2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)

Electrical Appliance Classification using Deep Convolutional Neural Networks on High Frequency Current Measurements

Daniel Jorde, Thomas Kriechbaumer, and Hans-Arno Jacobsen
Chair for Application and Middleware Systems
Technische Universität München, Germany
Email: daniel.jorde@in.tum.de

***Abstract*—Monitoring the energy demand of appliances can raise consumer awareness and therefore reduce energy consumption. Using a single-point measurement of mains energy consumption can keep costs and hardware complexity to a minimum. This data stream of raw voltage and current measurements can be used in machine learning tasks to extract information. We apply Deep Convolutional Neural Networks on an electrical appliance classification task, using raw high frequency start up events from two datasets. We further present Data Augmentation techniques to improve the model performance and evaluate different data normalization techniques. We achieve a perfect classification on WHITED and a F1-Score of 0.69 on PLAID.**

I. Introduction

Researchers are striving to find solutions to reduce the total energy consumption, due to climatic change and the increasing energy demand of both industrial and residential consumers. By making consumers aware of their detailed energy consumption it is possible to achieve significant energy savings. Energy consumption data from electrical appliances can be recorded and analyzed to provide appropriate feedback to the consumers and the utilities [1], [2]. Other applications are motivated by the insights from performing such analysis: non-invasive monitoring of elderly people [3], anomaly detection of malfunctioning devices to reduce maintenance costs [4], or condition monitoring on naval vessels [5]. The required information to enable such applications can be derived by monitoring the electric appliances. One major approach is Non-Intrusive Load Monitoring (NILM). It aims to use aggregated electrical signals, for example, measured at the electrical mains of a building. NILM is a step-wise process and one of its major steps is to identify individual appliances from the voltage and current signals. The appliance identification problem is particularly challenging and therefore still not sufficiently solved [2], [6]. Past research focused mainly on residential areas and their appliances, which resulted in multiple public datasets of their measured electrical signals [7], [8].

Smart Meters are the most widespread devices to record such aggregated data, usually at a low sampling rate ($\sim$ 1 Hz) [4]. Low frequency data provides only enough information to identify some of the major appliances, making it challenging to identify multiple, smaller ones at the same time [4]. With higher sampling rates, it is possible to distinguish even such appliances. Most of the ongoing research is dealing with low frequency data from Smart Meters due to their growing deployment rate. There is a clear lack of methods for the high frequency domain [9]. Methods using datasets that are sampled with a frequency of 1 Hz or less are considered as low frequency methods according to [10]. We do not take low frequency data into account, due to the previously described limitations. Subsequently, approaches using datasets with a higher sampling frequency are considered as high frequency methods. In order to identify single appliances and to disaggregate loads, most methods use manually derived appliance signatures as features for various supervised machine learning classifiers [2], [9]. Such features require extensive domain knowledge and are dependent on different appliance types [9].

The main contribution of this paper is to investigate the usage of the raw, high resolution current signal in order to classify individual appliances with Deep Convolutional Neural Networks (DCNN). It is valid to assume the switching continuity principle because of the high sampling rate of the measurements [11] and therefore to use datasets that treat the appliance measurements isolated from one another. We compare our DCNN classification approach to the state of the art approaches using handcrafted (manually-derived) features. Furthermore, we investigate multiple Data Augmentation (DA) techniques to improve the performance of the appliance identification classifier. Besides this, we investigate different standardization and scaling methods to normalize the data. Our classifier is designed to distinguish between multiple appliance types. We use publicly available high frequency energy datasets: WHITED [7] and PLAID [8]. These datasets already provide a clean signal trace with annotated device labels suitable for machine learning. Datasets with long-term measurements, such as BLOND [12] or UK-DALE [13], do not provide this level of ground truth granularity. The evaluation considers multiple DCNN architectures and hyperparameter configurations.

The rest of the paper is organized as follows: In Section II, related work is presented. In Section III, we present the DA and the normalization techniques together with our model architecture. Section IV presents the experimental setup, followed by the evaluation of the experiments in Section V. Section VI then concludes this paper.

978-1-5386-7954-8/18/$31.00 ©2018 IEEE

II. Related Work

Appliance identification, a subtask of energy disaggregation, can be modeled as a typical machine learning classification problem. Most of the machine learning techniques rely on using pre-computed sets of features, i.e., individual appliance signatures, to distinguish between appliances [9]. Frequently applied algorithms are Hidden Markov Models [14]–[16], Support Vector Machines [17], [18], and k-Nearest Neigbors [9], [17]. Besides these algorithms, Artificial Neural Networks (ANN) are increasingly applied on the problem, due to their success in other research fields. They are applied to both low [19], [20] and high frequency datasets [21], [22]. When looking at the high frequency domain, most of the works propose simple, fully-connected feedforward neural networks for residential settings [23]–[28]. Apart from fully-connected ANNs, Baets et al. [22] proposed a DCNN for identifying different appliances using VI-trajectory-based appliance signatures. On the other hand, several authors have applied Recurrent Neural Networks (RNNs) on low frequency data to both energy disaggregation [19], [29], [30] and appliance identification [20], [31] with great success, yet high frequency data have not been extensively tested with RNNs. The lack of methods using RNNs in the high frequency domain can be explained by the high computational requirements for high-dimensional (high frequency) input data. Even in the low frequency domain, some works in NILM rely on first using convolutional layers to down-sample the input signal and to detect features before applying recurrent layers to it [19], [29]. Besides the different algorithms that are applied to the task, no approach directly uses the raw signal to perform the appliance classification to the best of our knowledge. Studies have shown that different appliances and system settings require different appliance signatures [9]. Using a high frequency signal, without dropping information by pre-processing it extensively, promises to allow to distinguish between multiple, even smaller appliances [4]. In general, ANNs and in particular DCNNs, are able to efficiently process high dimensional data and to automatically extract meaningful features from it [32]. Problems in other research fields, especially in audio related tasks, exhibit a structure that is similar to the one of the appliance identification problem. In [33], the authors successfully used DCNNs to recognize environmental sounds, supporting the idea to also apply DCNNs to distinguish different appliance types. They propose a DCNN architecture which is able to handle long input sequences, containing up to 32000 data points. The input sequences we use for the appliance identification task exhibit a similar length to the data used in the audio related task [33]. Roos et al. [34] manually derived an appliance hierarchy, using the appliances' inherit electrical components and behaviour. Furthermore, the authors claim that a hierarchical classification approach is likely to enhance the identification capabilities of a classifier. DCNNs are designed to automatically extract hierarchical features while performing the classification task [32]. Therefore, it seems possible that DCNNs can make use of the hierarchy of the appliance types to efficiently perform the classification task. The manually designed appliance hierarchy supports the claim to deliver state of the art results on the identification problem.

III. Appliance Identification Approach

Our classification approach and the experiments we developed are designed based on the following main design principles: *Raw Input* and *Flexibility*. To fulfill the first requirement, as little pre-processing as possible should be applied on the data. Therefore, only minimal normalization and scaling of the raw input signal is to be used. For the *Flexibility* requirement, the chosen model needs to be flexible with respect to new appliances. In order to being able to compare the results to the ones obtained in [9], we used the same datasets and input sizes. Only the current signal is used as an input to the DCNNs in the following, because it exhibits most of the information necessary to perform the identification task [9]. Both datasets are comparatively small with high dimensional samples, resulting in a particular hard classification problem [32]. To overcome this issue, the number of samples is increased by applying two Data Augmentation techniques to both datasets and the architecture of the DCNN is designed to learn hierarchical features while gradually down-sampling the input signal. By doing this, the network learns a lower dimensional feature representation that is used as an appliance signature in the identification task.

A. Data Pre-Processing

We only apply a minimum amount of pre-processing to the raw current signal and we evaluate three pre-processing techniques and their influence on classification performance. Furthermore, we evaluated the performance when applying no pre-processing at all to the input signal. Applying some pre-processing to the data is motivated by the circumstance that neural networks usually benefit from a standardized input [32]. The first technique we applied, is a *z-score* normalization to adjust the input to have zero mean and unit variance by applying:

$$x_{norm} = \frac{x - \mu}{\sigma}$$

To estimate the standard deviation σ, we take the standard deviation over all samples in the dataset. In addition to this, we take the mean of each appliance window to estimate the expected value μ for the normalization. This is a small variation to the procedure proposed by [19], that has shown to produce more normalized input signals. Instead of normalizing the signal, we also only scaled the input by using two different min-max scaling procedures. Therefore, we evaluated scaling the data into two value ranges, i.e., $[0, +1]$ and $[-1, +1]$ respectively. The transformation to the first interval is given by

$$x_{scaled} = \frac{x - min(x)}{max(x) - min(x)}$$

In order to scale the data into the second interval, we used the formula below.

$$x_{scaled} = 2 \times \frac{x - min(x)}{max(x) - min(x)} - 1$$

Besides the presented techniques, we also evaluated the classifier without transforming the input data at all. In contrast to other approaches like [22], no down-sampling is applied before feeding the data into the classifier.

B. Data Augmentation

One of the challenges in applying Deep Learning models to NILM datasets like WHITED and PLAID is the small number of samples per class. Collecting sub-metered data usually requires a high effort and special metering devices [35], especially when sampling at high frequencies [4]. In order to support the training of neural networks, the number of samples can be increased. In addition to this, training the classifier with augmented data has a regularization effect on the model, helping it to generalize better to new samples [32]. Special care must be given to DA transformations to avoid changing the class of the sample. In order to increase the dataset size by an order of magnitude, we applied two transformations to the samples:

1) *Phase Shift*: Shift the activation window by one phase.
2) *Half-Phase Flip*: Shift the activation window by a half-phase and flip the signal, i.e., change the signs.

A phase shift increases the time in-variance, whereas a half-phase flip increases the flexibility of the classifier to variations in the signal. Figure 1 shows the DA process. First, the original window is shifted one phase to the left to generate a sample. Afterwards, the signal is shifted by a half-phase and the signs of the measurement points are inverted.

C. Model Architecture

Appliance identification can be modelled as a multi-class classifications problem, where each instance gets assigned exactly one out of many possible classes. To increase the flexibility of the classifier and to facilitate parallel training and inference, we reformulated the multi-class problem into multiple binary ones for an One-versus-All (OvA) approach. We trained one binary classifier per appliance type and aggregated the results afterwards. This results in n classifiers for the n appliance types we want to distinguish, with the n_i^{th} classifier's positive output belonging to appliance class $c_i \in C$. Each of the binary classifiers outputs two probabilities, one for the positive and one for the negative class (i.e., all other classes). In the aggregation step we then take the highest positive probability over all classifiers and assign it to the particular sample. If, for each classifier, the negative class probability is higher then the positive one, we assign the sample to a "none" class. Using a "none" class allows us to group and further investigate samples which are unknown to the classifier. As appliance identification is one of the base steps for a lot of applications, miss-classifications can have a huge effect on the whole process. The conservative treatment of samples for which the network is uncertain about helps to detect problems early in the analytical pipeline. Each binary classifier is a fully DCNN with the following architectural components: The non-linearities we used in the hidden layers of the networks are Rectified Linear Units (ReLU). Furthermore, we chose to use the Nesterov Accelerated Gradient optimizer, with a momentum weight of 0.9 [36]. For parameter initialization we used a Glorot initializer [37]. Works like [33], [38] have shown how one can learn useful features from high-dimensional input signals by gradually down-sampling the input when it passes through the network. To prevent the network from overfitting the data, we used L2-regularization and early stopping. Motivated by the findings of Dai et al. [33] on the similar audio classification tasks, we used large receptive fields in the convolutional layers and multiple pooling layers to handle the input. We then designed three DCNN architectures and evaluated them (Table I). Some of the hyperparameters differ slightly between the datasets due to the different size of the input vectors. The notation for the convolutional layers is as follows: (filter,kernel,stride) x number. For the pooling layers, we denoted the kernel and the stride we used.

To obtain variations of the base architecture we stacked different amounts of convolutional layers on one another. Besides this, we evaluated multiple hyperparameters (e.g., stride and kernel sizes) to obtain a configuration that produces the best results over all binary classifiers. All the pooling and the convolutional layers use *same* zero-padding, except the Avg-Pooling layer, which uses *valid* padding. It takes the average value of every feature map at the end and produces, after flattening the output, a one-dimensional representation for the appliances. We apply the *softmax* function at the end of the pipeline to condition the output signal. The cost function we use is the cross-entropy cost function. When building multiple binary classifiers, the distribution of the class labels in the training data becomes highly imbalanced. For each classifier n_i the amount of data for the class c_i is much smaller then the count of samples labelled with the negative class. Many real-world datasets are highly imbalanced, therefore a lot of solutions have been proposed by researchers from different fields [39]. This extrinsic between-class-imbalance can be solved by various methods, one type being cost-sensitive approaches. In general, one uses a cost matrix to assign different cost values to the classes in order to adjust for the imbalance. In the binary case, one typically uses two values Cost(*Maj,Min*) and Cost(*Min,Maj*), with the first being the cost to classify one sample from the minority class as a majority one and the latter one exactly the other way round. To prevent imbalances, we assigned a higher cost to miss-classifying minority class samples, i.e., Cost(*Maj,Min*) $>$ Cost(*Min,Maj*) [39]. In [40] the authors evaluated multiple cost-sensitive approaches for neural network classifiers. Among the proposed ones, we decided to choose an approach that aims to adapt the network output by assigning class specific costs because of its simplicity. The factor we used to scale the outputs stems from the relative class imbalance, i.e. we use the proportion of minority class to majority class samples.

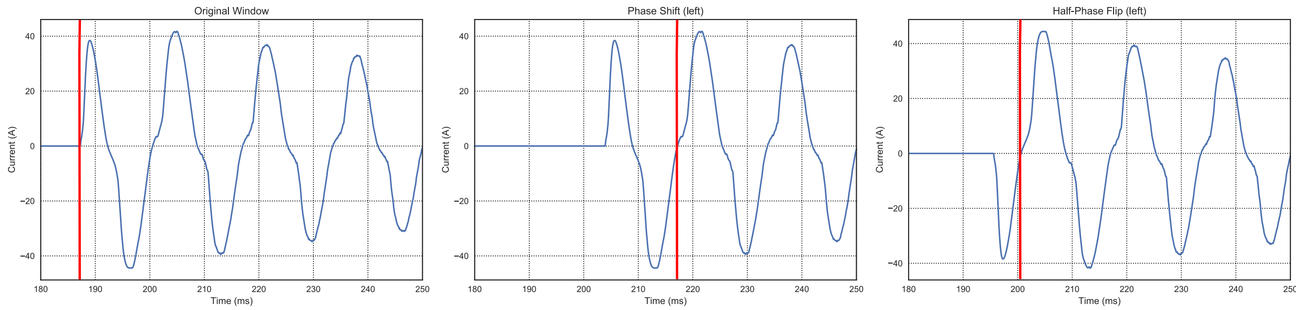

Fig. 1. Augmentation techniques applied to the activation window of a vacuum from PLAID [8]

TABLE I
BASIC DCNN ARCHITECTURES FOR WHITED AND PLAID

	WHITED			PLAID		
	DCNN4	DCNN5	DCNN6	DCNN4	DCNN5	DCNN6
Input Layer		22050			15000	
Convolutional Layer		(64, 80, 2) x 1			(64, 80, 2) x 1	
Max-Pooling Layer		5, 5			3, 3	
Convolutional Layer	(64, 3, 1) x 1	(64, 3, 1) x 2	(64, 3, 1) x 2	(64, 2, 1) x 1	(64, 2, 1) x 2	(64, 2, 1) x 2
Max-Pooling Layer		3, 3			4, 4	
Convolutional Layer	(128, 3, 1) x 1	(128, 3, 1) x 1	(128, 3, 1) x 2	(128, 3, 1) x 1	(128, 3, 1) x 1	(128, 3, 1) x 2
Max-Pooling Layer		5, 5			5, 5	
Convolutional Layer		(256,3,1)			(256,3,1)	
Max-Pooling Layer		2, 2			5, 5	
Avg-Pooling Layer		49, 1			25, 1	
Softmax-Output Layer		2			2	

IV. EXPERIMENTAL METHODOLOGY

We conducted several experiments to select the best performing DCNN architecture and hyperparameter settings and to evaluate the pre-processing and DA techniques. For model selection, we trained three binary classifiers on three representative appliances. We then used the results from these experiments to train one DCNN per appliance type in the dataset. To compare the performance of our models to the ones using handcrafted features, we adapted the experimental setup from an extensive feature study by Kahl et al. [9].

A. Datasets

The DCNNs are trained on start-up events of WHITED [7] and PLAID [8]. Both datasets are sampled with high frequencies: WHITED with 44.1 kHz [7] and PLAID with 30 kHz [8]. Similar to [9] we further adapted the 500ms activation window size and used subsets of the two datasets. This activation window size results in input vectors containing 22050 measurement points for WHITED and 15000 for PLAID. For WHITED, we used a typical household subset with 27 appliance types. PLAID on the opposite contains multiple models per appliance type. We used all of the 11 appliance types in the dataset. Therefore, we trained 27 binary classifiers for WHITED and 11 for PLAID. To increase the sample sizes of the datasets, we applied the previously described Data Augmentation techniques. We first applied the Phase Shift four times and afterwards the Half-Phase Flip transformation to each sample. This results in a 10 factor increase of the amount of data. For the experiments, we split the data into 80% for training and divided the remaining data into equal parts for validation and testings.

B. Metrics

To evaluate the performance of the appliance classifiers, we apply the commonly used F1-Score. The F1-Score is based on using the amount of True Positives (*TP*), False Positives (*FP*), True Negatives (*TN*), and False Negatives (*FN*). The F1-Score is defined as follows:

$$F1\text{-}Score = \frac{2 \times recall \times precision}{recall + precision}$$

$$precision = \frac{TP}{TP + FP} \qquad recall = \frac{TP}{TP + FN}$$

To compute the performance over all appliance models for a given architectural configuration, we take the non-weighted average score over all appliances and compute a macro F1-Score. We do not report the Accuracy metric, because it can be deceiving in settings with highly imbalanced data [39].

V. EXPERIMENTAL RESULTS

Looking at the results of the experiments for both datasets, one can see that the effect of the normalization procedures and the model architectures are highly dataset dependent. For the final model, we selected the model configuration that performs best over all appliance modes to obtain a general setting. Another approach is to select different configurations for the individual binary models, possibly leading to better classification results. We chose the first approach, because it generalizes better and requires less human interference. The best F1-Scores for the respective datasets and model architectures are shown in Table II. For WHITED, the DCNN4

TABLE II
BEST F1-SCORES ON THE AUGMENTED DATASETS

Dataset	Metric	DCNN4	DCNN5	DCNN6
WHITED	F1-Score	**1**	0.91	0.59
PLAID	F1-Score	0.65	**0.69**	0.58

architecture turned out to perform best on the augmented data without normalizing it, using the corresponding hyper-parameter configurations in Table III. The kernel and filter size parameters alter the respective values in the first layer of the architecture and subsequently the upper layers according to the base architecture (Table I). The model configurations

TABLE III
HYPERPARAMETER CONFIGURATIONS FOR THE BEST MODELS

Model	Learning Rate	Batch Size	Kernel	Filter
WHITED DCNN4	0.01	25	80	64
PLAID DCNN5	0.01	40	126	128

with a high learning rate and a small batch size performed best for both datasets. When looking at the normalization methods we applied, one can see that the z-normalization and applying no normalization clearly outperformed both min-max scaling approaches. We chose to apply no normalization in the final model instead of using z-normalization, because it follows the declared minimal pre-processing approach and preserves all information in the raw data. The best model achieved a macro F1-Score of **1** (Figure 2). Over all parameter configurations, the models on the augmented data have a higher average F1-Score compared to the models trained on the non-augmented data. Despite this, some configurations still achieved a F1-Score of 1, even on the non-augmented data. The best performing approach in [9] also achieves a macro F1-Score of 1, but requires extensive feature engineering to do so. For PLAID, the best model, our DCNN5 model, achieved a macro F1-Score of **0.69** (Figure 3). For both of the confusion matrices it has to be noted that the "none" class is omitted from the results, because the information is intrinsically contained in the matrices. In contrast to the results on WHITED, the min-max scaling outperforms the other normalization techniques, followed by applying no normalization to the data. The effect of the Data Augmentation techniques is significant over all architectures. The best DCNN4 architecture achieved a F1-Score of 0.65 on the augmented data and only a score of 0.28 on the non-augmented one. The other classifiers showed similar results, further supporting the effect of the proposed Data Augmentation techniques. When comparing the results to the ones by Kahl et al. on PLAID [9], one sees that our approach is outperformed. The authors' handcrafted feature based k-Nearest Neighbor approach achieved a macro F1-Score of 0.89 [9]. Despite this, our experiments clearly showed the feasibility of using the raw current signal as an input to perform the classification task.

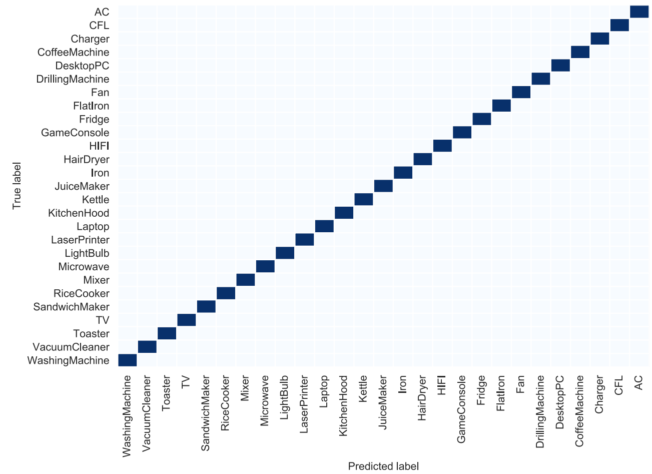

Fig. 2. Confusion Matrix of the DCNN4 architecture on WHITED

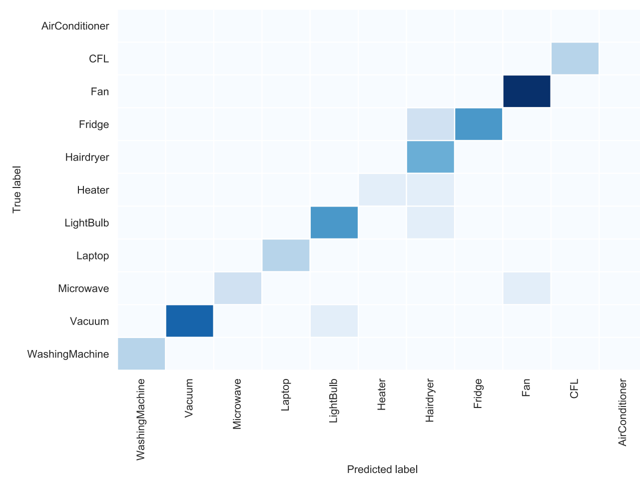

Fig. 3. Confusion Matrix of the DCNN5 architecture on PLAID

VI. CONCLUSIONS

In this paper, we developed Deep Convolutional Neural Networks to identify electrical appliances from raw, high frequency activation events. The results show that by applying Data Augmentation techniques and by carefully selecting among different pre-processing techniques, we achieved state of the art results on WHITED and good ones on PLAID. We performed as well as classifiers based on extensive feature engineering on WHITED, and showed the feasibility of our approach on PLAID. Our approach has the advantage that no explicit feature engineering by a domain expert is necessary. Different appliance types require different sets of features to identify them [9] and therefore significant feature engineering by a domain expert. Our approach circumvents this, because it provides a generic approach to the problem. This is particularly useful in volatile settings with a lot of different appliances.

ACKNOWLEDGMENT

This research was partially funded by the Alexander von Humboldt Foundation established by the government of the Federal Republic of Germany.

REFERENCES

[1] M. R. Asghar, G. Dán, D. Miorandi, and I. Chlamtac, "Smart meter data privacy: A survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2820–2835, 2017.

[2] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-Intrusive Load Monitoring Approaches for Disaggregated Energy Sensing: A Survey," *Sensors*, vol. 12, no. 12, pp. 16 838–16 866, 2012.

[3] J. M. Alcalá, J. Ureña, Á. Hernández, and D. Gualda, "Assessing Human Activity in Elderly People Using Non-Intrusive Load Monitoring," *Sensors*, vol. 17, no. 2, p. 351, 2017.

[4] K. Carrie Armel, A. Gupta, G. Shrimali, and A. Albert, "Is Disaggregation the Holy Grail of Energy Efficiency? The Case of Electricity," *Energy Policy*, vol. 52, pp. 213–234, 2013.

[5] T. DeNucci, R. Cox, S. B. Leeb, J. Paris, T. J. McCoy, C. Laughman, and W. C. Greene, "Diagnostic Indicators for Shipboard Systems using Non-Intrusive Load Monitoring," in *IEEE Electric Ship Technologies Symposium*. Piscataway, NJ: IEEE, 2005, pp. 413–420.

[6] G. W. Hart, "Nonintrusive Appliance Load Monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992.

[7] M. Kahl, A. U. Haq, T. Kriechbaumer, and H.-A. Jacobsen, "WHITED - A Worldwide Household and Industry Transient Energy Data Set," in *3rd International Workshop on Non-Intrusive Load Monitoring*, 2016.

[8] J. Gao, S. Giri, E. C. Kara, and M. Bergés, "PLAID: A Public Dataset of High-resolution Electrical Appliance Measurements for Load Identification Research," in *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*. New York, NY, USA: ACM, 2014, pp. 198–199.

[9] M. Kahl, A. U. Haq, T. Kriechbaumer, and H.-A. Jacobsen, "A Comprehensive Feature Study for Appliance Recognition on High Frequency Energy Data," in *Proceedings of the 8. ACM e-Energy conference*, vol. 5. New York, NY, USA: ACM, 2017, pp. 121–131.

[10] J. Liang, S. K. K. Ng, G. Kendall, and J. W. M. Cheng, "Load Signature Study—Part II: Disaggregation Framework, Simulation, and Applications," *IEEE Transactions on Power Delivery*, vol. 25, no. 2, pp. 561–569, 2010.

[11] S. Makonin, "Investigating the Switch Continuity Principle assumed in Non-Intrusive Load Monitoring (NILM)," in *2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*. Piscataway, NJ, USA: IEEE, 2016, pp. 1–4.

[12] T. Kriechbaumer and H.-A. Jacobsen, "BLOND, a building-level office environment dataset of typical electrical appliances," *Scientific Data*, vol. 5, 2018.

[13] J. Kelly and W. Knottenbelt, "The UK-DALE Dataset: Domestic Appliance-Level Electricity Demand and Whole-House Demand from Five UK Homes," *Computing Research Repository*, vol. abs/1404.0284, 2014.

[14] Z. J. Kolter and T. Jaakkola, "Approximate Inference in Additive Factorial HMMs with Application to Energy Disaggregation," in *Proceedings of Machine Learning Research*, 2012.

[15] M. Zhong, N. Goddard, and C. Sutton, "Signal Aggregate Constraints in Additive Factorial HMMs, with Application to Energy Disaggregation," in *Advances in Neural Information Processing Systems*, 2014, pp. 3590–3598.

[16] J. A. Mueller, A. Sankara, J. W. Kimball, and B. McMillin, "Hidden Markov models for nonintrusive appliance load monitoring," in *2014 North American Power Symposium (NAPS)*, 2014, pp. 1–6.

[17] O. Kramer, T. Klingenberg, M. Sonnenschein, and O. Wilken, "Non-Intrusive Appliance Load Monitoring with Bagging Classifiers," *Logic Journal of IGPL*, vol. 23, no. 3, pp. 359–368, 2015.

[18] L. Du, Y. Yang, D. He, R. G. Harley, T. G. Habetler, and B. Lu, "Support Vector Machine Based Methods for Non-Intrusive Identification of Miscellaneous Electric Loads," in *38th Annual Conference on IEEE Industrial Electronics Society*, 2012, pp. 4866–4871.

[19] J. Kelly and W. Knottenbelt, "Neural NILM," in *Proceedings of the 2nd ACM BuildSys*, D. Culler, Ed. New York NY: ACM, 2015, pp. 55–64.

[20] T.-T.-H. Le, J. Kim, and H. Kim, "Classification Performance using Gated Recurrent Unit Recurrent Neural Network on Energy Disaggregation," in *2016 ICMLC*. Piscataway, NJ: IEEE, 2016, pp. 105–110.

[21] H. Lange and M. Berges, "The Neural Energy Decoder: Energy Disaggregation by Combining Binary Subcomponents," in *3rd International Workshop on Non-Intrusive Load Monitoring*, 2016.

[22] L. de Baets, J. Ruyssinck, C. Develder, T. Dhaene, and D. Deschrijver, "Appliance Classification using VI Trajectories and Convolutional Neural Networks," *Energy and Buildings*, vol. 158, pp. 32–36, 2018.

[23] L. de Baets, C. Develder, D. Deschrijver, and T. Dhaene, "Automated Classification of Appliances using Elliptical Fourier Descriptors," in *8th IEEE International Conference on Smart Grid Communications (SmartGridComm 2017)*, 2017, pp. 1–6.

[24] H.-H. Chang, K.-L. Lian, Y.-C. Su, and W.-J. Lee, "Energy Spectrum-Based Wavelet Transform for Non-Intrusive Demand Monitoring and Load Identification," in *2013 IEEE Industry Applications Society annual meeting*. Piscataway, NJ, USA: IEEE, 2013, pp. 1–9.

[25] K. S. Barsim, L. Mauch, and B. Yang, "Neural Network Ensembles to Real-time Identification of Plug-level Appliance Measurements," in *3rd International Workshop on Non-Intrusive Load Monitoring*, vol. 2, 2016.

[26] J. Liang, S. K. K. Ng, G. Kendall, and J. W. M. Cheng, "Load Signature Study—Part I: Basic Concept, Structure, and Methodology," *IEEE Transactions on Power Delivery*, vol. 25, no. 2, pp. 551–560, 2010.

[27] Y.-H. Lin and M.-S. Tsai, "A Novel Feature Extraction Method for the Development of Nonintrusive Load Monitoring System based on BP-ANN," in *2010 International Symposium on Computer, Communication, Control and Automation* , Q. Luo, Ed. Piscataway, NJ: IEEE, 2010, pp. 215–218.

[28] 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.

[29] W. He and Y. Chai, "An Empirical Study on Energy Disaggregation via Deep Learning," in *2nd International Conference on Artificial Intelligence and Industrial Engineering (AIIE 2016)*, ser. Advances in Intelligent Systems Researc. Paris, France: Atlantis Press, 2016.

[30] L. Mauch and B. Yang, "A New Approach for Supervised Power Disaggregation by using a Deep Recurrent LSTM Network," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. Piscataway, NJ: IEEE, 2015, pp. 63–67.

[31] J. Kim, T.-T.-H. Le, and H. Kim, "Nonintrusive Load Monitoring Based on Advanced Deep Learning and Novel Signature," *Computational Intelligence and Neuroscience*, vol. 2017, no. 2-3, pp. 1–22, 2017.

[32] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Cambridge, Massachusetts and London, England: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org/

[33] W. Dai, C. Dai, S. Qu, J. Li, and S. Das, "Very Deep Convolutional Neural Networks for Raw Waveforms," *2017 IEEE ICASSP*, pp. 421–425, 2017.

[34] J. G. Roos, I. E. Lane, E. C. Botha, and G. P. Hancke, "Using Neural Networks for Non-Intrusive Monitoring of Industrial Electrical Loads," in *IEEE Instrumentation and Measurement Technology Conference, 1994*. Piscataway, NJ,USA: IEEE, 1994, pp. 1115–1118.

[35] T. Kriechbaumer, A. Ul Haq, M. Kahl, and H.-A. Jacobsen, "MEDAL: A Cost-Effective High-Frequency Energy Data Acquisition System for Electrical Appliances," in *Proceedings of the 8th International Conference on Future Energy Systems*. New York, NY, USA: ACM, 2017, pp. 216–221.

[36] Y. Nesterov, "A Method for Unconstrained Convex Minimization Problem with the Rate of Convergence O(1/kˆ2)," *Doklady AN USSR*, vol. 269, pp. 543–547, 1983.

[37] X. Glorot and Y. Bengio, "Understanding the Difficulty of Training Deep Feedforward Neural Networks," in *Proceedings of the 13. International Conference on Artificial Intelligence and Statistics*, 2010, pp. 249–256.

[38] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "WaveNet: A Generative Model for Raw Audio," *CoRR*, 2016.

[39] H. He and E. A. Garcia, "Learning from Imbalanced Data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[40] M. Kukar and I. Kononenko, "Cost-Sensitive Learning with Neural Networks," in *Proceedings of the 13th European Conference on Artificial Intelligence*. John Wiley & Sons, 1998, pp. 445–449.