² Department of Electrical and Mining Engineering, University of South Africa, Florida 1710, South Africa

transient state (transient power, start-up current waveforms, voltage noise), combined steady and transient states features, and features obtained or inferred from the behavior of the appliance [13].

In NILM recognition systems, power series spanning over long time periods are often required to avail sufficient features for model training, since power series methods suffer poor signal feature localization, and normally require involved signal formatting and pre-processing. A shapelets learning method that can benefit the NILM power series based recognition scheme is proposed by [14] to improve on the recognition of general time-series with very limited data samples. These shapelets represent tendencies in the signal thereby placing the signal in a certain class. However, the shapelets method is still power series signal based. In this paper for improved recognition based on the powerful computer vision model(s) we change the power series feature space to image space. The image equivalent to the power series contains a rich set of localized signal features. We transform the power series into image through the use of the Gramian angular summation fields (GASF). However, it is also possible to encode the power series to image using Gramian angular difference fields (GADF) and Markov transition fields (MTF). The main advantage of Gramian angular fields (GAF) over other time series visualization methods is that we can readily reconstruct the power series from the image parameters [15]. Some researchers [16, 17, 22] have proposed image-based NILM recognition systems with varying degrees of success. However, the image-based approach was mainly implemented in the classification stages rather than the entirety of the NILM recognition to include the disaggregation.

Having reviewed the related literature we propose the development and improvement of the image-based NILM recognition system. In this paper, we introduce an improved feature extraction image based approach that performs both disaggregation and classification of power systems load signals via less complex deep learning model configurations having reduced computation times. The developed system is completely evaluated in the laboratory setup. We then propose the installation of the designed NILM recognition system at the mains powerpoint into the building housing the appliances as a practical implementation of the system. The appliance classification is achieved through the Oxford Visual Geometry Group (VGG) convolutional neural network (VGG-CNN) [16] due to its very high image classification count. The load signal image disaggregation is achieved through the powerful stacked denoising autoencoder noise extraction network applied to images. In this study, we generated our own dataset from three mains lamps, a refrigerator and a microwave oven. In future, we can extend the image-based NILM recognition strategy to the recognition of multi-state appliances.

In literature, the traditional two-dimensional (2-D) convolutional neural network (CNN or ConvNet) is a common feature in most if not all NILM image-based recognition systems. To improve the performance of NILM image-based designs it is necessary to modify the basic CNN structure [17–22]. Based on a twenty-one appliance dataset, the authors of [17], proposed a 2-D CNN composed of a residual model and a Batch Normalization layer to correct the gradient disappearance issue during training. For the transformation of the time series signals to 2-D GADF images in their NILM recognition system, in [17], the authors recommended GADF over GASF or MTF images as GADF capture and represent more signal event/timing information than the other two. However, there is a need to improve the NILM recognition system performance of [17] which hovered at 97.2%.

The authors in [18] proposed a GASF image-based CNN NILM disaggregation strategy for the standard Dataport dataset. The model in [18] was able to achieve reasonable disaggregation with image pixel sizes of 30×30 for microwave and 100×100 for air conditioner. However, the disaggregation performance relative error in total energy varied from 14% for microwave to 32% for the air conditioner. Clearly, there is a need to improve the NILM disaggregation performance in [18]. The event-driven NILM recognition method proposed in [19] captures event-based information that includes establishing the signal's zero-crossing point, the similarity between current signals, threshold measure and point at which event starts and stops. All these event current characteristics are converted to gray-scale images as an input of a VGG-16 CNN model. The method in [19] achieved high NILM image-based recognition performance for a considerably reduced signal dataset when the number of appliances is few. However, there is a need to further improve the method in [19], as the NILM recognition accuracy degrades with an increase in the number of appliances. Furthermore, image-based event algorithms improve the complexity of the NILM design.

Voltage-current ($V-I$) trajectories constitute a form of 2-D NILM signature recognition scheme [20]. The $V-I$ trajectories provide a characteristic image for each appliance. The image is then recognized through the Hierarchical clustering classifier [20]. The authors of [4, 5] developed a much more robust NILM vision-based $V-I$ trajectories recognition method based on the convolutional and Siamese neural networks. The Siamese neural network is composed of two similar CNN networks in parallel feeding one output label. The inputs of these networks are single identical images. Siamese neural networks can also be successfully implemented in one-shot learning [5]. The aim is to find the similarity between two inputs for example that of the ground truth signal and the disaggregated signal. The constructive loss function gives a quantitative measure of the relationship

between the Siamese network inputs. A clustering algorithm known as the density-based spatial clustering of applications with noise (DBSCAN) [5] is then used to place the inputs into their classes. However, in [4, 5] the F1-measure which requires improvement was low for similar signature appliances. The authors in [21] proposed an image-based NILM recognition approach premised on the vector projection classification (VPC) technique that was formally applied to human faces recognition. In this case, appliance data images are projected onto some 2-D vector surface and their similarity noted, the closer the images to each other the more probable that there are in that class and hence recognized.

In [22], the authors proposed to represent event-based NILM $V-I$ appliance features in image form fusing the weighted recurrence graphs (WRGs). Traditional $V-I$ models are capable of only representing the phase relationships with the exclusion of the signal magnitude of the appliances. According to [22], the traditional $V-I$ trajectories approach is incapable of extracting adequate $V-I$ trajectories information on a purely resistive load. However, the WRGs approach is capable of combining and representing both the signal magnitude and $V-I$ trajectories into a single image which is then processed through an image-based CNN. By so doing, extracting adequate $V-I$ trajectories information from purely resistive loads can be addressed. Although the method in [22] is capable of very high NILM recognition performance there are some appliances that it wrongly identifies.

We have shown the diversity of NILM image recognition methods that often achieve high performance. The continued development of NILM image recognition systems has been made possible by the technological advancements in computing that has allowed for the development of deep machine learning algorithms with computer vision capable of outperforming human biological-based vision. One deep-learning image detection algorithm and its variants, which stands out of the rest and are used in most image recognition systems, are the CNN and its variants. The CNN has enhanced image feature extraction capabilities that allow it to achieve advanced levels of image recognition [23–25]. In this paper, we propose an improved NILM image disaggregation framework that is based on the stacked denoising autoencoder (sdAE) using CNN layers. The effectiveness of an image recognition system is in its ability to obtain a clean image from a poor and noisy representation of the image. Although a number of image cleaning techniques have been proposed [26], with the deep image denoising concept pioneered in 2015 [27], the CNN based sdAE has achieved very high image cleaning performance [26]. Hence, we aim to exploit this property of the sdAE to obtain a clean appliance signal image from the mains supply signal composite image. Furthermore, we aim to address some of the image-based NILM deficiencies [17–22]. As the authors in [18] we proposed GASF generated images for the disaggregation

strategy, however, in our case we use an in-house generated dataset and go a step further to include the image-based equipment classification part which was not done in [18] who use a standard dataport dataset. In the final analysis, we compare the performance of our proposed image recognition system to that of a power series one, also based on the convolutional neural network. The procedure involves measuring the current, real power and power factor load parameters for the aggregate and each appliance power series based signal, and converting each parameter power series into an image representation. We performed rigorous NILM recognition experimentation with the images generated from all the three signal parameters. The current and power signals do, in fact, individually provide all the features required to provide unique signal identity. However, it is also possible to provide signal identification by considering the PF. We get a boost on the signal recognition performance if we consider an increased dataset that includes current and power factor or active power and power factor. We then train the proposed sdAE disaggregation and VGG-CNN classification networks. Finally, we perform image-based disaggregation and recognition of each appliance based on the power series images. We make the following contributions in our study:

- Development and improvement of the NILM recognition scheme by basing it on a powerful computer vision appliance signal disaggregation and classification technique.
- Compare the performance of our proposed image based NILM recognition scheme to that of the power series signal system.

The remaining sections are structured as follows: “**Methodology**” details the design of the proposed image-based non-intrusive load monitoring system. The disaggregation is achieved through a number of trained stacked denoising autoencoders (sdAE) equal to the number of target mains loads and the classification through a single multiclass trained deep convolutional neural network (DCNN). We also show the intended application of the designed image-based NILM recognition system. We detail the experimental setup in relation to the creation of our in-house laboratory dataset from power series to image form, performance measure, proposed method pseudo code, model training/testing approach framework and procedure. A breakdown of the model architectures in terms of the deep learning network layers, and the comparison or relationship between the encoding, decoding, ConvNet classification and power series classification are also given here. “**Discussion of experimental results**” gives an in-depth presentation and analysis of the results. “**Conclusion**” gives a conclusion of the developed system. We also give an insight into future work related to the outcomes of this paper.

Methodology

Proposed overall topology

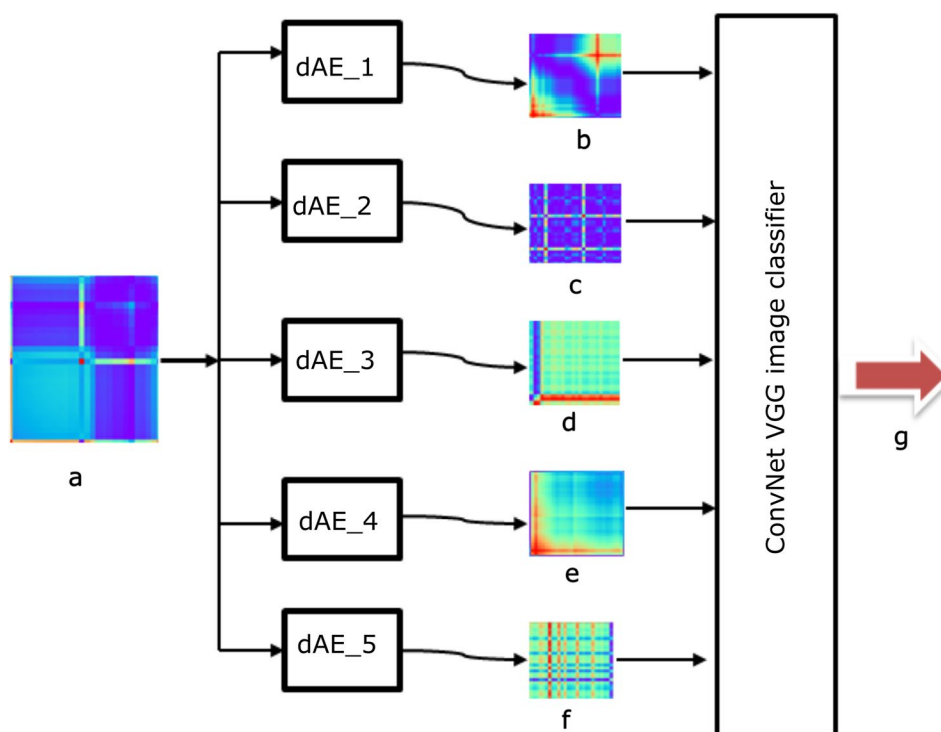
The proposed topology in Fig. 1 is made up of two parts, the disaggregation and classification. The disaggregation is made up of five sdAE networks, whilst the classification is made up of one ConvNet VGG network. The aggregate signal image is input into five trained sdAE networks, each is capable of disaggregating only one target appliance signal image. The output of each sdAE is a clean target appliance signal image. The image is the input into the trained VGG classifier for recognition.

In the classification part, we train the model to recognize and classify only the ground truth signature images of the appliances. In this case, we consider only the refrigerator and microwave oven input images. However, we can generate relevant signature images for the other appliances in the entire experiment. Both the disaggregation and classification networks are built around the CNN. The CNN can extract detailed image features and reduce the overall dimension of an image but preserve the image identity through the linear convolution of the input image. The CNN employs a number of filters whose dimensions are much smaller than the image to scan the entire image at intervals known as strides, thereby obtaining a representative mapping of these scanned areas. Nonlinearity is introduced into the convolution result through the application of a Rectified Linear Unit

(ReLU) operation which effectively removes all negatives in the result. The produced CNN + ReLU feature image is then passed through a pooling layer to reduce the dimensionality of the convolution result but maintaining the essential parts of the input information. For image recognition, the CNN uses max or sum pooling method. [4, 23–25]. To increase the number of detected image features the CNN requires an increase in the number of filters connected in parallel, with each filter detecting a specific image feature. The CNN can be made deeper by adding successive convolution layers and pooling layers to extract as much information as possible from the data. The pooling layers introduce blurring of the image hence it needs to have deeper networks to extract as much relevant information as possible. Figure 2 shows a 28×28 input image, the convolved output image and finally the image from a 2×2 max pooling based on 16 filters.

In the power series-based NILM recognition the network input is the aggregated power series signal and the target is the specific appliance signature. In the proposed system, the number of disaggregation networks is equal to the number of appliances under test, where the input power series is equal to the entire appliance activation. In disaggregation, the output power series length is also equal to the entire appliance activation. In our proposed method, the disaggregation output is the image equivalent of the power series output. Instead of power series based partial disaggregated signals (that are combined through some reconstruction filter or through addition and finding the mean) defined by the number of sliding windows, our proposed

Fig. 1 Proposed complete NILM recognition system showing five dAE networks and one ConvNet VGG classifier. The following are identified **a** aggregate input, **b** FR, **c** MW, **d** L1, **e** L2, and **f** L3 target images, with **g** the load identity output



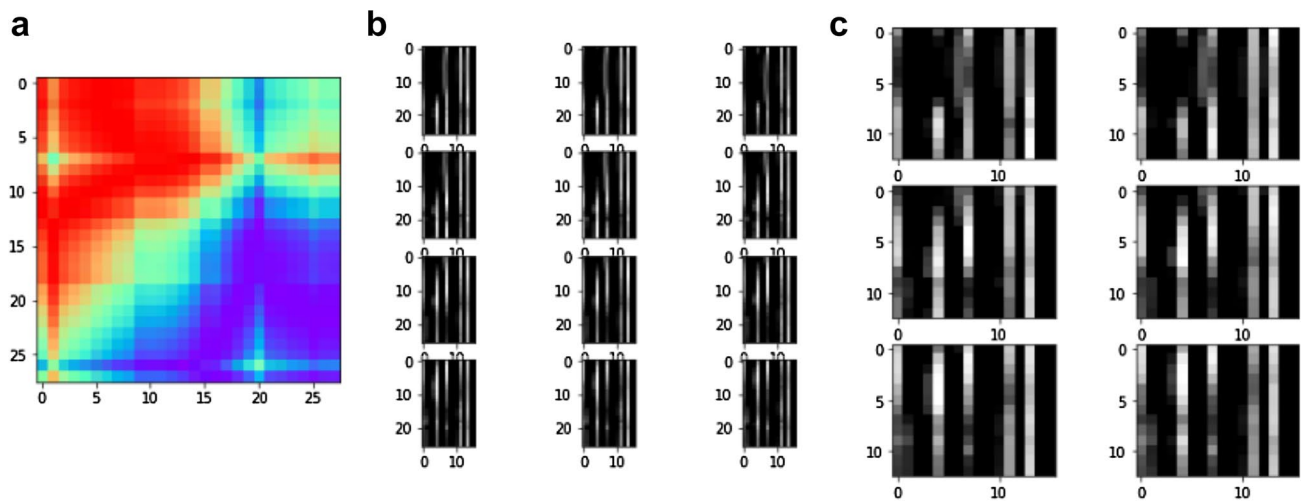


Fig. 2 Convolution on refrigerator GADF image...**a** Input image, **b** convolution + ReLU image, and **c** blurred first max pooling resultant image

system outputs an image representing the entire ground truth activation characteristics of the disaggregated image equivalent to the signature. To improve gradient convergence and avoid instability it is necessary to normalize all power series data and then apply standardization [zero mean (μ) and unit standard deviation (σ)] to the data. The disaggregated signals are then fed into a trained multi-class power series classification network. Long-short-term-memory (LSTM) recurrent networks that find wide application in speech recognition and language processing are highly adaptable to time-series disaggregation. Furthermore, whilst ConvNets are highly adapted to spatial based recognition they can also be used in one-dimensional (1D) power series univariate and multivariate based NILM disaggregation and classification systems [6] with acceptable performance. Power series deep learning NILM recognition systems are often based on (1) combined convolutional and recurrent neural network (CNN–RNN), and (2) autoencoder (AE) [7, 10, 13] that are well adapted to complex feature extraction, sequence prediction and signal reconstruction, all requirements that are crucial in signal disaggregation. In applications where we need to detect feature trends within the data (fixed sequence length) without worrying about the specific location of that feature we use one dimensional (1-D) CNN. RNN that is based on memory cells can direct the output predictions to be in the order determined by the position of the input signal elements. However, due to the vanishing gradient problem of the RNN, an enhanced form of the RNN known as the long-short-term-memory (LSTM) network is used instead [28]. In the backpropagation, weights are updated according to the gradient descent where a vanishing gradient deprives layers close to the input of error signal making these layers less effective in training, whilst an exploding gradient error signal causes instability in the same layers [29]. In the

multi-layer perceptron (MLP) as the hidden layers go wider and deeper, a number of issues arise. Wider implies more weights, hence strenuous computations. Deeper implies a vanishing and exploding gradient. The autoencoder (AE), which is made up of the same number of input neurons as output neurons and having a significantly reduced deep layer count that form an extension of the input, can address the pitfalls of the deep MLP. Disaggregation by ‘denoising’ the unwanted parts of the aggregate signal is an effective way of extracting the required load signal [6]. Our proposed system uses 2-D CNN based classification as opposed to 1-D CNN classification in the power series system.

The proposed system samples data from the common mains power cable supplying three mains lamps, a refrigerator and a microwave oven in the house. The hardware/software components include digital signal processing processors, main system processor, embedded system development board/platform, IoT module and python with Keras deep learning library. There is also a need to convert the high-level language to low-level format or machine code for loading the NILM program into the embedded system. The power supply for the whole recognition unit is tapped from the mains power cable. In implementation we consider both manual and online capture of signal information for training and disaggregation, respectively.

Disaggregation framework

Inspired by the ability of autoencoders to reconstruct a good signal from a composite of noise and signal, an NILM disaggregation system based on the stacked image denoising autoencoders (dAEs) [9] is proposed. The dAE will effectively disaggregate the required load image from a noisy environment due to other (aggregate) loads from

the aggregate image. We obtain the full benefits of the dAE by developing stacked dAEs which are basically deep dAE structures. By implementing stacked dAEs we obtain a better generalization of the recognition system. Our proposed stacked dAE recognition system is given in Fig. 3 with a number of hidden layers.

The aggregate signal $x(t)$ can be represented in-terms of appliance j signature $z_j(t)$ and an overall noise term due to the other appliances $z_i(t)$ and a spurious noise term $e(t)$ as [30]

$$x(t) = z_j(t) + v_j(t) \quad \text{for } j = 1, 2, \dots, N, \tag{1}$$

where

$$v_j(t) = \sum_{i=1}^N z_i(t) + e(t) \quad i \neq j. \tag{2}$$

The dAE will remove the $v_j(t)$ term from the aggregate signal so that there remains with only the appliance j signature $z_j(t)$ term. The dAE comprises of an aggregate input x_i followed by an encoder which gives an internal representation of the input to an encoding hidden layer y_h and then a decoder which moves this internalized representation to the target output z_i provided $i > h$ the number of neurons in the respective layers. These are actually back to back connected full networks where the first full network based on CNN incorporates max pooling and the second full network incorporates up-sampling [9, 31–33]. Normally during training the network a Gaussian or Salt-and-Pepper noise is added to the input to give a noisy term x' . Then a nonlinear encoding layer y [9] is given in Eq. 3 as

$$y = f_{\theta}(x') = \sigma(Wx' + b), \tag{3}$$

where b is an encoding layer bias, W is the $i \times h$ weight matrix, and σ is the ReLU activation function. The mapping to z from the output y is

$$z = g_{\theta'}(y) = s(W'y + b'), \tag{4}$$

where b' is decoding layer bias, W' is the $h \times i$ weight matrix that translates to W^T , s is a softplus activation function. $\theta = \{W, b\}$ for encoding layer y_h and $\theta' = \{W', b'\}$ for decoding layer z_i . For training and optimizing the parameters $\Theta = \{W, b, b'\}$ we apply the objective loss function

$$L(\Theta) = \sum_i \|v_i - g_{\theta'}(f_{\theta}(x'))\|_2^2, \tag{5}$$

where v_i is the clean signal.

Classification framework

The classification model was premised on a simplified three-section Oxford Visual Geometry Group (ConvNet VGG) in [16]. This is a multilayer very deep CNN structure that has achieved a very high level of multiclass recognition and good generalization of a large varying image dataset count. The ConvNet VGG model is a good benchmark classification model. Figure 4 shows our proposed VGG classification model.

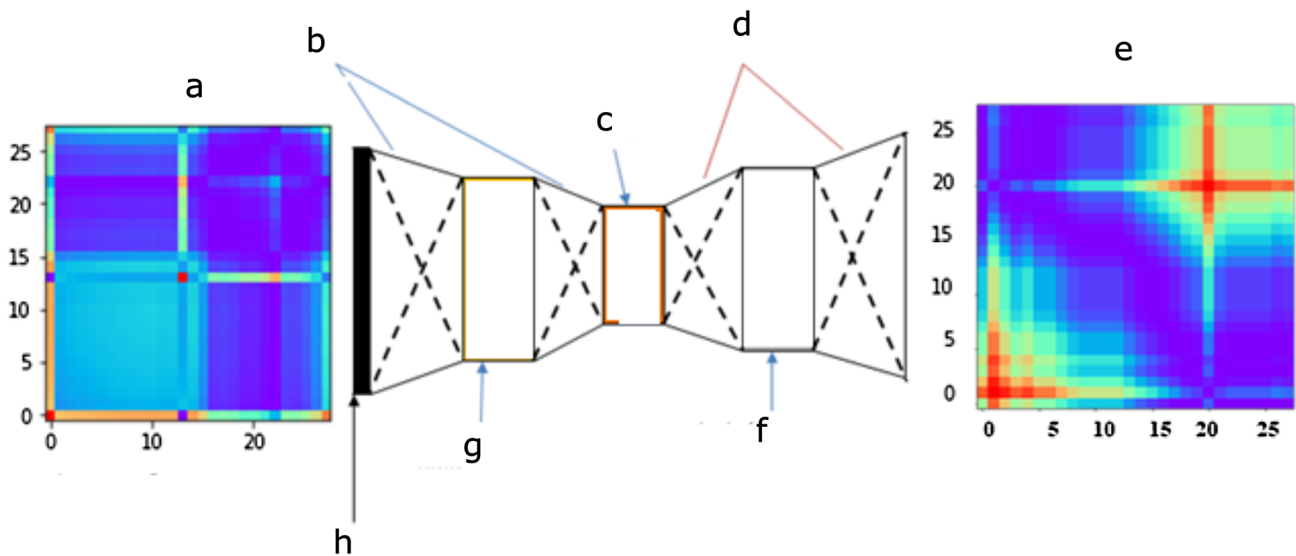
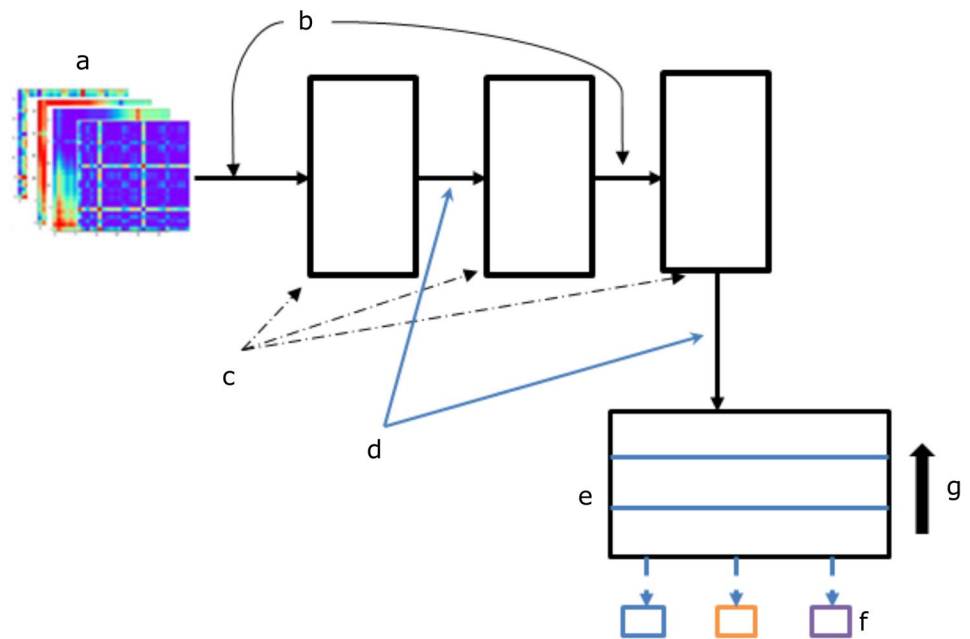


Fig. 3 Image denoising autoencoder for the NILM disaggregation. **a** Aggregate image, **b** encoder, **c** encoding hidden layer, **d** decoder, **e** target image, **f** decoder hidden layer, **g** encoder hidden layer, and **h** Gaussian added noise equivalent due to loads other than the target load

Fig. 4 Proposed CNN classification network. **a** Disaggregated input images, **b** convolution + ReLu, **c** feature maps, **d** max pooling, **e** fully connected network, **f** class outputs, and **g** backpropagation network path



Dataset creation

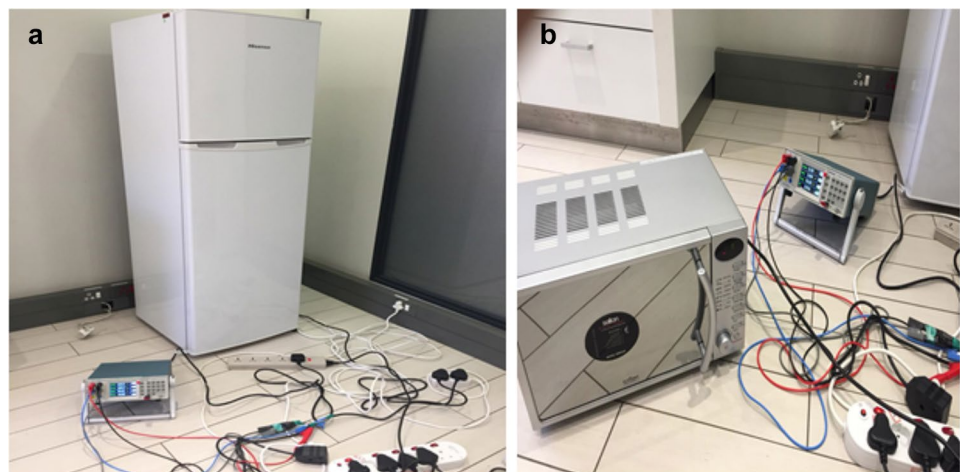
A number of public datasets exist for experimenting on developing NILM recognition systems. However, most of the load equipment in these datasets is either obsolete or advancement in technology has altered slightly their signature characteristics. This in itself is not a major issue for developing the models, but how to apply and validate these models becomes a problem. Also some datasets have varied data acquisition sampling time and are defined for activation periods of days and even months. This would generate enormous data which is beyond the scope of the CPU computations platform that we are using in this paper. To this end, we propose a simpler dataset, however, not necessarily less important for our experiment.

The data used in the experiments were obtained in a laboratory setup using the following appliances:

- Hisense refrigerator (RF),
- SMW20E Salton microwave oven (MW),
- Philips 5 W (60 W) LED lamp (L2),
- Radiant 12 W (100 mA) CFL lamp (L1) and
- Radiant 14 W (110 mA) CFL lamp (L3).

The data were acquired at a sampling rate of 1 Hz by using a Tektronix PA1000 Power Analyzer [34]. The programming environment was based on 64 bit python 3.6.3 64 bit software, keras 2.2.4, tensorflow 1.5.0 backend, numpy 1.17.0, pandas 0.20.3, pyts 0.8.0, scipy 1.3.1, and scikit-learn 0.20.1 packages, on an Intel® CPU 2.30 GHz 4.00 GB Ram 64bit HP ProBook 450 G3 laptop. Figure 5 shows the

Fig. 5 Photo of experiment setup for RF, MW, L1, L2 and L3 parameter measurement. **a** RF, and **b** MW



experiment setup for acquiring the data using the PA1000 Power Analyser.

The appliances are connected in parallel as per PA1000 instructions to an alternating-current (A.C) mains power source. Each appliance is connected to the mains power cable through switched mains power extension cables incorporating lamp holders in the case of lamps. For measurement reproducibility, each appliance and plug point is assigned a specific label. USB data logging, for datasets creation is at a frequency of 1 Hz. This sampling frequency determines whether we implement high transient, slow transient, event detection or non-event detection based, feature extraction algorithms. However, in the event of designing a data acquisition signal processing hardware we can make use of much higher sampling frequency since provision for buffer storage can be incorporated.

Power series signals

Figure 6 shows the current (I_{rms}) aggregate appliances signal for a laboratory experiment whose objective was to disaggregate and classify each appliance specified in this diagram using deep learning method.

The diagram gives the aggregate profile of a refrigerator (FR), a microwave oven (MW), and mains 12 W (L1), 5 W (L2) and 14 W (L3) lamps. The appliances under consideration have various activation periods. In the experiment, the refrigerator's activation period before the next appliance (microwave oven) is switched ON spans 1170 s. However, from about 28 s to 1170 s its response is fairly constant. The microwave oven is switched on to operate at idle for 120 s after the 1170 s of refrigerator operation. Figure 7 shows the point at which we switch on the microwave oven idle status into the circuit.

After 120 s the microwave oven is timed for a period of another 120 s to bring 100 ml of water to the boil and then switched off. The refrigerator continues to run and after

Fig. 6 Aggregate current (I_{rms}) profile showing the appliance activations in the laboratory experiment. The following are the load operational status: At point **a** FR “ON”, intervals **b** FR + MW on idle, **c** MW boils 100 ml of water, **d** FR + MW, point **e** MW “OFF”, intervals **f** FR + L1, **g** FR + L1 + L2, **h** FR + L1 + L2 + L3, **i** FR “OFF” + L1 + L2 + L3, **j** L2 + L3, **k** L2, and at **i**. All loads “OFF”

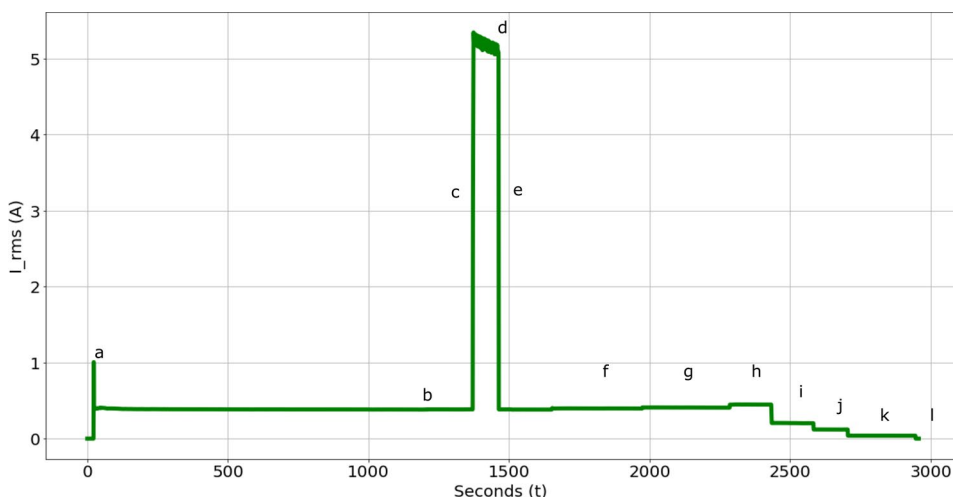
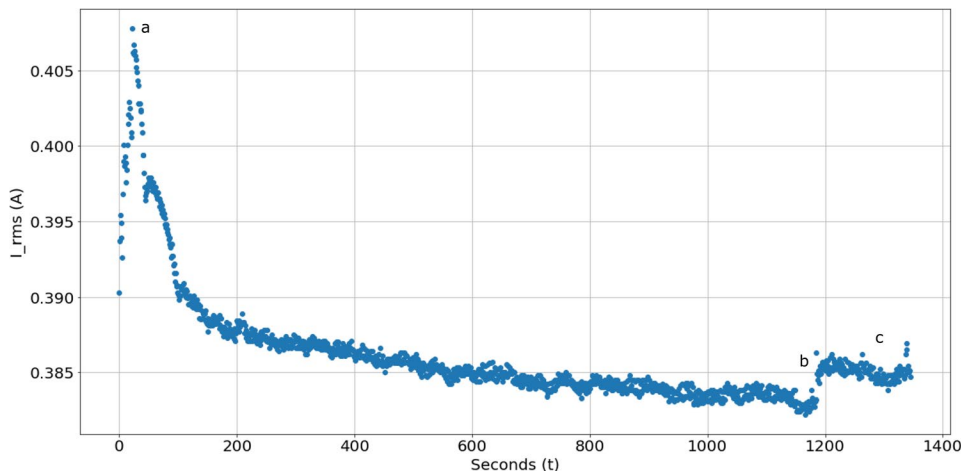


Fig. 7 Refrigerator and microwave oven activation points. **a** Refrigerator switch “ON”, **b** microwave oven switch “ON”, and **c** refrigerator and microwave oven “ON” but operating at idle



operating for 120 s L1 is switched on to operate with the refrigerator for a period of 360 s before L2 is added into the circuit and the combination operates for 300 s. We then add another lamp L3 into the circuit comprising RF, L1, and L2 to operate for an additional 300 s before the refrigerator cuts off automatically leaving L1, L2 and L3 switched on. The remaining L1, L2 and L3 combination operates for 120 s after which we switch off L1. Now we remain with L2 and L3 in the circuit for 240 s after which we disconnect L3. L2 operates for a short period before we finally remove it from the supply to remain with no connected appliances.

The data acquisition unit automatically samples additional load parameters that include voltage, frequency, active power (Watts), and power factor (PF) for both the aggregate and ground truth signals. The active power load profile was similar in shape to the current load profile. Figure 8 shows the PF aggregate profile for the experiment in this paper. PF which is the ratio of real power to apparent power ($PF = \frac{\text{watts}}{\text{voltage} \times \text{current}}$) normally measurements the energy efficiency of the appliance. As can be seen from the preceding expression the PF in appliance steady-state operation is a consequence of the active power, current and voltage. We performed rigorous NILM recognition experimentation with the images generated from all the three signal parameters. The PF energy efficiency characteristics in principle can be used to provide recognition of an appliance since PF varies as the active power which is directly related to the current. However, the I_{rms} and Watts signals give a direct representation of the operational features of the appliance and are therefore more appropriate parameters for the recognition. Hence, the I_{rms} and Watts parameters do, in fact, individually provide all the features required to provide unique signal identity. We get a boost on the signal recognition performance if we consider an increased dataset that includes current and power factor or active power and power factor.

The power series signals are taken as raw data spanning the entire aggregate signal sample length and target load activation windows. The pyts package in python facilitates the generation of the signal images from power series representation. We consider gramian angular fields (GAFs) to transform the power series into image equivalent form for input into our image-based NILM recognition system.

Gramian angular fields

The Gram Matrix (Gramian or metric) matrix [35] is the basis for encoding appliance power series signal to an image. The appliance signals are encoded to GAF using the procedure in [15, 36], and then rescaling the signals $X = \{x_1, x_2, \dots, x_n\}$ to fit in the ranges of -1 to 1 or 0 to 1 as given in Eqs. (6) and (7) respectively.

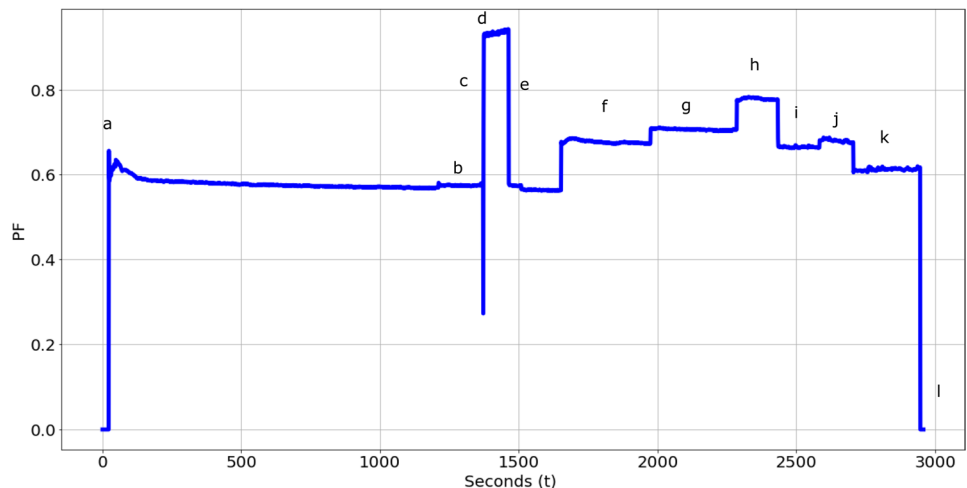
$$\tilde{X}_{-1}^i = \frac{(x_i - \max(X)) + (x_i - \min(X))}{\max(X) - \min(X)}, \tag{6}$$

$$\tilde{X}_0^i = \frac{x_i - \min(X)}{\max(X) - \min(X)}. \tag{7}$$

After rescaling, the time series is converted to polar coordinate as given in Eq. (8), where the value is the angular cosine and the time stamp (t_i) is the radius r θ is the polar coordinates angle and N is the regularization constant factor for the span of the polar coordinate system [15, 20]. On the polar plot advancing time scale concentric circles are accompanied by time scale values that warp through the various angular points. The angular limit for the scale $[0, 1]$ is $[0, \pi]$, and for $[-1, 1]$ is $[0, \frac{\pi}{2}]$ [15].

$$\begin{cases} \theta = \arccos(\tilde{x}_i), & -1 \leq \tilde{x}_i \leq 1, \tilde{x}_i \in \tilde{X} \\ r = \frac{t_i}{N}, & t_i \in N. \end{cases} \tag{8}$$

Fig. 8 Aggregate PF appliances profile for the laboratory experiment. Operating points and intervals are defined as in Fig. 1 since the time scale is the same



A Gramian Matrix [15] is realized from the polar coordinate vectors. Either, the image-based Gramian Angular Summation Field (GASF) or the Gramian Angular Difference Field (GADF) as defined in Eqs. (9)–(12) [15, 20] image form of the matrix is possible,

$$\text{GASF} = [\cos(\theta_i + \theta_j)], \quad (9)$$

$$= \tilde{X}' \cdot \tilde{X} - \sqrt{I - \tilde{X}^2}' \cdot \sqrt{I - \tilde{X}^2}, \quad (10)$$

$$\text{GADF} = [\cos(\theta_i - \theta_j)], \quad (11)$$

$$= \sqrt{I - \tilde{X}^2}' \cdot \tilde{X} - \tilde{X}' \cdot \sqrt{I - \tilde{X}^2}, \quad (12)$$

where I is the unit row vector $[1, 1, \dots, 1]$.

Equation (13) shows how the time series can be accurately reconstructed from the GASF main diagonal [20].

$$\cos(\theta) = \sqrt{\frac{\cos(2\theta) + 1}{2}} \quad \theta \in \left[0, \frac{\pi}{2}\right]. \quad (13)$$

In Fig. 9, we show a typical polar plot and the respective gramian angular field 28×28 images generated from the experiment PF composite signal.

The disaggregation algorithm is evaluated based on the training and validation/testing. To this end, we create a database composed of training and aggregate validation/testing images. We created the aggregate training dataset by adding synthetic power series data to the real validation data in the ratio 50:50. The validation data remains as only real data. Figure 10 shows aggregate training and validation images for the current parameter.

The outlook of poorly outlined transformed images can be improved by applying the logarithmic transform [37] to the image values using the expression in Eq. (14).

$$\tilde{I}_{\text{GASF}} = \log_{10}(1 + I_{\text{GASF}}), \quad (14)$$

where I_{GASF} is the GASF image after the time series polar plot transformation, and \tilde{I}_{GASF} is the log transformed image. A method of improving the image contrast is suggested in [37].

Markov transition fields

The Markov transition field (MTF) involves the encoding of time series into quartile bins. A Markov transition matrix is produced and the MTF result is given as in [15]. The MTF captures well, time-series dynamics as opposed to GAF that is good at static time series transformations. However, MTF has poor capability to reconstruct the time series from the image as opposed to GAF. This necessitates a future holistic

approach to our neural network image dataset creation to include both the GAF and MTF as done in [15].

Ground truth signals

The appliances various activation periods define the number of images that can be produced. The steady-state operation of the refrigerator is defined between 28 and 1170 s. Figure 11 shows the refrigerator switch ON characteristics and active microwave oven activation period.

The refrigerator dataset images are created from Fig. 11 and from various data lengths up to 1170 s. Likewise, the microwave oven dataset is created from the respective microwave oven power series. In this experiment, the refrigerator plus microwave oven on idle are ON for 120 s whilst in the active mode heating 100 ml of water the microwave oven is ON for 120 s. Hence, with an image set of 30 s (approximated from the 28 s above) long we can realize a minimum of four microwave ON images. These images are the training labels in the supervised learning case. Figure 12 shows typical refrigerator and microwave oven I_{rms} and PF GASF, and GADF ground truth images.

Performance metrics

We use the receiver operating characteristic (ROC) Curve to evaluate our classification performance. The area under curve (AUC) of the ROC characteristics gives the probability of correctly discriminating between two entities. The AUC in Fig. 13 is interpreted as [38, 39]

- 0.5–0.6 failed,
- 0.6–0.7 poor,
- 0.7–0.8 fair,
- 0.8–0.9 good, and
- 0.9–1 excellent discrimination.

Multiclass classification can also be achieved through the ROC curve [40].

We complement the ROC classification metrics with accuracy, precision, recall, F-measure [13], and confusion matrix. These metrics are defined as,

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (15)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (16)$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}, \quad (17)$$

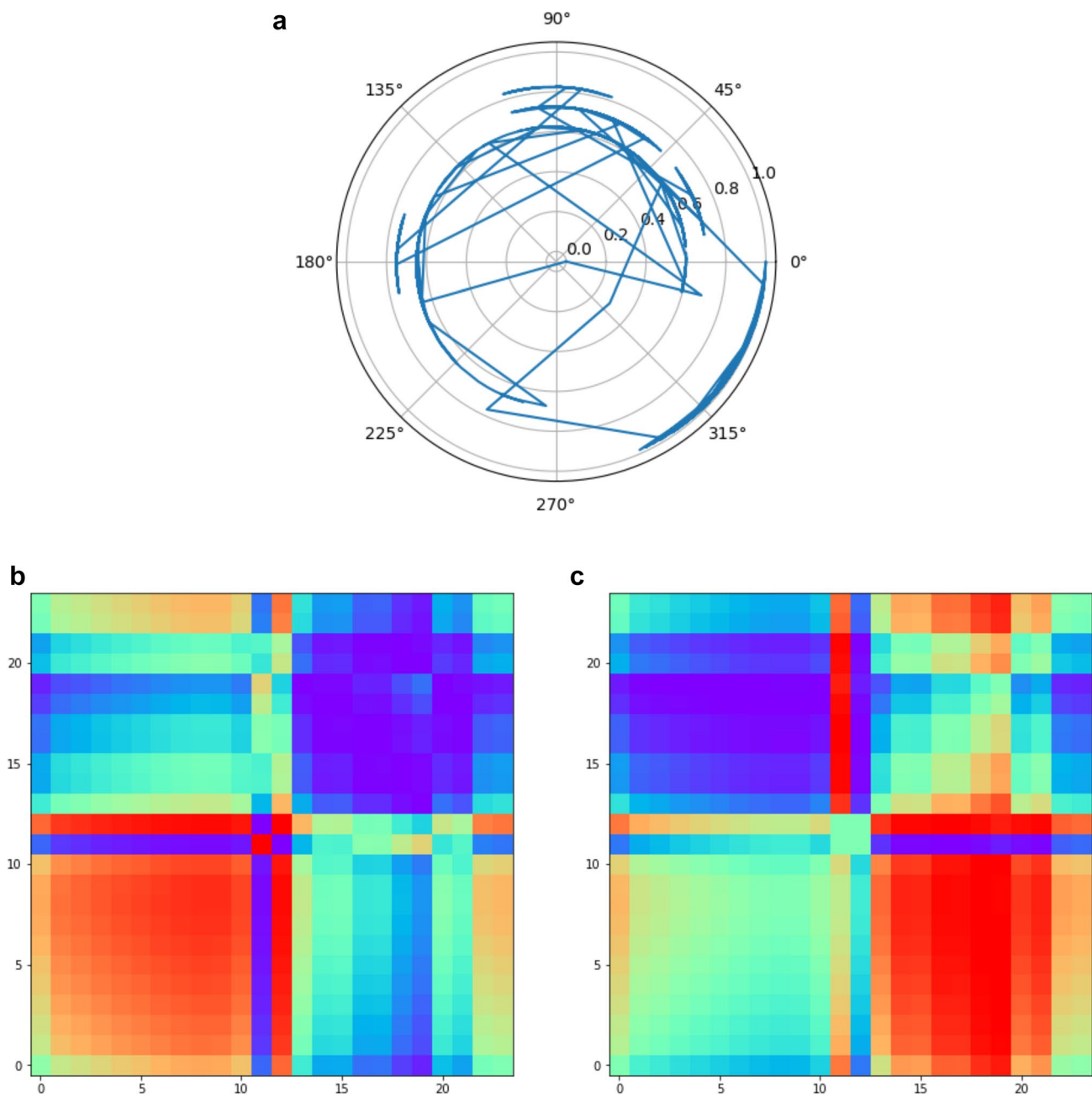


Fig. 9 PF aggregate signal transformed to image. We show **a** polar plot, **b** GASF, and **c** GADF generated images

$$F - \text{measure}(F_1) = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (18)$$

where TP is true positives, FP is false positives, FN is false negatives and TN is true negatives.

Accuracy defines the output popularity of an expected outcome in relation to the total possible outcomes in a sample. Say we have 480 TN outcomes in one class and 20 TP outcomes in another class. Then 480 outcomes over 500 total outcomes will give us an accuracy of 96%. This translates to TN = 480, FN = 20, TP = 0 and FP = 0. A classification

model trained on this unbalanced data may give a high accuracy in favor of the higher sample count hence accuracy on its own will not provide a good measure of the models' performance. The precision and recall determine how good the TP is acknowledged by the model. By looking at Eqs. (15) and (16) the preferred values of precision and recall are unity. Hence, precision and recall are preferred classification metrics so as to obtain the classification outcomes we want. This takes us to Eq. (18) the F-measure. The F_1 which contains the values of precision and recall give a better representation of the performance of the model in terms of

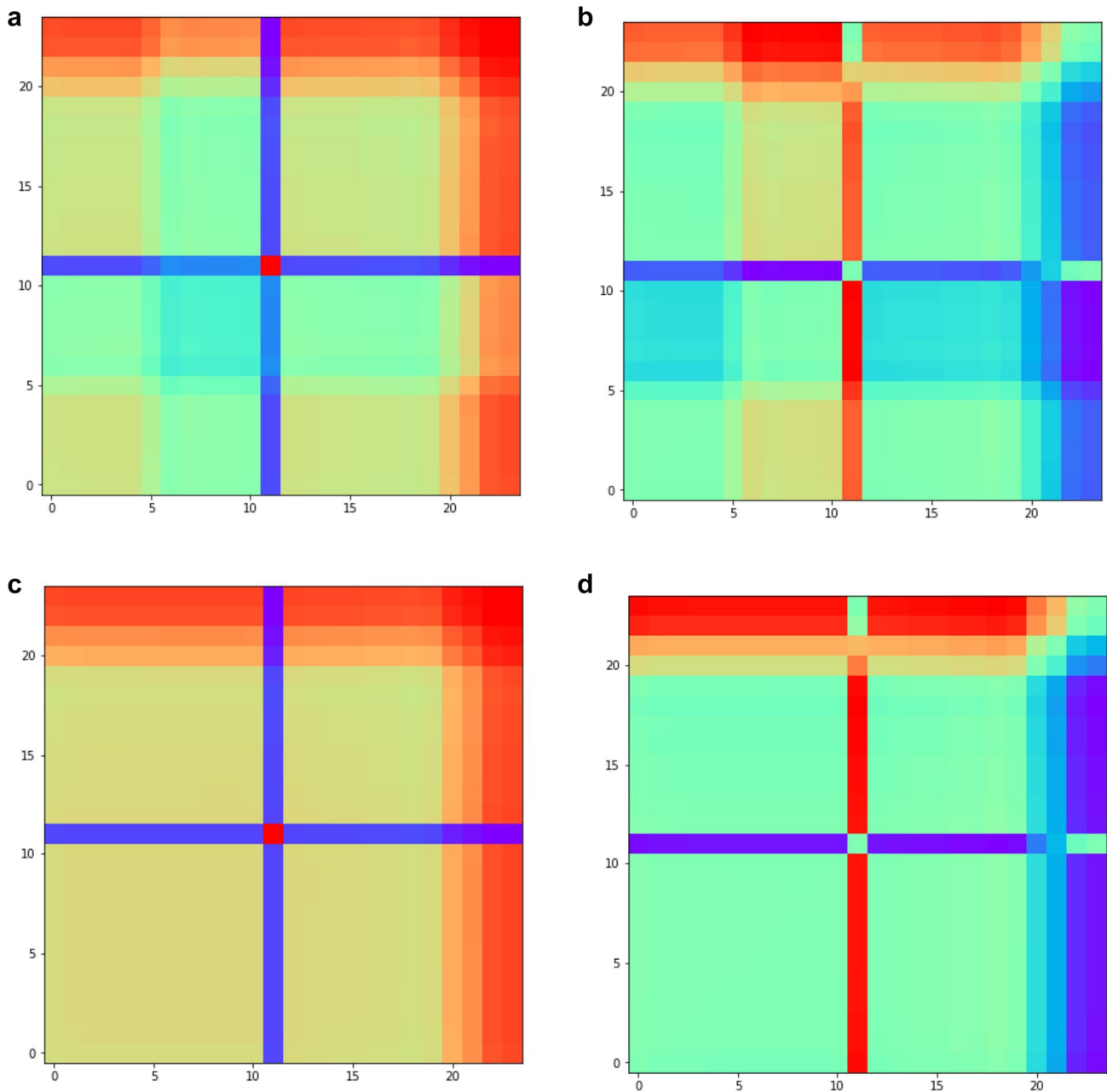


Fig. 10 Disaggregation training and validation aggregate I_{rms} based images. **a** Training GASF, **b** Training GADF, **c** validation GASF, and **d** validation GADF images

providing the right classification. The preferred value of F_1 score is unity. The confusion matrix gives a summary result of the expected against the predicted outcomes.

To evaluate the disaggregation performance we use the binary cross entropy (BCE) loss [41]. The cross-entropy loss (CE) is given as

$$CE = - \sum_i^w z_i \log (q_i), \quad (19)$$

where w is number of classes, q is class i predicted probability and z_i class i true probability. The CE gives the interpretation of the log-likelihood for z_i given a function q_i . The BCE is then given as in Eq. (20), for two classes.

$$BCE = - \sum_{i=1}^{w=2} z_i \log (q_i) = -z_1 \log (q_1) - (1 - z_1) \log (1 - q_1). \quad (20)$$

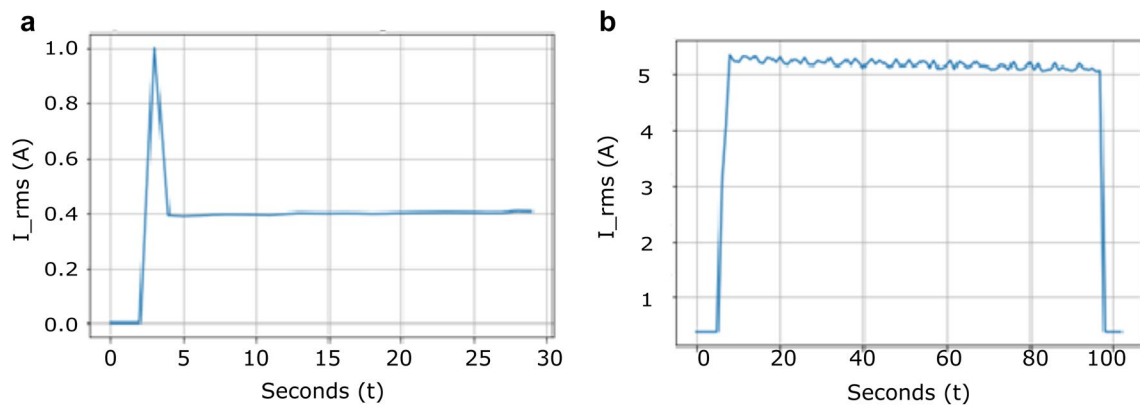


Fig. 11 Close up refrigerator and microwave oven operating characteristics: **a** refrigerator at switching “ON” and **b** microwave oven high power activation period

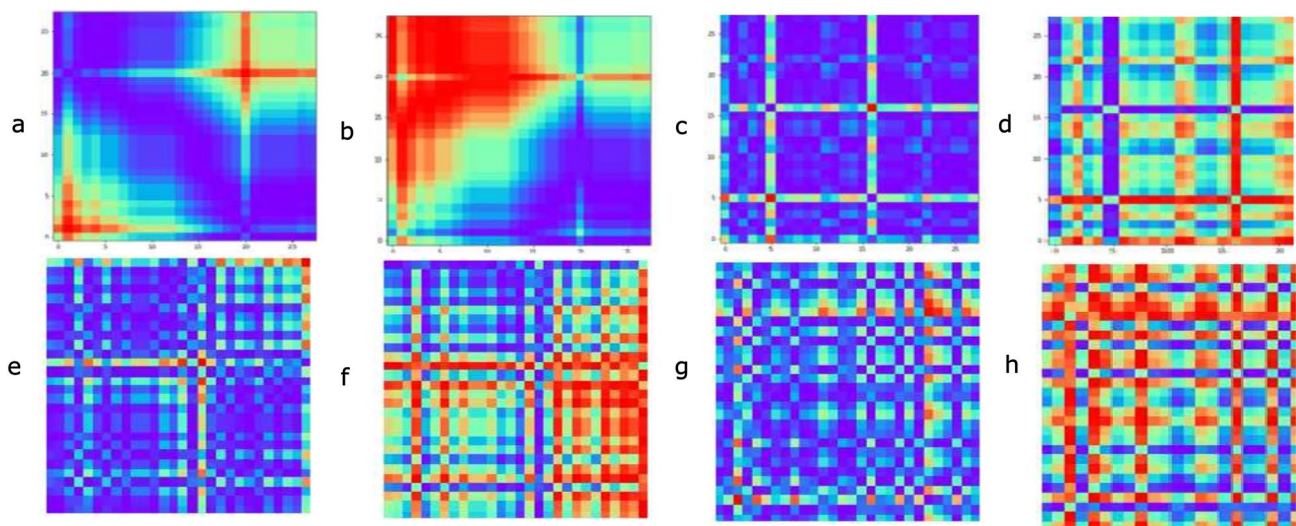


Fig. 12 Refrigerator and microwave oven ground truth signal images. **a** Refrigerator PF GASF, **b** refrigerator PF GADF, **c** microwave oven PF GASF, **d** microwave oven PF GADF, **e** Refrigerator I_rms GASF, **f** refrigerator I_rms GADF, **g** microwave oven I_rms GASF, and **d** microwave oven I_rms GADF

The kappa index which represents the level of agreement between two raters is defined in the range $[-1, 1]$. A value of -1 is no agreement at all, 0 is a chance agreement and 1 is perfect agreement. The code based on the confusion matrix in [42] was taken as the basis for formulating the kappa index calculations which have been included in our Results.

Description of proposed method

Pseudo-code for proposed method

Pseudo-code for proposed method In the proposed method we verify the performance of an image-based NILM

disaggregation and classification scheme of five appliances from the aggregate mains supply. We use an image-based denoising autoencoder for the disaggregation and a ConvNet VGG architecture for the classification of the denoised appliance signature images. In our method, we identify five image-based disaggregation networks and one image based multi-class classification network. This basically translates to two main algorithms in our method given by Pseudocode 1, and Pseudocode 2 for the disaggregation and classification respectively. We also compare the classification performance of our system to a power series based on using the same ConvNet VGG model.

Pseudo code 1: Training and disaggregation process of the *sDAE*

- Acquire the training aggregate signal image(s) (*trag*)
 - Acquire the training appliance ground truth images (*trgr*), and set number of appliance images (*napsimg*) = 5
 - Acquire validation aggregate signal image(s) (*vag*)
 - For the encoder structure define the no. of 2D CNN hidden layers to include, no. and size of filters, activation function, and maxpooling. For the decoder structure define the no. of 2D CNN hidden layers to include, no. and size of filters, activation function, and upsampling
 - Define no. of epochs (*epcs*), set call-backs criteria and set no. of independent *sDAE* networks (*nosDAE*) = 5. (*napsimg* = *nosDAE*)
 - Initialize all weights
-

While *nosDAE* ≤ 5 **do**

1. Start training (*epcs* = 0)
 - i. Input *trag* into *sDAE*
 - ii. Input *trgr* into *sDAE*
 - iii. Iterate through *sDAE*
 - iv. *sDAE* converged → stop
 - v. *sDAE* not converged return [iii]
2. Train next *sDAE* network
 - i. *nosDAE* = *nosDAE* + 1
 - ii. *sDAE* = *sDAE* + 1
 - iii. Return [1 i.]
3. Stop disaggregation training
 - i. *nosDAE* > 5
 - ii. Exit

Otherwise: **do**

4. Start validation (Testing)
 - i. Input *vag* into *sDAE*
 - ii. Output disaggregated image (*dsimg*)
 - iii. Convert *dsimg* to power series
 - iv. Compare with *trgr* power series
 - v. *sDAE* = *sDAE* + 1 → return [i]
5. Stop disaggregation validation/testing Train and disaggregate all parameters:
 - i. *sDAE* > 5
 - ii. Exit

Pseudo code 2: Training and classification process of the *ConvNet VGG* network

- Acquire a ground truth training and validation dataset produced from five appliances comprising three mains lamps, a refrigerator and a microwave oven (*grttv*)
 - Acquire a ground truth testing dataset produced from five appliances comprising three mains lamps, a refrigerator and a microwave oven (*grt*)
 - Define no. of epochs (*epcs*), 2D CNN hidden layers, filters and size, dense layers (*dsl*), neurons per dsl, all according to ConvNet VGG architecture, activation functions, call backs, optimizers, loss function and metrics
 - Acquire the five denoised images from *sdAE_1*, *sdAE_2*, *sdAE_3*, *sdAE_4* and *sdAE_5* disaggregation networks
 - Initialize all weights
-

1. Start training (*epcs* = 0)
 - i. Input *grttv* into ConvNet
 - ii. Iterate through *ConvNet VGG*
 - iii. *ConvNet VGG* converged → Stop
2. *ConvNet VGG* not converged
 - i. *epcs* = *epcs* + 1
 - ii. return [1. ii.]
3. Start Testing
 - i. Input *grt* into *ConvNet VGG*
 - ii. Test okay → Stop
4. Test not okay
 - i. Redesign network based on *ConvNet VGG*
 - ii. Return [1i]

If test okay: *do*

5. Classification of disaggregated images
 - i. Input into *sdAE_1*, *sdAE_2*, *sdAE_3*, *sdAE_4* and *sdAE_5* into *ConvNet VGG*
 - ii. Output class value
 - iii. Exit

Else: *do*

6. Return [3]

Model architectures

Encoding The encoder model is made-up of three 2-D CNN layers. The first CNN layer which accepts the aggregate image input of shape of $28 \times 28 \times 3$ has 64 filters each of dimensions of 3×3 . The output of this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 2-D max pooling operator of

dimensions of 2×2 . The second CNN layer which accepts the 2-D max pooling output of the first CNN layer has 32 filters each of dimensions of 3×3 . The output of this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 2-D max pooling operator of dimensions of 2×2 . The third CNN layer which accepts the 2-D max pooling output of the second CNN layer has 16 filters each dimensions of 3×3 . The output of

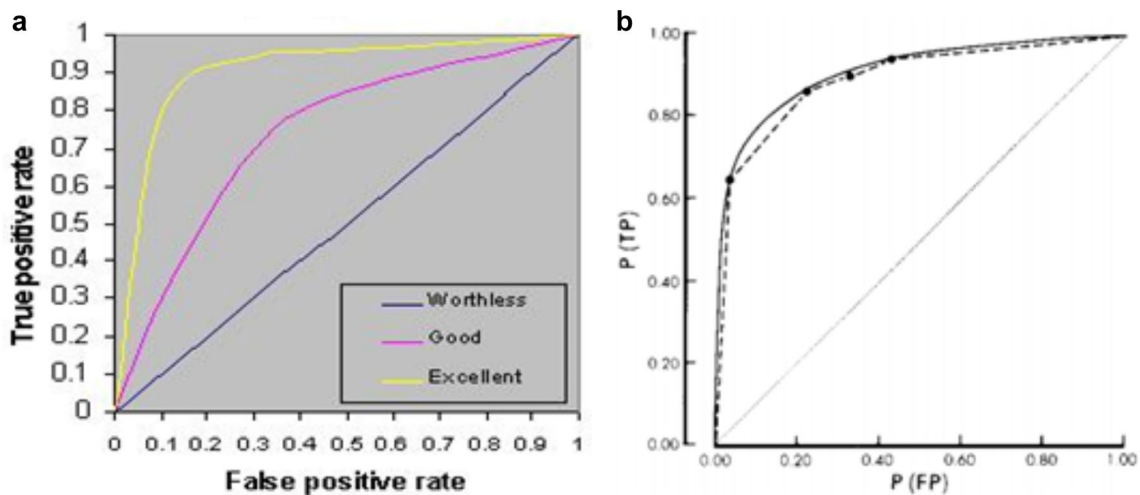


Fig. 13 Information from two ROC curve representations with true positive rate (P (TP)) and false positive rate (P(FP)) axis. **a** Comparing ROC Curves [39], and **b** Diagonal line, empirical and Gaussian (solid line) curves on the ROC [38]

this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 2-D max pooling operator of dimensions of 2×2 to give the encoded output.

Decoding The decoder model is made-up of three 2-D CNN layers. The first CNN layer which accepts the encoded input from the encoder has 16 filters each of dimensions of 3×3 . The output of this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 2-D up sampling operator of dimensions of 2×2 . The second CNN layer which accepts the 2-D up sampling output of the first CNN layer has 32 filters each of dimensions of 3×3 . The output of this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 2-D up sampling operator of dimensions of 2×2 . The third CNN layer which accepts the 2-D up sampling output of the second CNN layer has 64 filters each of dimensions of 3×3 . The output of this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 2-D up sampling operator of dimensions of 2×2 . The output is a CNN layer which accepts the 2-D up sampling output of the third CNN layer and has three filters each of dimensions of 3×3 acted upon by a sigmoid activation function.

The encoding and decoding model used the adam optimizer and binary_crossentropy loss function with early stopping based on the minimum validation loss.

ConvNet classification The ConvNet classification model is made-up of three 2-D CNN layers, followed by a flatten operation, and finally by two fully connected layers. The first CNN layer which accepts the disaggregated appliance power series inputs of shape of $28 \times 28 \times 3$ has 8 filters each of

dimensions of 3×3 . The output of this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 2-D max pooling operator of dimensions of 2×2 . The second CNN layer which accepts the 2-D max pooling output of the first CNN layer has 16 filters each of dimensions of 3×3 . The output of this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 2-D max pooling operator of dimensions of 2×2 . The third CNN layer which accepts the 2-D max pooling output of the second CNN layer has 64 filters each dimensions of 3×3 . The output of this layer is acted upon by a ReLU activation function. Then a Flatten layer operates on the 2-D max pooling output of the third CNN layer. The flattened output is input into a fully connected dense layer with 16 neurons, followed by a ReLU activation function and a dropout factor of 0.25. The output of the first fully connected dense layer is then channeled into the input of another fully connected layer with N output neurons (N is equal to 2 for binary classification, and N is equal to 4 for classification of four appliances). The output of this layer is then operated on by a softmax activation function for class output. We can use the sigmoid activation function which outputs probability values instead of the class values for the classification. The classification model used the RMSprop optimizer with a learning rate of 0.001.

(The Adam optimizer can be used with the sigmoid function).

Power series classification model for comparison The 1-D CNN classification model is made-up of five 1-D CNN layers, followed by a single output dense layer. The first CNN layer which accepts the disaggregated appliance power series inputs of shape of (4, 1) has eight filters each of dimension of 1. The output of this layer is acted upon by

a ReLU activation function. The second CNN layer which accepts the output of the first CNN layer has 16 filters each of dimension 1. The output of this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 1-D max pooling operator of dimension 1. The third CNN layer which accepts the output of the second CNN layer has 16 filters each of dimension 1. The output of this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 1-D max pooling operator of dimension 1, followed by a dropout factor of 0.5. The fourth CNN layer which accepts the output of the third CNN layer has 16 filters each of dimension 1. The output of this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 1-D max pooling operator of dimension 1, followed by a dropout factor of 0.5. The fifth CNN layer which accepts the output of the fourth CNN layer has 16 filters each of dimension 1. The output of this layer is acted upon by a ReLU activation function. The ReLU plus CNN output is then operated on by a 1-D GlobalAverage-Pooling layer, followed by dropout factor of 0.25. The final output layer which follows is a dense layer with 1 neuron and a sigmoid activation. The power series model used RMSprop optimizer with the hyperparameter settings: $lr = 0.001$, $\rho = 0.9$, $\epsilon = \text{none}$, and $\text{decay} = 0.0$.

A comparison between the encoding, decoding, ConvNet classification and power series models is shown in Table 1.

Training framework and procedure

With reference to the architecture of the CNN we added experiments where we changed the learning rate, type of optimizer, the number of epochs, the number of layer neurons and the batch size. In the sdDAE, we started with three CNN layers in the encoder having the numbers of filters (neurons) of 1024, 512, and 256, respectively. In the decoder, we had three CNN layers with the numbers of filters (neurons) of 256, 512, and 1024, respectively. Increasing the number of CNN layers above three layers for the encoder and decoder, respectively, did not provide any noticeable improvement in the performance of the model. However, a decrease below three layers for the encoder and decoder, respectively, did adversely affect the performance of the model. We trained the sdAE so that the binary cross-entropy (BCE) loss function was minimized between the disaggregated output image and the ground truth appliance image. The BCE loss parameter calculates the error to be used in the weights and bias updates. To address overfitting we used the adam algorithm with early stopping and a learning rate that varied from $1e-5$ to 0.01. We initially started with thirty epochs but the model had no noticeable convergence. We gradually increased the epoch count to 350 epochs and we incorporated early stopping facility based on minimum

validation loss and a patience of 10. The model achieved acceptable disaggregation results for learning rates of 0.01 and 0.001 with training batch size equal to one and early stopping at 200 epochs. We experimented with the Adadelta, Adamax, Adam and Adagad optimizers, but the Adam optimizer provided better convergence results. When we gradually increased the number of CNN layer neurons above the ones that we initially had specified, we obtained overfitting with increased program running time. Consequently, we gradually reduced the number of CNN layer neurons until we obtained best results when the encoder CNN layers had the numbers of filters (neurons) of 64, 32, and 16 in each layer, respectively. In the decoder best disaggregation results were obtained when the CNN layers had the numbers of filters (neurons) of 16, 32, and 64 in each layer, respectively. With this new CNN layer count, the microwave disaggregation achieved best results at a learning rate of 0.01. However, the refrigerator and lamps disaggregation achieved best results at a learning rate of 0.001. For a batch size of 1 and 120 epochs the program running time was also considerably reduced. Increasing the batch size reduced the performance of the model. Also to increase the learning (faster execution time) and lower the usage of system memory we set our initial batch size to 1 (online learning). In this case, the network weights are updated after each training instance. In all our experimentation to account for non-linearity we introduced the ‘relu’ non-linear function into the convolution process.

Our classification model is decided on by the set structure of the ConvNet VGG high rate image classifier with defined 3×3 filter dimensions. However, we had to limit the number of layers to three and had to reduce the number of CNN neurons to 16, 32, and 64 for each of the layers, respectively. Too high a number of CNN neuron layers resulted in overfitting and too low CNN neuron layers resulted in underfitting. The dropout was varied from 0.25 to 0.5. The output activation function was set to the multi-class ‘softmax’ function. We experimented with various optimizers that included the RMSprop, adam and stochastic gradient descent (SGD) for learning rates that varied from 0.001 to 0.00001. To realize the 1D CNN power series model we experimented with various filter numbers in the range 4 to 128 and found best results for five CNN layers with the number of filters (neurons) of 8, 16, 16, 16, and 16 per layer from the input respectively. We experimented with various dropouts from 0.25 to 0.5. The third and fourth CNN layers from the input were each followed by a regularization dropout factor of 0.5, and the fifth CNN layer by a dropout factor of 0.25. The output activation function of the 1D CNN model was set to the ‘sigmoid’ which can be used for both regression and classification analysis. The power series model used RMSprop optimizer with the hyperparameter settings: $lr = 0.001$, $\rho = 0.9$, $\epsilon = \text{none}$, and $\text{decay} = 0.0$. The image datasets for all the models were produced as 400×400 images that were

Table 1 Comparison between the encoding, decoding, ConvNet classification and power series models

Model	Function	Learning task	CNN layers	Dropout	Optimizer (learning rate)	Per layer CNN number of filters (filter size)	Out activation function (loss function)	No. of dense layers (neurons)	Pooling/upsampling
Encoder	Compression of data (latent space)	Unsupervised/supervised	3	–	Adam (0.001–0.01)	64,32,16 (3×3)	Relu (BCE)	–	Max pooling (2×2)
Decoder	Mapping latent space to output	Unsupervised/supervised	3	–	Adam (0.001–0.01)	16,32,64 (3×3)	Relu (BCE)	–	Upsampling (2×2)
ConvNet	Classification	Supervised	3	0.25	RMSprop (0.001)	8,16,32 (3×3)	Softmax (BCE)	Flatten – 1 (16) – 1 (2)	Max pooling (2×2)
Power series	Classification	Supervised	5	0.5 0.5 0.25	RMSprop (0.001)	8,16,16,16,16 (3×3)	Sigmoid (BCE)	– 1 (1)	Max pooling (1) Global-Average- Poolin1D

then reshaped to 28×28 normalised images before input into the CNN.

Discussion of experimental results

The models are first trained using the I_rms image Dataset A and then trained using the PF Dataset B. Both datasets are split into train and validation data in the ratio 3:1. However, the test data varies from as little as one image to a total of five images.

Dataset A

The disaggregation was simulated using the autoencoder image to image regression code idea in [43]. In this section, we present the results of the disaggregation based on Fig. 3 that uses the denoising autoencoder. We present the disaggregated microwave, refrigerator and L2 lamp target load signals. We are also able to extend to the disaggregation of the other appliances namely L1 and L3, when their respective power series signals are transformed to image. In Fig. 14, we show the target refrigerator ground truth and disaggregated images and their respective power series representation. During training, a learning rate of 0.01 produced a blank predicted image, but the result was satisfactory for a learning rate of 0.001.

The results in Fig. 14 show that we are able to successfully disaggregate (predict) the refrigerator I_rms from the complex aggregate mains signal. The image features rather than the color define the signal. In Fig. 2 we have shown the convolution and feature extraction where the colour is

represented by varying shades of white to grey to black. So as far as the classification is concerned this predicted image is classified as a refrigerator. Figure 15 gives the BCE train loss characteristics for the refrigerator I_rms disaggregation model.

Dataset B

In Fig. 16 we show the target microwave oven image, the aggregate image for all the appliance activations and the predicted (disaggregated) microwave oven image for an Adam learning rate of 0.001.

The results in Fig. 16 show that we are able to successfully disaggregate (predict) the microwave oven from the complex aggregate mains signal. Unlike in the I_rms dataset, in this dataset there is an improvement in the disaggregation output as the learning rate approaches 0.01 as shown in Fig. 17. Figure 17 shows that the disaggregated image is identical to the target image for that load equipment.

It was observed that as the learning rate is decreased to $1e-5$ the disaggregation performance also significantly decreased until there was no recognition at all. In Fig. 18, we obtain a further decrease in the binary cross-entropy loss function as the learning rate is increased from 0.001 to 0.01. An increase in learning rate means that the loss function decreases faster to reach minima. However, due to the erratic behaviour of parameter updates local minima might not be achieved. Very low learning rates cause the loss function to stagnate, whilst very high learning rates can cause divergence (increase) in the loss function.

We then evaluated the autoencoder model on disaggregating the second load which is the refrigerator. Figure 19

shows that our developed model is able to disaggregate the second appliance from the same aggregate image as the previous load appliance with very high accuracy. The cross-entropy plot in Fig. 20 consolidates the high disaggregation capability of the network on the second load appliance.

In the third case, we evaluated the autoencoder model on disaggregating the LED mains lamp (L2) load. Once again Fig. 21 shows that our developed model is able to disaggregate the third appliance from the same aggregate image as the previous load appliances with high accuracy. The diagram in Fig. 21 shows switching bars around the image. However, there is a slight loss in detail at the upper right-hand corner of the predicted image. Nonetheless, the predicted image is a true representation of the target image as can be attested to the stable cross-entropy plot in Fig. 22.

Dataset B recognition performance

The initial model development entry point is based on 30 training images, 8 validation images and 8 test images belonging to the two classes of refrigerator and microwave oven. Based on only one input channel of PF, the model achieved a 100% model evaluation capability and was able

to accurately classify the eight test images that had not been seen before where class (0) is fridge and class (1) is microwave oven. The ROC plot is shown Fig. 23.

The corresponding confusion matrix for the ROC plot above is shown in Fig. 24. The confusion matrix shows that all the eight test samples are accurately classified. The precision, recall and F_1 score values are all equal to unity implying a perfect classifier.

We compare the proposed image-based model to a one-dimensional power series convolutional neural network (Conv1D) model based on 144 training samples (with a validation split of 0.2) and 40 test samples belonging to the two classes of the refrigerator and microwave oven. Based on only one input channel of PF, the model achieved a 100% model evaluation capability and was able to accurately classify the forty samples that were seen before where class (0) is fridge and class (1) is microwave oven. The ROC plot is shown in Fig. 25. Again here the precision, recall and F_1 score values are all equal to unity showing that this is also a good classification model.

The respective confusion matrix related to the ROC plot information in Fig. 25 is shown below in Fig. 26.

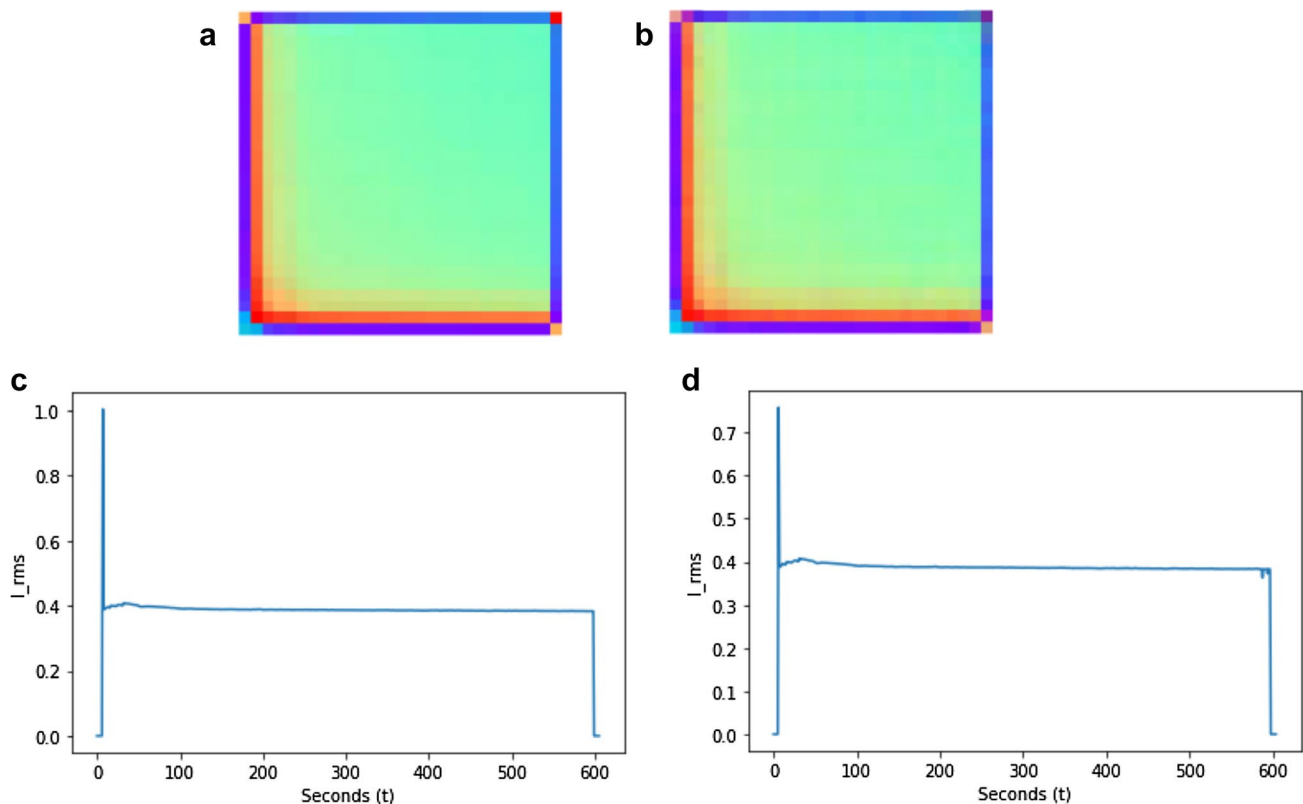


Fig. 14 Disaggregation of the refrigerator I_{rms} parameter. **a** Ground truth signal image, **b** disaggregated image, **c** ground truth signal, and **d** disaggregated signal

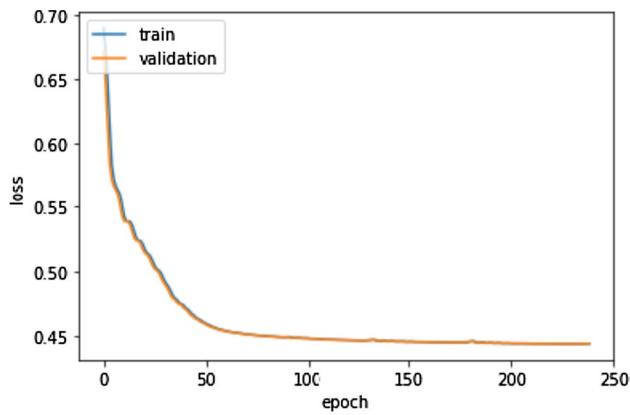


Fig. 15 BCE loss plot for the I_{rms} disaggregation

The situation changes when we test unseen data during the training for the power series signal. We obtain the performance ROC plot in Fig. 27 and the confusion matrix in Fig. 28, with the resulting precision, recall and F_1 score values are all equal to eighty per cent (0.8) which is an average classification result. The results show that our image proposed model achieves higher performance than the univariate power series which achieves eighty per cent recognition for unseen test data. By implementing a multivariate based power recognition whether by fusion techniques or otherwise we can investigate to see if the performance of the power series method can improve. If it does we would have used more data points than in our proposed method. Power series redundancies can contribute to the lower performance of the recognition network. Redundancies are not

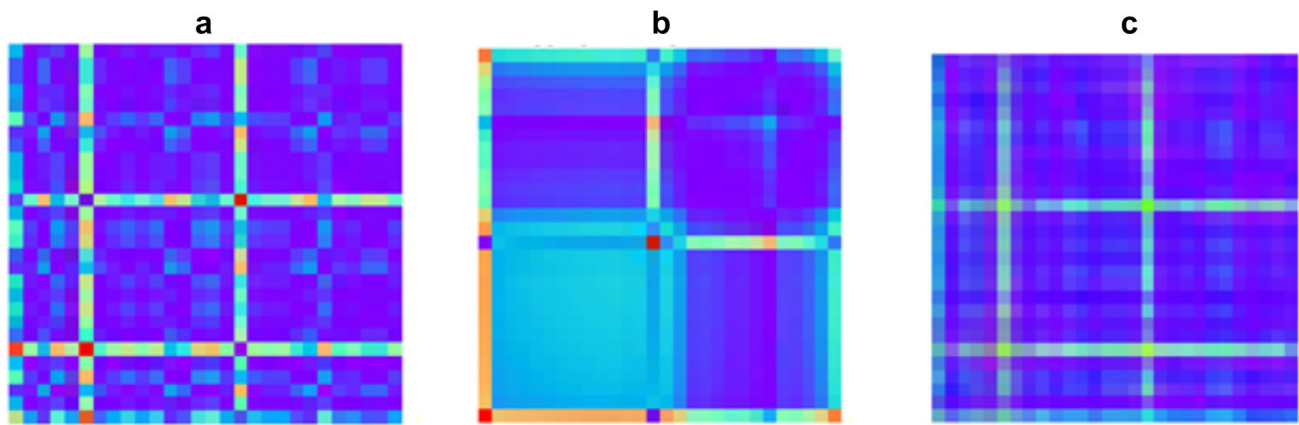


Fig. 16 Disaggregation of MW load signal from aggregate for an adam learning rate of 0.001. **a** Target MW load image without noise, **b** aggregate image with noise, and **c** disaggregated microwave oven signal image

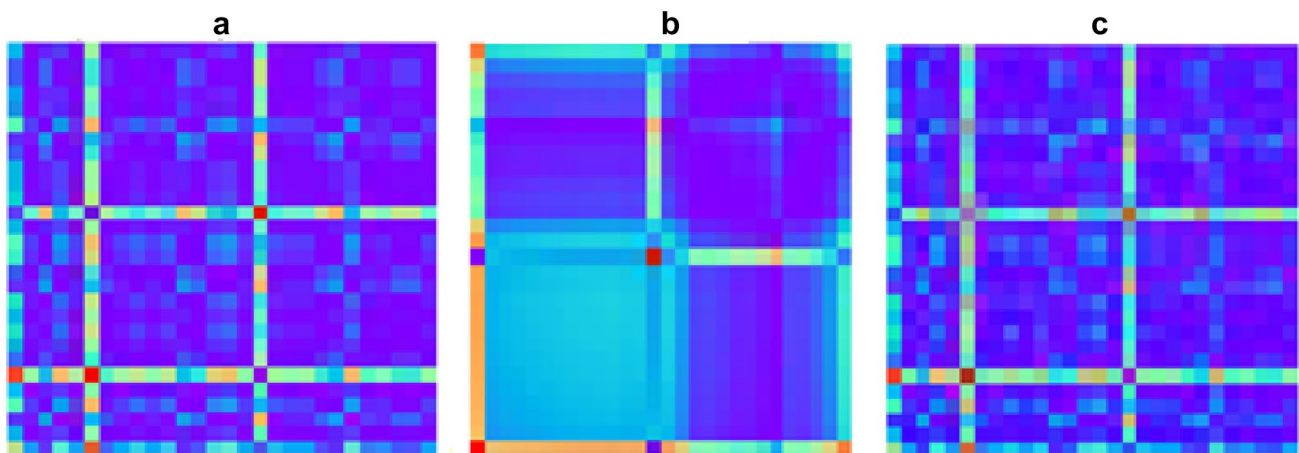


Fig. 17 Disaggregation of MW load signal from aggregate for an adam learning rate of 0.01. **a** Target MW load image without noise, **b** aggregate image with noise, and **c** disaggregated MW signal image

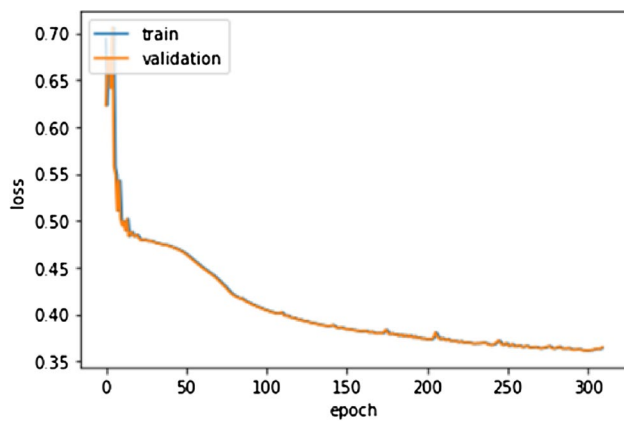


Fig. 18 MW Binary cross entropy loss for adam learning rate of 0.01

a major factor in the image recognition system as the generated redundancies images overlap into one during the power series to image transformation.

Comparative evaluation of our model based on the parameter type

A NILM recognition system can be developed bordering on various approaches of inputting data into the model. Primarily our data is acquired as a univariate power series that is individually acquired. We can then feed the data into the neural network as a stream of one power series or as parallel data in what is commonly termed the multivariate approach. The parallel data can also be fused to produce one composite data stream into the neural network. The multivariate or fusion approach has the advantage of availing more recognition features at the expense of more data handling and more data storage required capacity. The univariate approach is simpler, has less memory requirements but has

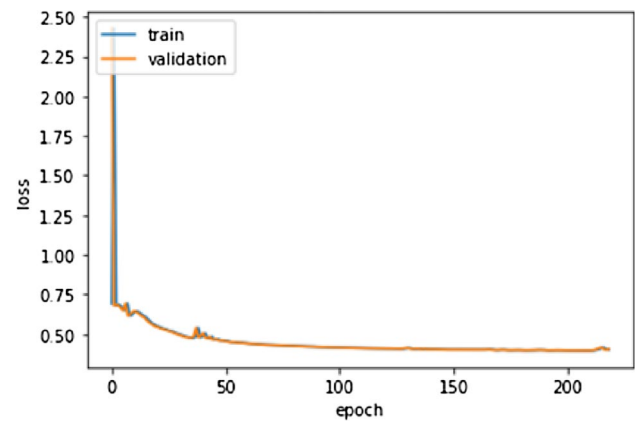


Fig. 20 RF Binary cross entropy loss for adam learning rate of 0.01

the disadvantage of availing less recognition features for the deep learning algorithm. Nonetheless in this paper for want of memory conservation and less data handling we used the univariate approach. Hence, we generate the required images from a single power series at a time and use this image in the designed recognition system. It is necessary to assess the developed models response to each signal image parameter. To this end, we evaluate the performance of the recognition model on the different parameters. The performance of a particular model on specific data can be improved by considering such aspects as transfer and ensemble learning. However, now we will not consider these approaches. Although we achieved excellent signature disaggregation the binary classification model results as can be seen in Table 2 show a need to improve the classification model design as explained in the last part of this results section.

The results in Table 2 show that all three parameters can successfully be used in the image-based NILM recognition system developed here. The recognition based on the

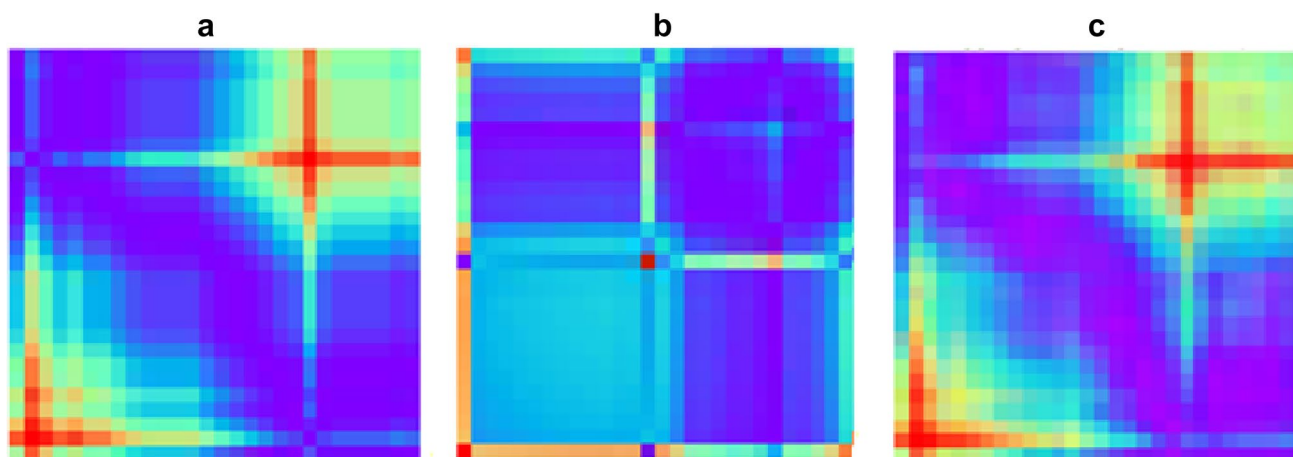


Fig. 19 Disaggregation of RF load signal from aggregate for an adam learning rate of 0.01. **a** Target RF load image without noise, **b** aggregate image with noise, and **c** disaggregated RF signal image

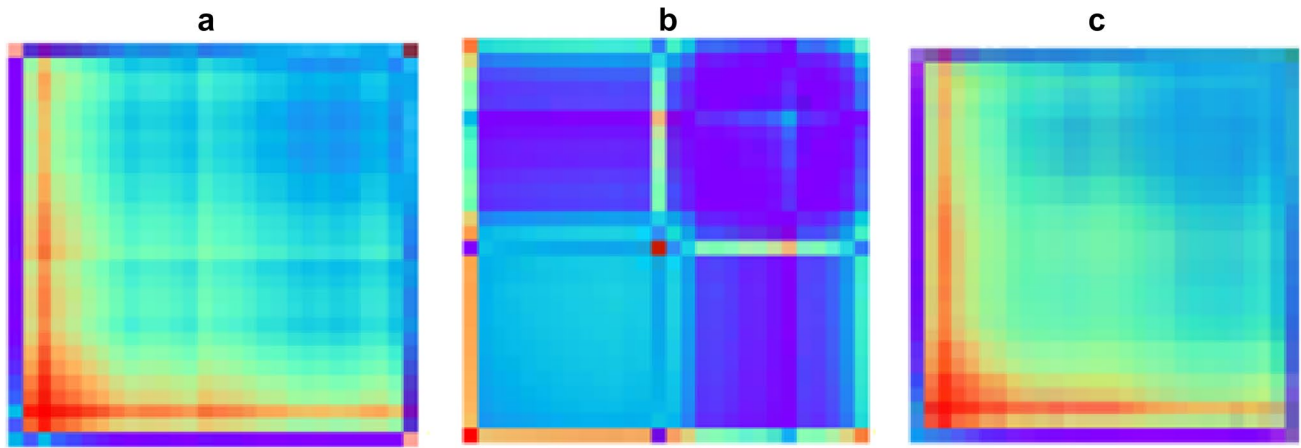


Fig. 21 Disaggregation of L2 load signal from aggregate for an adam learning rate of 0.01. **a** Target L2 load image without noise, **b** aggregate image with noise, and **c** disaggregated L2 signal image

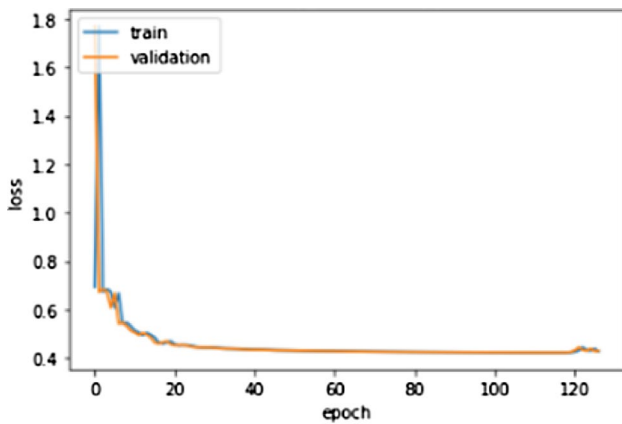


Fig. 22 L2 Binary cross entropy loss for adam learning rate of 0.01

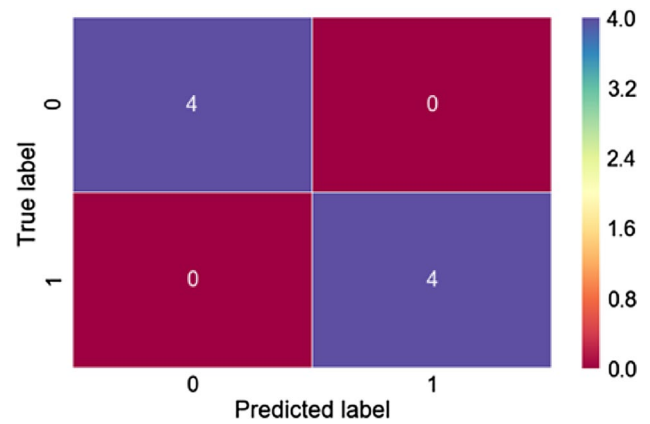


Fig. 24 Confusion matrix for PF RF and MW appliance classification

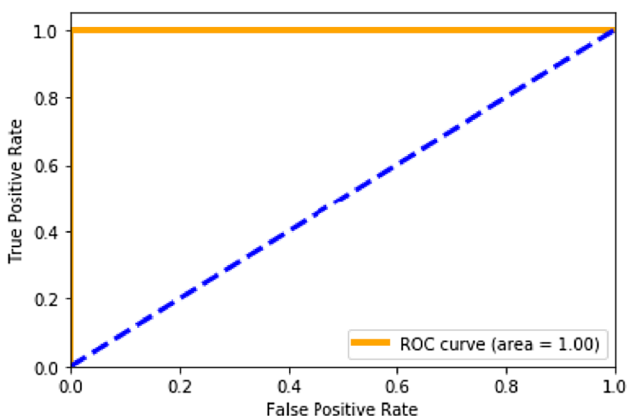


Fig. 23 ROC curve for PF RF and MW appliance classification

power signal parameter although somewhat less than that for PF produces acceptable average performance. In general, the current and power-based parameters provide a

more interpretable outcome, since it is easier to tell that the magnitude is high or low. On the other hand, PF is a more abstract energy efficiency measure parameter. The recognition model was trained with an RMSprop optimizer having a learning rate of 0.00001 arrived at through experimentation. The current accuracy and loss plots were noisy as shown in Fig. 29. In Fig. 30 we show the confusion matrix and the ROC plot for the Watt parameter.

By testing unseen data, the classification results show that our proposed model outperforms the power series based model with a dataset (Dataset B) of fewer image inputs as given by the kappa index in Table 3. The agreement between the raters is higher in the image-based system than for the power series system.

By carrying out more simulations and comparisons we were able to improve the results in Table 2 to those shown in Table 4 for I_{rms} based classification of four appliances.

In Fig. 31 we show the designed classification training and validation model characteristics for the refrigerator

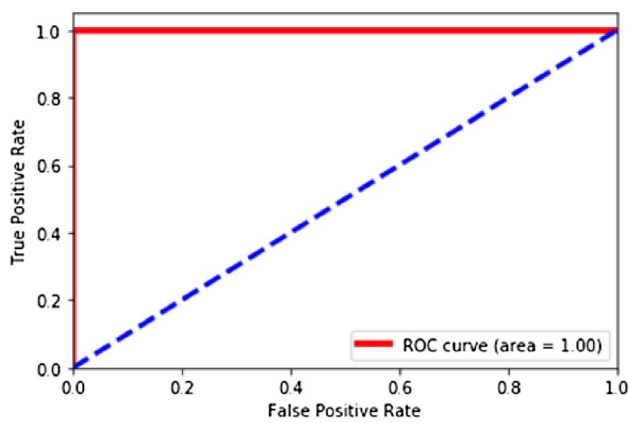


Fig. 25 ROC curve for PF RF and MW appliance classification in power series recognition for data

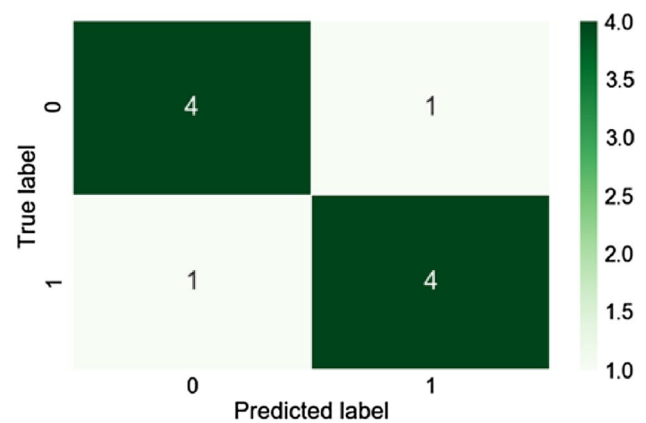


Fig. 28 Confusion matrix for PF RF and MW appliance unseen power series data classification

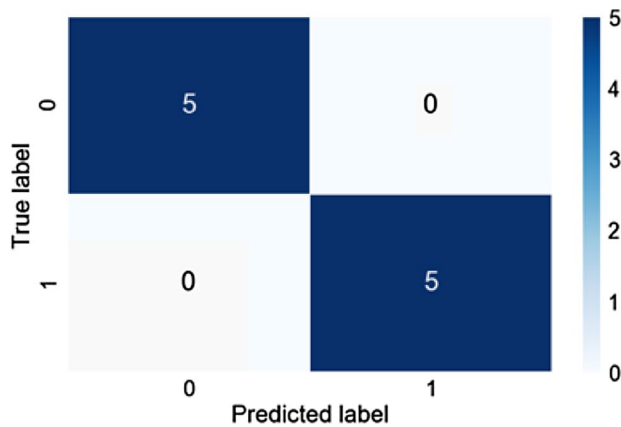


Fig. 26 Confusion matrix for PF RF and MW appliance seen power series data classification

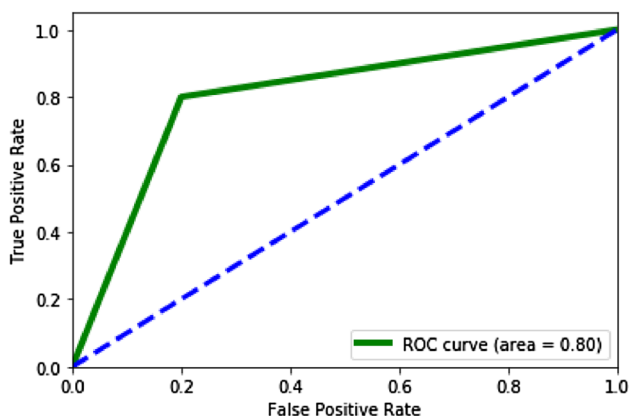


Fig. 27 ROC curve for PF RF and MW appliance classification in power series recognition for unseen data

(RF), microwave oven (MW), 12 W CFL (L1) mains lamp, and 5 W LED (L2) mains lamp.

We obtained 100% classification of three of the four appliance disaggregated images as shown in Fig. 32 confusion matrix. The poor recognition result of the LED (L2) lamp could be attributed to an insufficiently designed system which needs to factor in the low appliance signal which could be taken as noise in the system.

The results show that we can successfully implement an entirely image-based NILM mains load status recognition system and achieve acceptable results. This has the effect of considerably reducing the model dataset and pre-processing of raw data to be input into the neural network. In the classification model, we achieved acceptable values of accuracy, recall, precision and F-measure. We also achieved an overall appliance recognition rate of 75%. This is a good recognition rate considering that we had a model simulation platform which did not allow for extensively deeper models to be simulated. The disaggregation performance plots show the stability of the autoencoder model, and the training and validation losses decrease together as is expected in a good model. In the classification, we used balanced data to give a stable recognition model for the four appliances. To assess how good the disaggregation is we reconstructed the refrigerator I_{rms} disaggregated signal from the image gramian diagonal matrix, and found that it closely resembles the refrigerator I_{rms} ground truth signal as shown in Fig. 14.

Conclusion

The research objective of designing an image-based NILM recognition systems has been achieved in this paper. We have managed to provide extraction of appliance signal features in a simpler way by adopting an image-based deep learning self-feature extraction method. Secondly, by basing the recognition system on a computer vision approach that possesses a high input receptive field, we increase the

Table 2 Comparative evaluation of model performance on different signal parameters generated images

Parameter (appliance)	Precision	Recall	F1	Accuracy	Train sample	Test sample
L _{rms} (FR)	0.5	0.75	0.60	0.75	15	3
L _{rms} (MW)	0.5	0.25	0.33	0.75	15	3
Watt (FR)	1.00	0.67	0.80	0.8333	15	3
Watt (MW)	0.75	1.00	0.86	0.8333	15	3
PF (RF)	1.00	1.00	1.00	1.00	15	3
PF (MW)	1.00	1.00	1.00	1.00	15	3

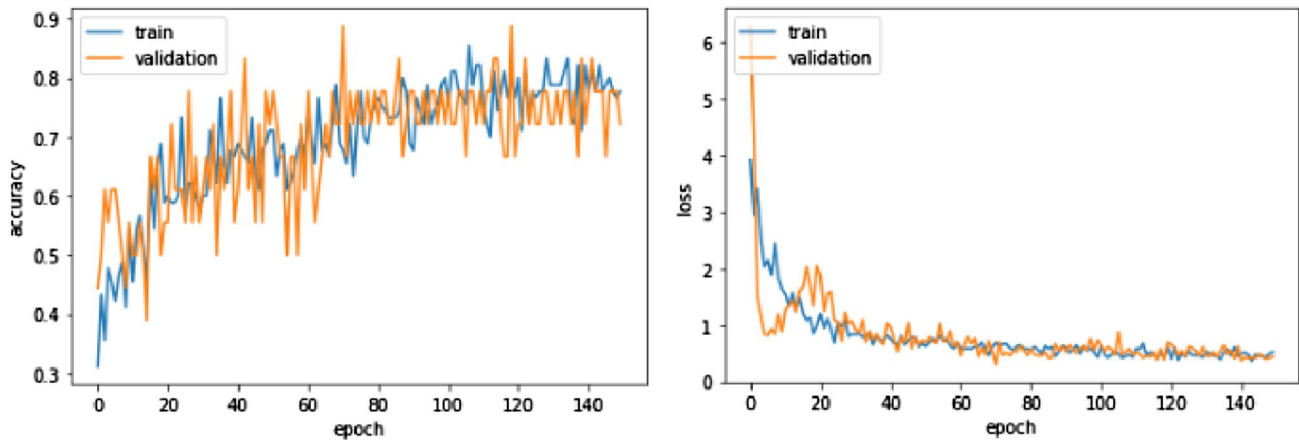


Fig. 29 Arms performance plots. **a** Model accuracy, and **b** categorical crossentropy loss

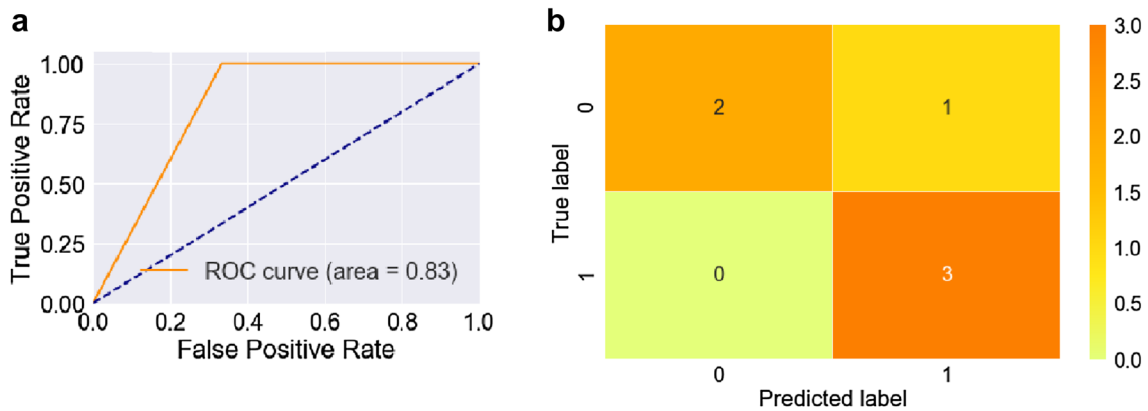


Fig. 30 Watt performance diagrams. **a** ROC, and **b** confusion matrix

field and depth of the features that we can extract without much need of data preprocessing. Immediate outcomes of this approach are the dispensing with the direct power series method, the reduction in the dataset and a high-performance system that is easier to handle. Under the constrained CPU platform that we did our simulations, we show that all the appliance parameters are capable of achieving acceptable NILM appliance recognition performance. However, with a detailed model design it is possible to achieve higher recognition performance.

In this paper, we obtain a rich set of localized aggregate and load features for more accurate NILM recognition through transforming the power series into the image by way of the Gramian Angular Fields technique. For recognition, we used a two-dimensional convolutional neural network that is very well adapted to computer vision applications and possesses a very high image feature extraction and detection capabilities. In this paper, the deep convolutional neural network is configured for both image classification and disaggregation. The disaggregation is in the form of an image-based denoising autoencoder model. We are able to harness

Table 3 AUC and Kappa index (*k*) values for the image and time series models

Model	AUC	p_0	p_yes	p_no	pe	<i>k</i>
Image based	1.0	1.0	0.25	0.25	0.5	1.0
Power series based	0.83	0.66	0.083	0.466	0.5	0.333

Table 4 Classification report for the recognition of the four appliances

	Precision	Recall	F1-score	Support
CFL mains lamp	0.5	1.00	0.67	1
refrigerator	1.00	1.00	1.00	1
LED mains lamp	0.00	0.00	0.00	1
Microwave	1.00	1.00	1.00	1
Micro avg	0.75	0.75	0.75	4

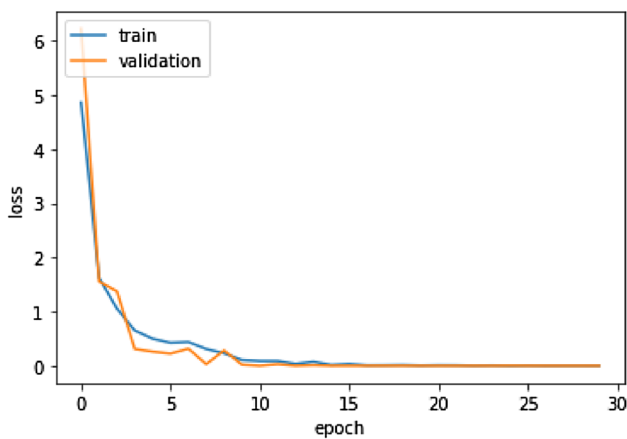


Fig. 31 Categorical crossentropy loss for the recognition of four appliances based on image disaggregation

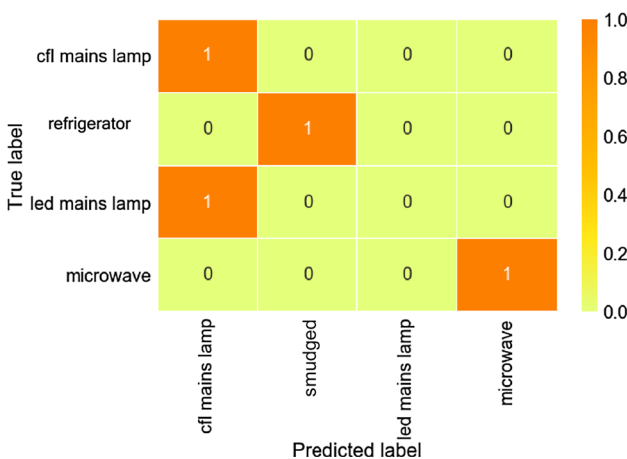


Fig. 32 Confusion matrix for the recognition of four appliances based on image disaggregation

the powerful denoising capabilities of the autoencoder to come up with an effective disaggregation method in the NILM recognition scheme. Our models perform excellent image-based disaggregation and classification respectively. In the final analysis, we compare the performance of our proposed recognition system to that of a one-dimensional power series convolutional neural network recognition system. The results show that our proposed method achieves acceptable performance.

In future, to reduce power series noise and improve on recognition we will consider various sensor (information) fusion techniques that include the Kalman filter, fuzzy fusion etc., and image fusion of the various signal parameter images. We need to investigate a few short learning as a means of increasing the performance and reducing the dataset of the NILM image-based recognition system.

Acknowledgements On behalf of all authors, the corresponding author states that there is no conflict of interest. This study was funded by National Research Foundation (Grant numbers 112108 and 95687, 112108 and 95687), Eskom Tertiary Education Support Programme Grant and URC of University of Johannesburg.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Zhang P et al (2011) An improved non-intrusive load monitoring method for recognition of electric vehicle battery charging load. *Energy Procedia* 12:104–112
- Hossen SS, Agbossour K, Kelouwani S, Cardenas A (2017) Non-intrusive load monitoring through home energy management systems: a comprehensive review. *Renew Sustain Energy Rev* 79:1266–1274. <https://doi.org/10.1016/j.rser.2017.05.096>
- Esa NF, Abdullah MP, Hassan MY (2016) A review disaggregation method in non-intrusive appliance load monitoring. *Renew Sustain Energy Rev* 66:163–173
- Baets LD, Ruyssinck J, Devellder C, Dhaene T, Deschrijver D (2018) Appliance classification using VI trajectories and convolutional neural networks. *Energy Build* 158:32–36

5. Baets LD, Develder C, Dhaene T, Deschrijver D (2019) Detection of unidentified appliances in non-intrusive load monitoring using siamese neural networks. *Electric Power Energy Syst* 104:645–653
6. Kelly J, Knottenbelt W (2015) Neural NILM: deep neural networks applied to energy disaggregation. In: *The 2nd ACM International Conference on embedded systems for energy-efficient built environments*, pp 55–64. <https://doi.org/10.1145/2821650.2821672>
7. He W, Chai Y (2016) An empirical study on energy disaggregation via deep learning. In: *2nd International Conference on Artificial Intelligence and Industrial Engineering (AIIE)*, vol 133, pp 338–342. <https://doi.org/10.2991/aiie-16.2016.77>
8. Zhang Y, Yang G, Ma S (2019) Non-intrusive load monitoring based on convolutional neural network with differential input. In: *11th CIRP Conference on industrial product-service systems*, vol 83, pp 670–674. <https://doi.org/10.1016/j.procir.2019.04.110>
9. Felan Carlo C, Garcia FCC, Creayla CMC, Macabebe EQB (2017) Development of an intelligent system for smart home energy disaggregation using stacked denoising autoencoders. *Procedia Comput Sci* 105:248–255
10. DdeP P, Castro ARC (2018) Home appliance identification for NILM systems based on deep neural network. *Int J Artif Intell Appl* 9(2):69–80
11. Rehman AU, L TT, Valles B, Tito SR (2018) Low complexity event detection algorithm for non-intrusive load monitoring systems. In: *IEEE innovative smart grid technologies-Asia (ISGT Asia)*, Singapore, pp 746–751. <https://doi.org/10.1109/ISGT-Asia.2018.8467019>
12. Meziane MN, Ravier P, Lamarque G, Bunetel Jean-CL, Raingeaud Y (2017) High accuracy event detection for nonintrusive load monitoring. In: *2017 IEEE International Conference on acoustics, speech and signal processing (ICASSP)*, New Orleans, LA, pp 2452–2456. <https://doi.org/10.1109/ICASSP.2017.7952597>
13. Nalmpantis C, Vrakas D (2018) Machine learning approaches for non-intrusive load monitoring: from qualitative to quantitative comparison. *Artif Intell Rev*. <https://doi.org/10.1007/s10462-018-9613-7>
14. Wang H, Zhang Q, Wu J, Pan S, Chen Y (2019) Time series feature learning with labeled and unlabeled data. *Pattern Recogn* 89:55–66
15. Wang Z, Oates T (2015) Imaging time-series to improve classification and imputation. In: *Proceedings of the 24th International Conference on artificial intelligence*. AAAI Press, Buenos Aires, pp 3939–3945. [arXiv:1506.00327](https://arxiv.org/abs/1506.00327) [cs.LG]
16. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. *ICLR, International Conference on Learning Representations*, pp 1–14. [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) [cs.CV]
17. Jia D et al (2020) Identification of electrical equipment based on two-dimensional time series characteristics of power. In: *IOP Conf. Ser.: Mater. Sci. Eng.*, vol 768, p 062019. <https://doi.org/10.1088/1757-899X/768/6/062019>
18. Mottahedi M, Asadi S (2016) Non-intrusive load monitoring using imaging time series and convolutional neural networks. In: *16th International Conference on computing in civil and building engineering*, pp 705–710
19. Yang D, Gao X, Kong L, Pang Y, Bi Z (2020) An event-driven convolutional neural architecture for non-intrusive load monitoring of residential appliance. *IEEE Trans Consum Electron* 66(2):173–182
20. Lam HY, Fung GSK, Lee WK (2007) A novel method to construct taxonomy of electrical appliances based on load signatures. *IEEE Trans Consum Electron* 53(2):653–660
21. Borin VP, Barriquello CH, Campos A (2016) Vector projection classification for home appliances recognition: a load signature comparative analysis. In: *2016 12th IEEE International Conference on industry applications (INDUSCON)*, pp 1–8. <https://doi.org/10.1109/INDUSCON.2016.7874469>
22. Faustine A, Pereira L, Klemenjak C (2020) Adaptive weighted recurrence graphs for appliance recognition in non-intrusive load monitoring. *IEEE Trans Smart Grid*. <https://doi.org/10.1109/TSG.2020.3010621> (Accepted for Publication: January)
23. Anuse A, Vyas V (2016) A novel training algorithm for convolutional neural network. *Complex Intell Syst* 2:221–234. <https://doi.org/10.1007/s40747-016-0024-625>
24. Kollias D, Tagaris A, Stafylopatis A, Kollias S, Tagaris G (2018) Deep neural architectures for prediction in healthcare. *Complex Intell Syst* 4:119–131. <https://doi.org/10.1007/s40747-017-0064-626ss>
25. Yamashita R, Nishio M, Do RKG, Togashi K (2018) Convolutional networks: an overview and application in radiology. *Insights Imaging* 9(4):611–629. <https://doi.org/10.1007/s13244-018-0639-9>
26. Tun NM, Gavrilov AI, Tun NL (2020) Facial image denoising using convolutional autoencoder network. In: *International conference on industrial engineering, applications and manufacturing (ICIEAM)*, Sochi, Russia, pp 1–5
27. Xu J, Zhang L, Zhang D (2015) Denoising convolutional neural network. In: *IEEE Int. Conf. on Information and Automation, Lijiang*, pp 1184–1187. <https://doi.org/10.1109/ICInfA.2015.7279466>
28. Karim F, Majumdar S, Darabi H, Chen S (2017) LSTM fully convolutional networks for time series classification. *IEEE Access* 6:1662–1669. <https://doi.org/10.1109/ACCESS.2017.2779939>
29. Lacko P (2017) From perceptrons to deep neural networks. In: *Proceedings of the IEEE 15th International Symposium on applied machine intelligence and informatics (SAMi)*, Herl'any, pp 000169–000172. <https://doi.org/10.1109/SAMI.2017.7880296>
30. Bonfigli R et al (2018) Denoising autoencoders for non-intrusive load monitoring: improvements and comparative evaluation. *Energy Build* 158:1461–1474
31. Xu Y, Du J, Dai L-R, Lee C-H (2015) A regression approach to speech enhancement based on deep neural networks. *IEEE/ACM Trans Audio Speech Lang Process* 23:7–19
32. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA (2010) Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *J Mach Learn Res* 11:3371–3408
33. Chopra P, Yadav SK (2015) Fault detection and classification by unsupervised feature extraction and dimensionality reduction. *Complex Intell Syst* 1:25–33. <https://doi.org/10.1007/s40747-015-0004-2>
34. Tektronix (2018) PA1000 Power Analyzer User Manual, Tektronix, Inc., USA. https://download.tek.com/manual/PA1000_User_Manual_26.pdf
35. Rosenfeld M, Graham RL et al (eds) (2013) In praise of the gram matrix. *The Mathematics of Paul Erdős's I*, pp 551–557. <https://doi.org/10.1007/978-1-4614-7258-235>
36. Martinez-Arellano G, Terrazas G, Ratchev S (2019) Tool wear classification using time series imaging and deep learning. *Int J Adv Manuf Technol*. <https://doi.org/10.1007/s00170-019-04090-6>
37. Ha KW, Jeong JW (2019) Motor imagery EEG classification using capsule networks. *Sensors* 19(2854):1–20
38. Hanley JA, McNeil B (1982) The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology* 148(1):29–36

39. Chris D (2020) Re: What is the value of the area under the roc curve (AUC) to conclude that a classifier is excellent? https://www.researchgate.net/post/What_is_the_value_of_the_area_under_the_roc_curveAUC_to_conclude_that_a_classifier_is_excellent/5eb02bdf39db6760dc56c904/citation/download. Accessed 06 Jul 2020
40. (2020) ROC or CAP CURVE for a multiclass classification in python. Available via stakeoverflow. <https://stackoverflow.com/questions/49722561/roc-or-cap-curve-for-a-multiclass-classification-in-python>. Accessed 07 Jul 2020
41. (2020) The cross-entropy error function in neural networks. Available via stakexchange. <https://datascience.stackexchange.com/questions/9302/the-cross-entropy-error-function-in-neural-networks> Accessed 15 Jul 2020
42. <https://stackoverflow.com/questions/58735642/how-to-correctly-implement-cohen-kappa-metric-in-keras>
43. Building Autoencoders in Keras (2020) The Keras Blog. <https://blog.keras.io/buildingautoencoders-in-keras.html>. Accessed 15 Jul 2020

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.