Towards Combined Event Detection and Classification for Non-Intrusive Load Monitoring Using Convolutional Neural Networks

Florian Liebgott*, Annika Liebgott* and Bin Yang
Institute of Signal Processing and System Theory, University of Stuttgart, Germany
Email: annika.liebgott@iss.uni-stuttgart.de

***Abstract*—Event-based non-intrusive load monitoring (NILM) commonly consists of two separate modules: an event detector, which identifies state changes in the aggregate power signal, and an event classifier, which determines the appliance causing the change. Consequently, the overall performance of a system for NILM event classification strongly depends on the reliability of the event detector, in addition to the quality of the classifier itself, leading to two possible sources of error within the system. We propose to use an end-to-end approach for simultaneous event detection and classification by training a convolutional neural network on short-time Fourier transform frames of the aggregate power signal. Our experiments on a public dataset show that our combined system performs competitively with very high detection capabilities. Moreover, we show that our method yields the potential to be easily adapted to different data using transfer learning.**

***Index Terms*—Non-intrusive load monitoring, event detection, event classification, CNNs**

I. Introduction

The aim of non-intrusive load monitoring (NILM) is to estimate appliances' individual load curves from an aggregate measurement [1]. A common approach [2], [3] is the construction of load curves based on the appliances' state changes, so-called events. The key idea of an event-based NILM system is, that the load curve of an appliance exhibits specific steady states with nearly constant power levels. The transitions between these steady states are called events. Fig. 1 illustrates this concept. If we detect all events of an appliance, we can then construct its load curve. Clearly, this approach is only suitable for appliances which exhibit steady states.

The essential prerequisite for the event-based approach is a reliable event detection. There exists a multitude of event detection algorithms, most of which use the power signal to detect state changes. They include simple detection, if consecutive samples vary by more than a predefined threshold [1], a generalized likelihood ratio (GLR) test [4], [5] or goodness of fit test [6] to detect changes in the sample distribution, matched filters [7] and clustering of the power data [8]. There are also a few approaches to detect events from the current signal, e.g. by detecting changes in the envelope of the current signal [9] or changes in the current's harmonic composition [10], [11]. A more recent approach to detect events from the current signal employs a denoising autoencoder [12].

*These authors contributed equally to this work.

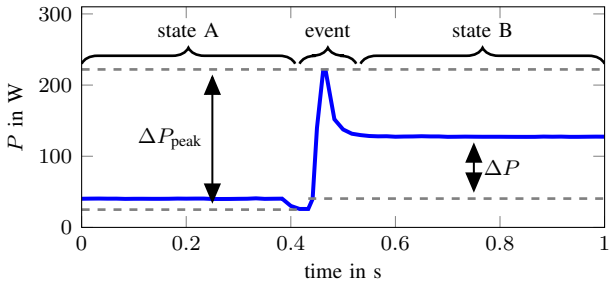

Fig. 1. Example of a switching on event.

For each detected event, features are extracted and used for event classification. The classification of the detected events is carried out with the use of techniques including clustering [1], support vector machines [13], k-nearest neighbor [14] and neural networks [15], [16].

In NILM, deep neural networks (DNN) are mostly used for non-event-based systems [17]–[20]. These systems train one DNN for each appliance to directly produce the load curve or activation curve of this appliance. For training, they need the individual load or activation curve of each appliance. In a practical setting, this would require the individual measurement of all appliances, before a NILM system can be used. An event-based system, on the other hand, can be deployed without having to measure the load of individual appliances. Instead, the events can be detected from the aggregate measurement and the user can be queried to label events, ideally using semi-supervised or active learning to reduce the labeling effort [13].

Fig. 2 shows the usual structure of an event-based NILM system and our contribution, i.e. combining the event detection, feature extraction and event classification stage to improve such systems. We propose to use a convolutional neural network (CNN) to simultaneously detect and classify events from the short-time Fourier transform (STFT) of the aggregate current. To the best of our knowledge, such an approach to event-based NILM has not been studied before.

In addition to merging the aforementioned steps into one end-to-end model, using deep learning yields the potential to efficiently adapt models trained on one dataset to other datasets by employing transfer learning. This is especially interesting for manufacturers who could pre-train a system and then adapt it to the condition at a client's local site. Hence, we also investigate the feasibility of this approach.

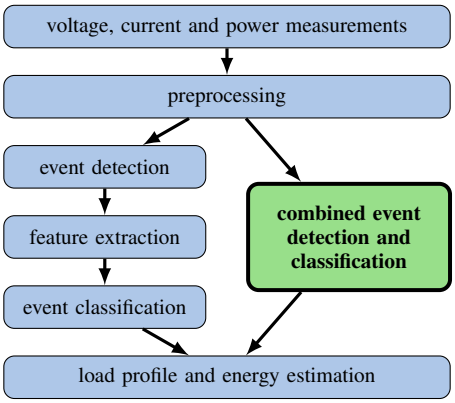

Fig. 2. Common event-based NILM framework and our contribution (green).

II. Methods

A. Data Preprocessing

The main idea behind using the STFT of the aggregate current signal is exploiting the harmonic structure of the current to detect and classify events. To this end, current data sampled with a sampling frequency F_S at least several times the line frequency F_L, i.e. in the kilohertz range, is required. The STFT is calculated over blocks of two periods of the current signal, i.e. with a length of $2\frac{F_S}{F_L}$, resulting in $\frac{F_S}{F_L}+1$ frequency bins. The blocks are windowed with a Hamming window and two adjacent blocks overlap by 50 %.

We used STFT frames with a length of 300 current periods, which corresponds to 5 s in a 60 Hz power grid, as input data. This length was chosen to ensure that most events, which vary significantly in their duration, fit entirely in one frame. We extracted one frame every two current periods, i.e. every $\frac{1}{30}$ s in a 60 Hz power grid.

B. CNN Model for Event Detection and Classification

Initial experiments showed that building a model too deep may lead to an increased risk of overfitting when training on the BLUED dataset. In order to avoid this, we chose to implement an architecture utilizing inception blocks (see Fig. 3). The main advantage of such structures is that applying different convolutional kernel sizes in parallel allows to extract features of varying local regions from an input image. That way, the amount of captured information can be increased without going deeper and risking overfitting. The final model we implemented is based on the first stage of the GoogLeNet architecture [21] and depicted in Fig. 4.

As loss function, we use the categorical cross entropy. We chose ADAM as optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1$), combined with a scheduled exponential learning rate decay starting at an initial rate of 0.01.

C. Data Augmentation

To further decrease the risk of overfitting, we employed data augmentation to our training data. To simulate fluctuations in the measured power signal, we added random normal distributed noise with zero mean and a randomly chosen standard deviation between 0.01 and 0.04 to the STFT frames.

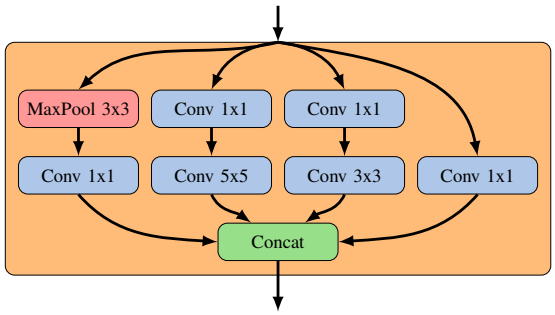

Fig. 3. An inception block as proposed in [21] and used in our model.

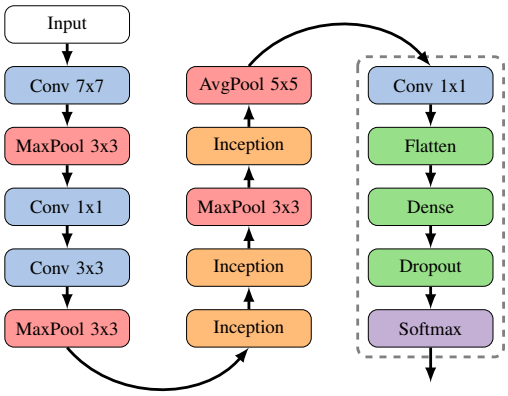

Fig. 4. The architecture of our CNN model. The part, which is adapted to new data by transfer learning in some experiments, is highlighted on the right.

III. Datasets

We tested our approach on two datasets, the "Building-Level fUlly-labeled dataset for Electricity Disaggregation" (BLUED) [5] and our own dataset "ISS kitchen". Both datasets contain event labels, so for each event the beginning and end of the event as well as the appliance causing the event are known.

As the input for the CNN are the STFT frames, we transferred the labels to the frames. To this end, we defined a detection window with a length of 60 current periods located 60 periods from the end of the STFT frame. If this detection window contained the last sample of an event, the label of this event was used as the frame label. If more than one event ended within the detection window, the frame was labeled with the label of the last event ending within the detection window. Additionally, we labeled 5000 randomly selected frames that did not contain any events as "no event". The remaining unlabeled frames were not used in our investigations.

This study is meant as a proof of concept of our ongoing research. We therefore only included frames, in which at least one event ended within the detection window, and frames without any event. We deliberately excluded all other cases from our study, but plan to use them in our next prototype.

A. BLUED

We tested our framework on BLUED [5], which is a publicly available dataset acquired for one week in a single-family home in the USA. It contains measurements of two phases and labels for all events identifying the corresponding appliance. However, the labels only give information about

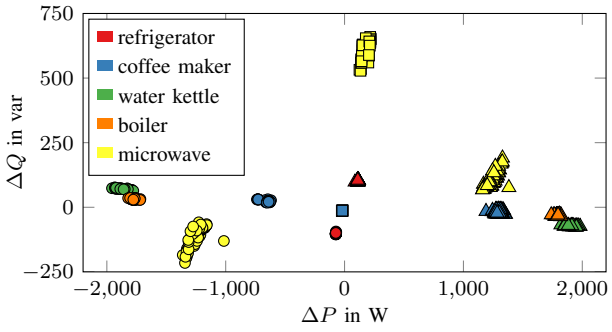

Fig. 5. The ISS kitchen dataset in the ΔP-ΔQ plane. Switching on events are marked with a triangle, switching off events with a circle. Squares mark other state changes.

which appliance an event belongs to, the type of state transition is unknown. We hence chose to identify the individual state changes by applying a clustering technique. For each appliance, we determined the change in active and reactive power ΔP and ΔQ, respectively, during each event and performed a clustering in the ΔP-ΔQ plane. Each resulting cluster was then assumed to belong to one state transition.

In our final datasets, we only included event clusters which contain at least 20 events and assigned each of those clusters a class label. The phase A dataset "BLUED A" consists of 681 events distributed over 9 classes, the phase B dataset "BLUED B" contains a total of 906 events and 21 classes. After preprocessing and transferring the labels to the STFT frames, we had a total of 25,347 samples in the BLUED A dataset and 32,085 samples in BLUED B. These numbers include the additional "no event" class, leading to a total number of 10 classes for BLUED A and 22 classes in the BLUED B dataset. We also combined both phases to form a bigger dataset "BLUED AB" containing a total of 57,432 samples. The eventless classes of both datasets were merged into one common class, leading to a total number of 31 classes.

B. ISS Kitchen

The second dataset we used to investigate the generalizability of our approach was measured in our institute's kitchen. It contains measurements of a refrigerator, a water kettle, a microwave, a coffee maker and a boiler as well as the aggregate signal and was acquired over a period of eight days. In contrast to BLUED, which was measured with 12 kHz in a 120 V/60 Hz power grid, we measured the voltage and aggregate current with a sampling frequency of 10 kHz in a 230 V/50 Hz power grid. The distribution of the 12 classes in the ΔP-ΔQ plane is given in Fig. 5. Each class contains at least 20 samples, which results in 710 samples overall. The final dataset after preprocessing and adding a class for data frames without events contains 33,926 samples in 13 classes.

IV. Experiments and Results

A. Experimental Setup

We used 5-fold cross-validation to train our model. The training set contains 60 % of the samples, while 20 % each were used for validation after each epoch and testing the final model, respectively. Data splitting was performed such that frames belonging to the same event were contained entirely in one of the folds. The training data was then augmented to four times the original size by applying the aforementioned augmentation to each STFT frame three times with different noise. Each model was trained with a batch size of 256 for 100 epochs (BLUED A) or 150 epochs (BLUED B, BLUED AB). As final results to evaluate our models, we used the mean accuracy and recall over all folds.

We first conducted experiments on the combined event detection and classification. To evaluate the results in more detail, we then investigated the event detection and event classification capabilities of our model separately. Additionally, we tested the generalization of the event detection by applying the model trained on BLUED AB to the ISS kitchen dataset.

Finally, we investigated the possibility to adapt our trained model to a new dataset. To this end, we applied transfer learning to a trained model by freezing all layers up to the average pooling layer, i.e. only the last convolutional layer, dense layer and a new classification layer were trained on the unseen dataset, as highlighted in Fig. 4. We conducted two experiments adapting a model trained on BLUED B to BLUED A and on BLUED AB to ISS kitchen, respectively. Each model adaption was trained for an additional 100 epochs.

TABLE I
The mean test accuracies over all folds for the event classification, event detection and the combination of both as well as the event detection recall.

	BLUED A	**BLUED B**	**BLUED AB**
accuracy combined	98.74 %	93.76 %	96.65 %
accuracy classification	98.43 %	93.86 %	96.01 %
accuracy detection	99.73 %	98.30 %	99.65 %
recall detection	99.46 %	97.13 %	99.58 %

B. Results of Event Detection and Classification

The results of our experiments are given in Table I. The combined event detection and classification worked well on all datasets with the highest performance on BLUED A, followed by BLUED AB and BLUED B. Closer investigation of the results revealed that for BLUED A and BLUED AB, there is little confusion between different events and almost no confusion of the "no event" class with any event class. On BLUED B, 7 to 12 % of events with small power changes (monitor and office lights) are predicted as "no event". The model is also at risk of mistaking these classes, which exhibit similar electrical characteristics, for each other.

To be able to compare the results to our previous work on feature-based event classification [13], we examined the classification capabilities of our architecture, i.e. excluding frames from the "no events" class. The results are slightly lower than the values for the combined detection and classification. Again, performance on BLUED A is the highest, followed by BLUED AB and BLUED B.

As the detection of events is crucial for our proposed system to work, we evaluted how well our models are able

to differentiate between the "no events" and the remaining classes. The performance achieved is very high for all three datasets. As detection of all events is more important than accidentally classifying an eventless frame as event, we additionally evaluated the recall, which led to similarly high values.

C. Generalization of the Event Detection

To further investigate how well the model learned to detect events in the STFT frames, we exemplarily applied a model trained on BLUED B to the BLUED A data. With respect to distinguishing between events and eventless frames, the model achieved a recall of 97.48 % and an accuracy of 98.74 %, which is only slightly lower than the performance after training a model directly on BLUED A. Additionally, we applied a model trained on BLUED AB to the ISS kitchen dataset, leading to a recall of 91.02 % and an accuracy of 95.42 %.

D. Adaptation of the Combined Model to New Data

Adapting a model trained on BLUED B to BLUED A via transfer learning led to a test accuracy of 98.48 %, which is less than 0.3 percentage points lower than the performance achieved by training the model on BLUED A from scratch. Using transfer learning to fit a model trained on BLUED AB to the ISS kitchen dataset, we achieved a test accuracy of 90.46 %. The main error of the adapted model is that all boiler events are labeled as water kettle events. As both appliances are resistive loads with almost the same power consumption (see Fig. 5), they exhibit a particularly similar behavior when being switched on or off.

V. Discussion

Our proposed system for combined event detection and classification from STFT frames performed well for all BLUED datasets. The lower performance on the BLUED B dataset compared to BLUED A and BLUED AB data can be attributed to BLUED B containing event classes which are harder to distinguish than those of BLUED A. For example, there are two different event classes containing data from monitors, which exhibit very similar characteristics and are accordingly often confused by our model. Other classes the model has difficulties recognizing are power circuits where different, unidentified appliances are plugged in, which can exhibit considerably varying power characteristics.

Considering the classification of events, the results we achieved with our system cannot fully match the performance of the support vector machine we trained in our previous work [13]. However, in this case we only trained our model to distinguish between event classes without taking detection into consideration. Moreover, the system proposed in this study is designed as a first proof of concept for the feasibility of end-to-end event detection and classification. Further investigation of different architectures, especially such taking temporal dependencies into account, yields room for improvement of our system's performance.

One motivation for our proposed system was the usability within an online setting. With an average overall processing time per frame of 0.3 ms, this requirement is satisfied.

TABLE II
Comparison of the recall for event detection on BLUED.

	BLUED A	**BLUED B**	**BLUED AB**
proposed approach	99.46 %	97.13 %	99.58 %
GLR [5]	98.16 %	70.41 %	–
KFDA of harmonics [11]	98.78 %	92.17 %	–
clustering of P, Q [8]	97.20 %	68.18 %	78.53 %
current envelope [9]	94 %	88 %	–
clustering of P, I_{RMS} [22]	98.70 %	87.85 %	–
MEED (autoencoder) [12]	–	69 %	–

A. Event Detection

Our proposed system performs very well with respect to event detection on all BLUED datasets. The recall of BLUED AB is the highest, which may be due to the higher amount of different event patterns contained in this combined dataset. We compared our results to other known approaches for NILM event detection, for which performance metrics on BLUED have been published. Since we include all event samples, but only a randomly chosen subset of the "no event" samples, we chose to use the recall or true positive rate as metric to compare our results to other event detectors. As shown in Table II, most detectors perform well on BLUED A, although not as well as our proposed approach, but exhibit a significant decrease in performance for BLUED B, where our system still leads to a high recall. An evaluation on BLUED AB is only available for the event detector proposed by Barsim et al. [8], which achieved a significantly lower recall than ours on this dataset.

B. Translation to Other Datasets

Both the application of the event detector to unseen data and the adaptation of a trained model to a new dataset via transfer learning showed promising results in our experiments. Directly employing the trained model of BLUED B for event detection on BLUED A resulted in recall rates a little lower than after direct training. However, comparing them to the recall rates from the proposed event detectors in Table II, the event detection capabilities of our proposed system are definitely competitive. The BLUED AB model being directly able to achieve recall and accuracy over 90 % on the ISS kitchen set, which was measured on a power grid with different voltage and frequency, supports this claim. Training on data acquired in both power systems may further improve the generalization capability.

Regarding transfer learning, results for the translation from BLUED B to BLUED A were almost as good as when training on BLUED A from scratch. Adapting a BLUED AB model to ISS kitchen yielded a little lower test accuracies. However, taking into account yet again the different power grids the measurments were acquired on, these results are very promising. They indicate that the inception blocks in our setup can extract features from the STFT frames that are general

enough to represent events reliably, so the model only needs to learn the correct mapping to the new desired output space.

Easy translation to unknown data is especially interesting to enable pretraining systems on a large annotated dataset by a manufacturer and adapting them to local requirements at a client's site using as little effort as possible. Transfer learning yields the potential to accomplish this task efficiently. In our experiments, we used all available samples in the ISS kitchen dataset. Further reduction in adaptation expenses could be achieved by using only the most significant data, e.g. by employing techniques known from the field of active learning.

C. Outlook

Right now, our model represents a first proof of concept that the simultaneous detection and classification of events in NILM by training a CNN end-to-end is feasible. The determination of the time when the event happens is only an approximation yet and we do not utilize the temporal dependencies in the data. Future work will hence address the temporal dependencies for the exact determination of event start times and duration. Moreover, in a real-world scenario, it is common that new appliances are introduced to a power network at some point, e.g. when a new kitchen appliance is brought into a household. Appliances may also change over time, for example when an old refrigerator is exchanged for a newer model. To address this issue, we plan to make our system adaptive to new circumstances by incorporating an interface for user feedback, which can be utilized to retrain the model using transfer learning combined with methods from the field of active learning. This is especially beneficial to keep the labeling costs for the system adaption as low as possible.

VI. Conclusion

In this study, we investigated the feasibility of an end-to-end approach for simultaneous event detection and classification for non-intrusive load monitoring using deep learning. To this end, we implemented a CNN which we trained on windowed STFT frames of NILM events. We tested our system on the publicly available BLUED dataset (phase A, phase B and both phases combined) and achieved high performance on all of them. Additionally, we investigated the generalizability of our models' detection capabilities and the adaptability of our system to new data by means of transfer learning. To this end, we applied the models trained on BLUED to our own ISS kitchen dataset. The results show, that our proposed system can compete with other approaches to NILM event classification. With respect to event detection, our approach outperforms the state-of-the-art methods significantly.

Overall, we were able to show that combined event detection and classification by training an end-to-end CNN architecture is feasible. Especially the possibility to employ transfer learning techniques, ideally combined with active learning methods facilitating the incorporation of user feedback, yields the potential to adapt trained models to new data in a cost-effective manner and will be addressed in further research.

References

[1] George W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992.

[2] Zeifman M. and Roth K., "Nonintrusive appliance load monitoring: Review and outlook," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 1, pp. 76–84, 2011.

[3] Ahmed Zoha, Alexander Gluhak, Muhammad Ali Imran, and Sutharshan Rajasegarar, "Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey," *Sensors*, vol. 12, no. 12, pp. 16838–16866, 2012.

[4] Dong Luo, Leslie K. Norford, Steven R. Shaw, and Steven B. Leeb, "Monitoring HVAC equipment electrical loads from a centralized location - methods and field test results," in *AHSRAE Transactions*, 2002, vol. 108, pp. 841–857.

[5] Kyle Anderson, Adrian Ocneanu, Diego Benitez, Derrick Carlson, Anthony Rowe, and Mario Bergés, "BLUED: a fully labeled public dataset for Event-Based Non-Intrusive load monitoring research," in *Proceedings of the 2nd SustKDD*, Beijing, China, 2012.

[6] Yuanwei Jin, Eniye Tebekaemi, Mario Berges, and Lucio Soibelman, "Robust adaptive event detection in non-intrusive load monitoring for energy aware smart facilities," in *IEEE ICASSP*, 2011, pp. 4340–4343.

[7] Steven B. Leeb, Steven R. Shaw, and James L. Kirtley, "Transient event detection in spectral envelope estimates for nonintrusive load monitoring," *IEEE Transactions on Power Delivery*, vol. 10, no. 3, pp. 1200–1210, 1995.

[8] Karim Said Barsim and Bin Yang, "Sequential clustering-based event detection for non-intrusive load monitoring," in *CS & IT*, Zurich, Switzerland, 2016, pp. 77–85.

[9] José M. Alcalá, Jesús Ureña, Álvaro Hernández, and David Gualda, "Event-based energy disaggregation algorithm for activity monitoring from a single-point sensor," *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 10, pp. 2615–2626, 2017.

[10] Nabil Amirach, Bernard Xerri, Bruno Borloz, and Claude Jauffret, "A new approach for event detection and feature extraction for NILM," in *21st IEEE ICECS*, 2014, pp. 287–290.

[11] Benjamin Wild, Karim Said Barsim, and Bin Yang, "A new unsupervised event detector for non-intrusive load monitoring," in *IEEE GlobalSIP*, 2015, pp. 73–77.

[12] Daniel Jorde, Matthias Kahl, and Hans-Arno Jacobsen, "MEED: An unsupervised multi-environment event detector for non-intrusive load monitoring," in *IEEE SmartGridComm*, 2019, pp. 1–6.

[13] Florian Liebgott and Bin Yang, "Active learning with cross-dataset validation in event-based non-intrusive load monitoring," in *25th EUSIPCO*, 2017, pp. 296–300.

[14] Marisa B. Figueiredo, Ana de Almeida, and Bernardete Ribeiro, "An experimental study on electrical signature identification of non-intrusive load monitoring (NILM) systems," in *Adaptive and Natural Computing Algorithms*, 2011, vol. 6594 of *Lecture Notes in Comp. Sc.*, pp. 31–40.

[15] D. Srinivasan, W.S. Ng, and A.C. Liew, "Neural-network-based signature recognition for harmonic source identification," *IEEE Transactions on Power Delivery*, vol. 21, no. 1, pp. 398–405, 2006.

[16] Antonio G. Ruzzelli, Clement Nicolas, Anthony Schoofs, and Gregory M. P. O'Hare, "Real-time recognition and profiling of appliances through a single electricity sensor," in *7th IEEE SECON*, 2010, pp. 1–9.

[17] Lukas Mauch and Bin Yang, "A new approach for supervised power disaggregation by using a deep recurrent LSTM network," in *IEEE GlobalSIP*, 2015, pp. 63–67.

[18] Lukas Mauch and Bin Yang, "A novel DNN-HMM-based approach for extracting single loads from aggregate power signals," in *IEEE ICASSP*, 2016, pp. 2384–2388.

[19] Jack Kelly and William Knottenbelt, "Neural NILM: Deep neural networks applied to energy disaggregation," in *Proceedings of the 2nd ACM BuildSys*, 2015, pp. 55–64.

[20] Chaoyun Zhang, Mingjun Zhong, Zongzuo Wang, Nigel Goddard, and Charles Sutton, "Sequence-to-point learning with neural networks for non-intrusive load monitoring," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, vol. 32, pp. 2604–2611.

[21] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," in *IEEE CVPR*, 2015.

[22] Zhuang Zheng, Hainan Chen, and Xiaowei Luo, "A supervised event-based non-intrusive load monitoring for non-linear appliances," *Sustainability*, vol. 10, no. 4, pp. 1001, 2018.