

Received September 21, 2021, accepted October 4, 2021, date of publication October 8, 2021, date of current version October 18, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3118947

Deep Transfer Learning-Based Feature Extraction: An Approach to Improve Nonintrusive Load Monitoring

DIEGO L. CAVALCA¹ AND RICARDO A. S. FERNANDES^{1,2}, (Member, IEEE)

¹Graduate Program in Computer Science, Federal University of São Carlos, São Carlos, São Paulo 13565-905, Brazil

²Department of Electrical Engineering, Federal University of São Carlos, São Carlos, São Paulo 13565-905, Brazil

Corresponding author: Ricardo A. S. Fernandes (ricardo.asf@ufscar.br)

This work was supported in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior—Brazil (CAPES)—Finance Code 001, and in part by São Paulo Research Foundation (FAPESP) under Grant 2019/15192-9.

ABSTRACT The development of techniques that allow the efficient identification of residential loads (nonintrusive load monitoring) is a key factor for the practical implementation of demand response programs. Recently, in terms of nonintrusive load monitoring, the use of deep learning has gained attention, mainly the models based on convolutional neural networks. However, the efficient training of these models is strongly dependent on the quantity and balance of the data, i.e., characteristics that are not normally found in nonintrusive load monitoring datasets. To deal with these challenges, this paper proposes an approach based on three stages, that are: (i) time series transformation into 2D images; (ii) feature extraction using deep transfer learning; and (iii) classification/labelling of loads. Moreover, it was analyzed and defined the better window size per load in relation to the f1-score reached by the classifiers. In this sense, it was considered five loads present in the Reference Energy Disaggregation Dataset, where the proposed approach was able to obtain an average f1-score of 83.2%. From the results analysis, it was demonstrated the greater capacity of the proposed approach to infer and generalize its responses.

INDEX TERMS Convolutional neural network, deep transfer learning, feature extraction, nonintrusive load monitoring, recurrence plots.

I. INTRODUCTION

Although the technological evolution has allowed the development/improvement of electrical devices, there is an increase in electricity consumption. In this sense, the identification of consumers' electricity usage profile becomes a valuable task, mainly to demand response programs [1]. To this end, nonintrusive load monitoring (NILM) emerged as a research area that effectively contributes to this identification, mainly for the residential sector [2]–[4]. Thus, NILM approaches make it possible to identify the operating status of each device and, consequently, disaggregate the electricity consumption.

Researches on NILM began in the 90's, where the first proposed approaches basically analyzed real and reactive power to identify the residential loads [5]–[10]. However, due to the advances in the artificial intelligence area, it is

possible to verify the increase use of machine learning-based approaches [11]–[13]. From these advances, as shown in the review proposed by [14], NILM researches were divided into: (i) micro level – that considers loads' current signatures, i.e., using meters with high sampling rates that are, consequently, more expensive; and (ii) macro level – that uses currents or apparent power in RMS (Root Mean Square) and, in this case, the meters may have a low sampling rate, being more adequate as a real life application.

Within micro level approaches, the feature engineering stage (extraction and selection) is quite common [11], [12], [15]–[18]. This characteristic allows to advance the state-of-the-art, reaching high precision results in the identification task. On the other hand, macro level approaches, mainly due to the use of RMS values, focused on the proposition of new supervised [19]–[22] and unsupervised [23]–[31] algorithms to identify residential loads.

According to the previously mentioned context, it is noted that feature engineering for macro level has become a

The associate editor coordinating the review of this manuscript and approving it for publication was Amin Zehabian¹.

research gap, since it is not trivial to transform RMS values into other features. However, due to the advances in deep learning algorithms for different research areas, the possibility of transforming time series data into 2D images gained notoriety.

In this sense, as expected, the use of time series transformation as a feature extraction stage and deep learning algorithms were primarily performed to the micro level. In [32] and [33] the authors transformed voltage and current signals into V-I trajectories. From these 2D images, they have used a convolutional neural network (CNN) to identify the residential loads. To the best of our knowledge, in terms of macro level, only the framework proposed in [34] transforms the RMS current time series into recurrence plots (2D images), using them as inputs to a CNN.

Given the intrinsic feature extraction process performed by CNNs and the potential of recurrence plots (RP) to generate 2D images that highlight recurring actions in the state space, this paper seeks to advance the state-of-the-art by proposing/investigating:

- A deep transfer learning feature extraction (DTLFE) stage, which was proposed to improve the identification of residential loads. The effectiveness of DTLFE was evidenced by using classical machine learning classifiers, that are, multilayer perceptron (MLP), support vector machine (SVM) and extreme gradient boosting (XGBOOST);
- The better window size per load to improve the identification performance. Since the windowing process is normally performed by the meter, the definition of a maximum window size is important to determine the hardware buffer size, i.e., this information could assist in the specification of a meter to be properly used for NILM purposes.

The remainder of the paper is organized as follows. Section II addresses the most related approaches found in the literature. Section III details the proposed methodology, highlighting the DTLFE. The results are discussed in the Section IV. Finally, the conclusions are presented in Section V.

II. RELATED WORKS

This section brings the works that use data from the Reference Energy Disaggregation Dataset (REDD) to evaluate their performances, more specifically those that consider the house 3. This consideration was made, since house 3 is commonly used to validate novel approaches. The characteristics of this dataset will be presented in more details in subsection III-A.

Kolter and Johnson [35] presented the first results using REDD as benchmark dataset in 2011. They used a Factorial Hidden Markov Model (FHMM) to identify the loads, which was able to achieve an accuracy of only 0.333 for house 3. Due to this work, others were motivated to explore classifiers based on Hidden Markov Model (HMM). This way, in [36] the authors proposed a Bayesian HMM, obtaining an

accuracy of 0.815, i.e., a great advance when compared to [35]. Also, it can be highlighted the HMM-based approaches of [37] and [38], which reached accuracies of 0.906 and 0.800, respectively. However, the REDD can be considered as an unbalanced dataset, since loads like the fridge are constantly on, while loads like the microwave are sporadically used. For this reason, the most recent work using FHMM [39] employs f1-score to assess its performance instead of accuracy, demonstrating a great result by reaching 0.809.

A multi-label classification framework was proposed by [40]. The computational experiments were conducted in order to evaluate the performances of k-Nearest Neighbors (kNN) and RANdom k-lAbELsets (RAkEL) classifiers. In addition, the authors analyzed the use of power time series and its decomposition by using a Wavelet transform. However, the directly use the power time series, when classified by the kNN, presented the better average f1-score (0.530).

In [41], it was proposed the use of a Long Short-Term Memory (LSTM) without any feature engineering stage. The accuracy obtained for house 3 was about 0.920, i.e., greater than that reached by [37].

Sparse coding-based approaches have been recently proposed [22], [42], [43], presenting the advantage of using fewer training samples. Due to this characteristic, sparse coding algorithms are less prone to the effects of unbalanced datasets. Despite this factor, these approaches were able to reach accuracies between 0.465 and 0.650. Using fewer training samples, the authors of [44] investigated the behaviour of semi-supervised learning algorithms. It was observed that Manifold Regularization presented the best overall results. For house 3, it obtained an accuracy of 0.892.

Kong *et al.* [45] proposed an approach that uses HMM to model the home appliances and the Segmented Integer Quadratic Constraint Programming (SIQCP) to disaggregate the consumption. This approach shows good accuracy, obtaining 0.835.

Recently, a framework based on RP and CNN was evaluated by [34]. It was possible to demonstrate the robustness of the proposed framework, which reaches average f1-score and accuracy of 0.727 and 0.956, respectively. However, as pointed out by the authors, these results clearly demonstrate the unbalance inherent to the REDD.

Based on the above-mentioned researches, the lack of feature engineering techniques that are effective for macro level is evident. In this sense, the approach proposed in this paper aims to fill this research gap, ensuring that even classical machine learning algorithms can demonstrate high performances.

III. PROPOSED METHODOLOGY

As previously mentioned, the proposed methodology seeks to demonstrate the potential of DTLFE stage to improve the identification of residential loads. A general overview of this methodology is shown in Fig. 1, being each stage detailed in the sequence.

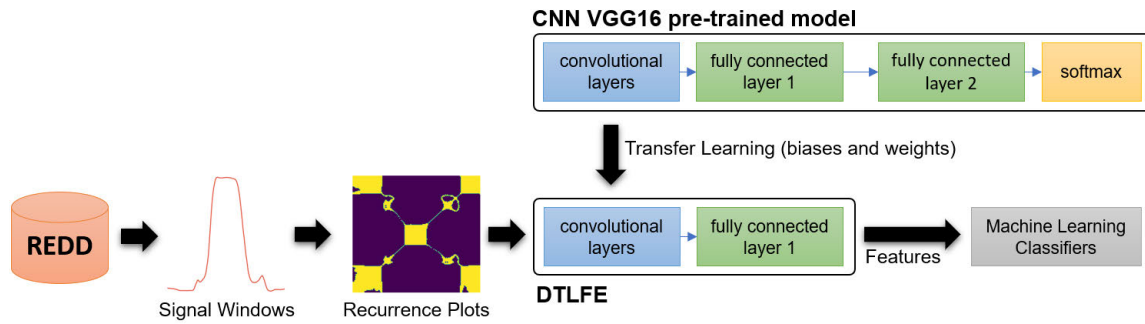


FIGURE 1. Overview of the proposed methodology.

The entire methodology was implemented by using Python programming language, considering the following packages: (i) PyTS to extract the RPs; (ii) Tensorflow/Keras to implement the CNN; and Scikit-learn/XGBOOST to run the other machine learning classifiers. The source code was published on Github.¹ For the computations, an Intel Core i7-7700 CPU (3.60GHz) with 16GB DDR4 RAM and NVIDIA Quadro M5000 was used.

A. DATASET AND PREPROCESSING

First of all, it is important to mention that REDD [35] is divided into low and high frequency measurements acquired from six houses. However, it was considered only the low frequency measurements obtained for the house 3. These data were acquired from the main distribution panel at 1 Hz. The training set was composed of data from 16th April 2011 to 16th May 2011 (70% of available data), while the test set was composed of data from 17th May 2011 to 30th May 2011 (30% of remaining data).

Next, the RMS current time series were processed using a sliding window. Thus, the definition of the window size is an important part of the NILM problem, since a small window can lose information that characterize the load operation cycle and a large window can include noise and/or redundant information. For this reason, the training and test sets were further subdivided according to the size of the window (in seconds), that are: 30, 60, 90, 180, 360, 540, 720, 900, 1080 and 2040.

Each obtained window have its class labelled using the timestamps of the main distribution panel and the timestamps of the measurements acquired from the individual loads. In this sense, it was generated a binary array with 5 dimensions, representing single-state (microwave) and multiple-state loads (fridge, dishwasher, washer dryer 1, washer dryer 2). These loads were chosen in accordance with their energy consumption in the house 3, i.e., the loads with more contribution on the aggregated energy consumption.

B. RMS CURRENT TIME SERIES TRANSFORMATION

From the windowed RMS current data, it was possible to transform them into RPs, which are commonly used to

identify patterns in nonlinear and dynamic systems (mainly when dealing with time series). Considering that these patterns are recurrent [46], the RP allows to visualize them [47] as 2D images. An RP is defined as a $M \times M$ matrix generated from a time series with M samples, being expressed by (1):

$$R_{i,j} = \Theta(\varepsilon - x'_j - x'_i), \tag{1}$$

where $x' \in \mathbb{R}^n$; $i, j = 1, 2, \dots, M$; M is the number of states (x'_i or x'_j) considered; n is the immersion dimension; ε is the radius of the neighborhood (threshold) at the sample (x'_i or x'_j); and $\Theta(\cdot)$ is the Heaviside function.

Therefore, if $R_{i,j} = 1$, the state is recurrent and a black pixel is marked on the graph; and if the $R_{i,j} = 0$, the state is non-recurrent and a white pixel is marked. In this way, each RP has differences in terms of texture (isolated pixels, diagonal, vertical and horizontal lines) and typology [48].

In this paper, the RPs were specifically parameterized for NILM applications, being $\varepsilon = 10\%$ and $n = 1$. These parameters were obtained after exhaustive tests.

C. DEEP TRANSFER LEARNING FEATURE EXTRACTION

Considering the RPs (2D images) as inputs, this stage was proposed to extract features from them. For this purpose, a CNN was used, as this model is able to capture the singularities present in the images, usually arranged in large dimensional spaces [49]. Despite its advantages in recognizing patterns on images, the computational resource and the volume of data necessary for training are limiting factors for real life applications, as is the NILM case. For this reason, a transfer learning strategy was adopted. Thus, a pre-trained model (CNN VGG16) [50] was reused.

However, in this paper, the CNN VGG16 was just used to extract features. Its original architecture is divided into five convolutional layers and three subsequent fully-connected layers, in addition to the softmax output layer. Thus, its last fully-connected and softmax output layers were removed, resulting in the convolutional architecture showed in Fig. 2. Thus, the first fully-connected layer was maintained, since it represents a feature vector (embeddings) used as input to train and validate the ML models presented in the sequence.

¹<https://github.com/diegocavalca/phd-thesis>

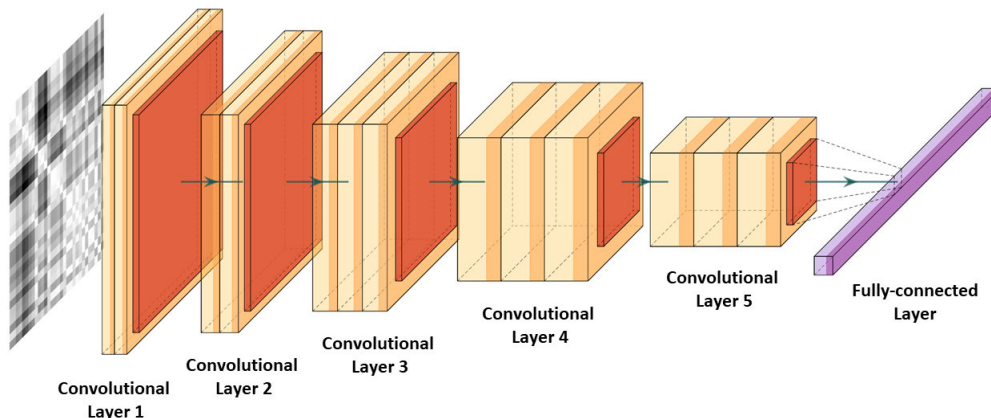


FIGURE 2. CNN VGG16 architecture used in the DTLFE.

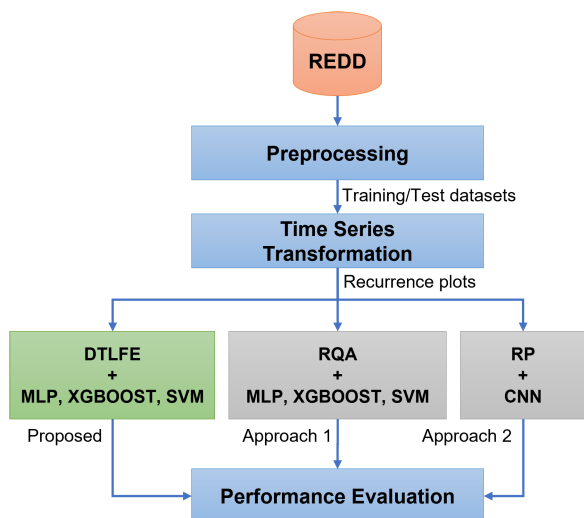


FIGURE 3. Block diagram that represents the comparative analysis considered in this study.

D. CLASSIFICATION USING MACHINE LEARNING

In the last stage of the proposed methodology, the feature vector extracted by the DTLFE was presented to the following classifiers: MLP, SVM and XGBOOST. In order to assess the performance of the DTLFE in conjunction with the classifiers, two other comparative approaches were also considered: (1) based on the extraction of features by means of recurrence quantification analysis (RQA), using the indicators called determinism and recurrence rate as inputs to the three same classifiers; and (2) based on the use of RPs as inputs to a CNN classifier, such as proposed in [34]. A block diagram representing the comparison test is presented in Fig. 3.

Since this paper is focused on improvements in terms of feature extraction, the classifiers were parameterized with default values, that are: MLP (α equals to $1e^{-3}$, hidden layer size equals to 10); SVM (radial basis function kernel); XGBOOST (number of estimators equals to 100).

From the identification of loads reached by the machine learning classifiers, their performances were evaluated according to the most used metrics for NILM, accuracy (acc) and f1-score ($f1$), which can be described as:

$$acc = \frac{TP + TN}{TP + FP + TN + FN}, \tag{2}$$

$$f1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}, \tag{3}$$

being $Precision = \frac{TP}{TP+FP}$, $Recall = \frac{TP}{TP+FN}$, TP is the number of true positive results, TN is the number of true negative, FP is the number of false positive and FN is the number of false negative.

In the sequence, the results were analyzed and discussed to demonstrate the robustness and effectiveness of the DTLFE against the other two approaches considered in this paper and the state-of-the-art works.

IV. RESULTS AND DISCUSSIONS

The results were obtained considering the previously mentioned test dataset for different window sizes. Thus, they were analyzed in terms of: (A) overall performance using a fixed window size to compare the DTLFE with the two other approaches highlighted in Fig. 3; (B) DTLFE generalization capacity; (C) impact of window sizes on the performance; and (D) comparison with the related works.

A. OVERALL PERFORMANCE ANALYSIS

As can be seen in Fig. 4, the DTLFE provided the best average accuracy and f1-score for a fixed window size of 2,040 seconds (the maximum window size considered). Since the dataset is unbalanced (i.e., the amounts of samples per class are different), the use of accuracy to measure the performance of classifiers can be biased. In this sense, the use of the f1-score metric becomes adequate. Thus, the performance gain of the DTLFE when compared to the other two approaches that use RQA and RP as feature extraction methods, ranged between 6.4% and 22.4%.

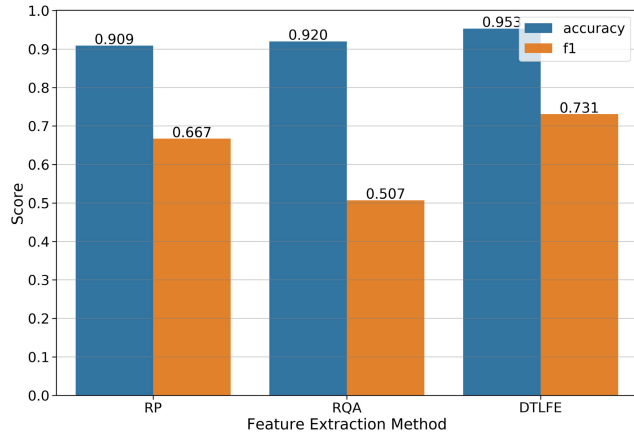


FIGURE 4. Average accuracy and f1-score reached by the DTLFE compared with RP and RQA feature extraction approaches.

TABLE 1. Performances obtained for compositions between feature extraction approaches and classifiers using a fixed window size.

| Feature Extraction | Classifier | acc | f1 |
|--------------------|------------|-------|-------|
| DTLFE | MLP | 0.952 | 0.802 |
| DTLFE | SVM | 0.956 | 0.696 |
| DTLFE | XGBOOST | 0.951 | 0.695 |
| RP | CNN | 0.909 | 0.667 |
| RQA | XGBOOST | 0.917 | 0.562 |
| RQA | MLP | 0.921 | 0.479 |
| RQA | SVM | 0.921 | 0.479 |

TABLE 2. MLP performances to classify each load using a fixed window size of 2,040 seconds.

| Loads | On samples | Off samples | acc | f1 |
|-----------------|------------|-------------|-------|-------|
| Dish washer | 12 | 217 | 0.932 | 0.625 |
| Fridge | 228 | 1 | 0.993 | 0.832 |
| Microwave | 22 | 207 | 0.898 | 0.744 |
| Washer dryer #1 | 14 | 215 | 0.973 | 0.909 |
| Washer dryer #2 | 14 | 215 | 0.966 | 0.898 |

Analyzing the compositions between feature extraction approaches and classifiers, the dominance of the DTLFE was also noticed, as shown in Table 1 (ordered by f1-score).

The combination of DTLFE with the MLP classifier obtained the best result in relation to f1-score and the difference in relation to the other approaches ranged from 10.6% to 32.3%. This result shows that the proposed approach was efficient even in view of the different loads considered and the evident unbalance between the classes (labelled as “on” or “off”) present in the REDD. The results of MLP for each load and the number of “on” and “off” samples are shown in Table 2.

Considering the best performance presented by DTLFE in conjunction with an MLP-based classifier, a generalization analysis was performed, as shown in the next subsection.

B. GENERALIZATION ANALYSIS

In order to assess the generalization of MLP learning, it was submitted to a test dataset with randomly chosen samples. For this evaluation, a 10-fold cross-validation strategy was

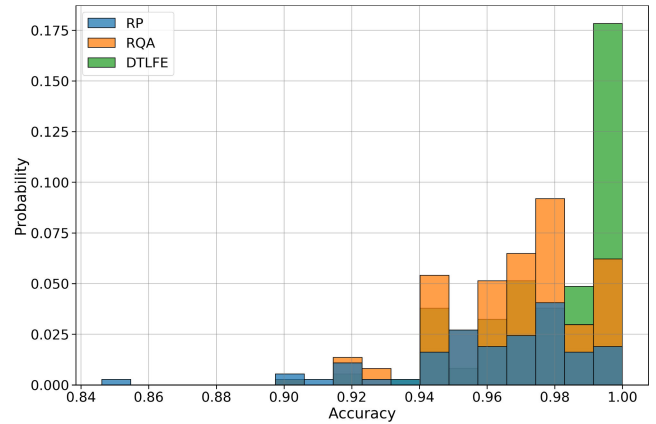


FIGURE 5. Probability distributions obtained for accuracy.

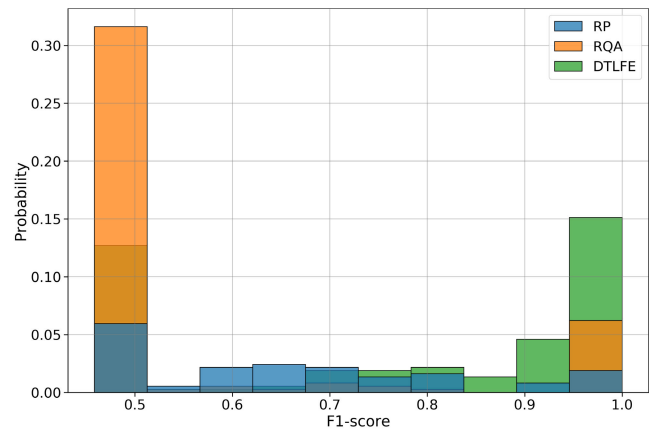


FIGURE 6. Probability distributions obtained for f1-score.

adopted. Thus, each of the feature extraction approaches were analyzed, where only in the case of using RP a classifier based on CNN was considered instead of the MLP. The probability distributions obtained for accuracy and f1-score are respectively presented in Figs. 5 and 6.

In general, it was possible to observe that the DTLFE presents a good probability distribution, indicating that it has the best learning and generalization capacity when compared to the other two approaches discussed in this paper. In time, it is interesting to note that the approach using RQA features provides the most unstable results in terms of f1-score. This way, such features do not provide enough information for MLP training and, consequently, impair its generalization.

C. IMPACT OF WINDOW SIZE

Despite the use of fixed window size is common in NILM researches, as previously stated, this size is a factor that directly affects the performances of some approaches. This is because the window size must be adjusted so that it is not so small as to lose load operating cycles and not so wide to have great overlap with other loads and to maximize the data unbalance, being this a non trivial task. Based on this statement, in this paper, it was analyzed the impact of window

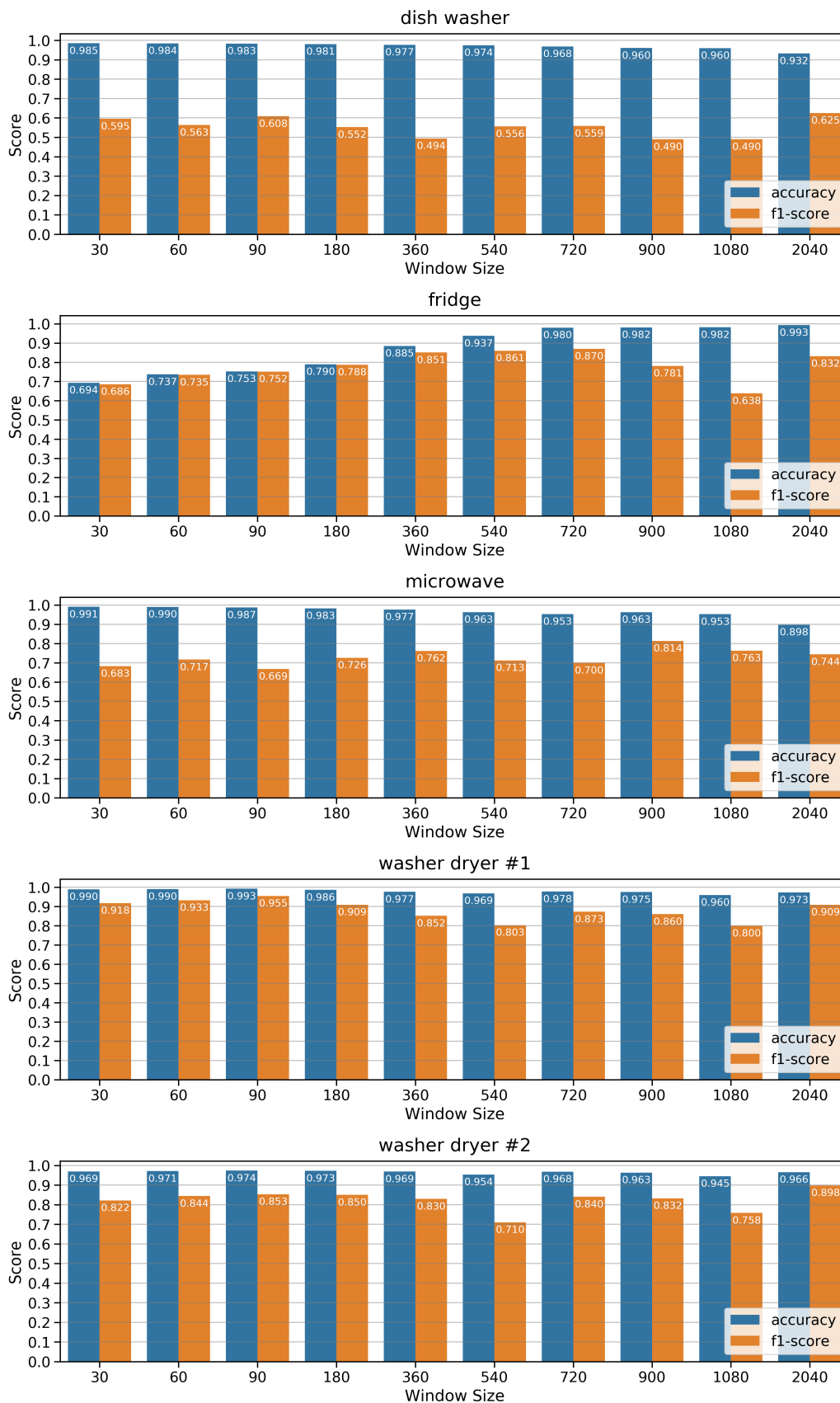


FIGURE 7. Impact of window size on the results of an MLP classifier for each load, considering the use of DTLFE proposed approach.

TABLE 3. Comparing the f1-score between fixed and variable sliding window.

| Loads | Fixed window (f1) | Variable window (f1) |
|----------------------|-------------------|----------------------|
| Dishwasher | 2,040 (0.625) | 2,040 (0.625) |
| Fridge | 2,040 (0.832) | 720 (0.870) |
| Microwave | 2,040 (0.744) | 900 (0.814) |
| Washer Dryer #1 | 2,040 (0.909) | 90 (0.955) |
| Washer Dryer #2 | 2,040 (0.898) | 2,040 (0.898) |
| Mean f1-score | 0.802 | 0.832 |

TABLE 4. Comparison with the state-of-the-art papers.

| Reference | Identification Method | Results |
|-------------------|-------------------------------------|---------------------------|
| [22] | Co-sparse coding | acc = 0.465 |
| [34] | CNN | f1 = 0.726 |
| [35] | FHMM | acc = 0.333 |
| [36] | Bayesian HMM | acc = 0.815 |
| [37] | HMM | acc = 0.906 |
| [38] | HMM | acc = 0.800 |
| [39] | FHMM | f1 = 0.809 |
| [40] | kNN | f1 = 0.530 |
| [41] | LSTM-RNN | acc = 0.920 |
| [42] | Dictionary learning + Sparse coding | acc = 0.650 |
| [43] | Deep sparse coding | acc = 0.483 |
| [44] | Manifold regularization | acc = 0.892 |
| [45] | HMM + SIQCP | acc = 0.835 |
| Proposed approach | DTLFE + MLP (fixed window size) | acc = 0.937 f1 = 0.802 |
| Proposed approach | DTLFE + MLP (variable window size) | acc = 0.967 f1 = 0.832 |

size per load, considering the DTLFE in conjunction with an MLP classifier. The results are shown in the graphs of Fig. 7.

It is possible to observe that the window size has an impact on the f1-score, since the results vary, from fixed to variable sliding windows, in a range between 3.8% and 7% (Table 3). Furthermore, in average, there was an increase of 3% (in terms of f1-score) when using different windows per load.

D. COMPARATIVE ANALYSIS WITH RELATED WORKS

Moreover, Table 4 presents a high-level comparison with the state-of-the-art papers that considered the house 3 of REDD, independent on their methodological aspects. The proposed approach achieved the best performances for f1-score and accuracy, considering both fixed or variable sliding window per load. Therefore, the use of the proposed DTLFE demonstrated to be a robust feature extraction technique.

V. CONCLUSION

The use of a methodology based on multiple stages (time series transformation into images, automatic extraction of features using deep transfer learning and classification) has shown effective results regarding the task of identifying loads to disaggregate their electricity consumption. The proposed DTLFE approach was able to extract features that contribute to better classify the loads and to improve the generalization of an MLP classifier. In addition, the impact of window size on the classifier's performance was mitigated by using different sizes of window for each load.

Given the results obtained by the proposed methodology, it is still possible to analyze, in future research, the capacity of the pre-trained model to perform the transfer learning in relation to other houses contained in REDD or consider a fine-tuning strategy for the same domain (i.e., using other public NILM datasets). Moreover, it can be analyzed an optimized parameterization of RPs and other techniques to overcome the obstacles generated by the unbalanced data.

ACKNOWLEDGMENT

The authors would like to thank the Federal University of São Carlos for the facilities provided and the NVIDIA Corporation for the donation of the Quadro M5000 used for this research.

REFERENCES

- [1] A. L. A. da Fonseca, K. M. S. Chvatal, and R. A. S. Fernandes, "Thermal comfort maintenance in demand response programs: A critical review," *Renew. Sustain. Energy Rev.*, vol. 141, May 2021, Art. no. 110847.
- [2] S. S. Hosseini, K. Agbossou, S. Kelouani, and A. Cardenas, "Non-intrusive load monitoring through home energy management systems: A comprehensive review," *Renew. Sustain. Energy Rev.*, vol. 79, pp. 1266–1274, Nov. 2017.
- [3] D. A. M. Lemes, T. W. Cabral, G. Fraidraich, L. G. P. Meloni, E. R. De Lima, and F. B. Neto, "Load disaggregation based on time window for HEMS application," *IEEE Access*, vol. 9, pp. 70746–70757, 2021.
- [4] Y.-H. Lin, "Trainingless multi-objective evolutionary computing-based nonintrusive load monitoring: Part of smart-home energy management for demand-side management," *J. Building Eng.*, vol. 33, Jan. 2021, Art. no. 101601.
- [5] J. Powers, B. Margossian, and B. A. Smith, "Using a rule-based algorithm to disaggregate end-use load profiles from premise-level data," *IEEE Comput. Appl. Power*, vol. 4, no. 2, pp. 42–47, Apr. 1991.
- [6] F. Sultanem, "Using appliance signatures for monitoring residential loads at meter panel level," *IEEE Trans. Power Del.*, vol. 6, no. 4, pp. 1380–1385, Oct. 1991.
- [7] G. W. Hart, "Nonintrusive appliance load monitoring," *Proc. IEEE*, vol. 80, no. 12, pp. 1870–1891, Dec. 1992.
- [8] S. B. Leeb, S. R. Shaw, and J. L. Kirtley, Jr., "Transient event detection in spectral envelope estimates for nonintrusive load monitoring," *IEEE Trans. Power Del.*, vol. 10, no. 3, pp. 1200–1210, Jul. 1995.
- [9] L. K. Norford and S. B. Leeb, "Non-intrusive electrical load monitoring in commercial buildings based on steady-state and transient load-detection algorithms," *Energy Buildings*, vol. 24, no. 1, pp. 51–64, 1996.
- [10] L. Farnaccio and R. Zmeureanu, "Using a pattern recognition approach to disaggregate the total electricity consumption in a house into the major end-uses," *Energy Buildings*, vol. 30, no. 3, pp. 245–259, 1999.
- [11] D. Srinivasan, W. S. Ng, and A. C. Liew, "Neural-network-based signature recognition for harmonic source identification," *IEEE Trans. Power Del.*, vol. 21, no. 1, pp. 398–405, Jan. 2006.
- [12] R. A. S. Fernandes, I. N. da Silva, and M. Oleskovicz, "Load profile identification interface for consumer online monitoring purposes in smart grids," *IEEE Trans. Ind. Informat.*, vol. 9, no. 3, pp. 1507–1517, Aug. 2013.
- [13] C. C. Yang, C. S. Soh, and V. V. Yap, "A systematic approach in appliance disaggregation using k-nearest neighbours and naive Bayes classifiers for energy efficiency," *Energy Efficiency*, vol. 11, no. 1, pp. 239–259, Jan. 2018.
- [14] J. Liang, S. K. K. Ng, G. Kendall, and J. W. M. Cheng, "Load signature study—Part I: Basic concept, structure, and methodology," *IEEE Trans. Power Del.*, vol. 25, no. 2, pp. 551–560, Apr. 2010.
- [15] H. Y. Lam, G. S. K. Fung, and W. K. Lee, "A novel method to construct taxonomy electrical appliances based on load signatures," *IEEE Trans. Consum. Electron.*, vol. 53, no. 2, pp. 653–660, May 2007.
- [16] M.-S. Tsai and Y.-H. Lin, "Modern development of an adaptive non-intrusive appliance load monitoring system in electricity energy conservation," *Appl. Energy*, vol. 96, pp. 55–73, Aug. 2012.
- [17] T. Hassan, F. Javed, and N. Arshad, "An empirical investigation of V-I trajectory based load signatures for non-intrusive load monitoring," *IEEE Trans. Smart Grid*, vol. 5, no. 2, pp. 870–878, Mar. 2014.

- [18] A. L. Wang, B. X. Chen, C. G. Wang, and D. Hua, "Non-intrusive load monitoring algorithm based on features of V-I trajectory," *Electr. Power Syst. Res.*, vol. 157, pp. 134–144, Apr. 2018.
- [19] B. Zhao, L. Stankovic, and V. Stankovic, "On a training-less solution for non-intrusive appliance load monitoring using graph signal processing," *IEEE Access*, vol. 4, pp. 1784–1799, 2016.
- [20] Y. Zhou, Z. Shi, Z. Shi, Q. Gao, and L. Wu, "Disaggregating power consumption of commercial buildings based on the finite mixture model," *Appl. Energy*, vol. 243, pp. 35–46, Jun. 2019.
- [21] W. A. de Souza, F. D. Garcia, F. P. Marafão, L. C. P. da Silva, and M. G. Simões, "Load disaggregation using microscopic power features and pattern recognition," *Energies*, vol. 12, no. 14, p. 2641, Jul. 2019.
- [22] S. Singh and A. Majumdar, "Analysis co-sparse coding for energy disaggregation," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 462–470, Jan. 2019.
- [23] M. Dong, P. C. M. Meira, W. Xu, and F. Walimir, "An event window based load monitoring technique for smart meters," *IEEE Trans. Smart Grid*, vol. 3, no. 2, pp. 787–796, Jun. 2012.
- [24] Z. Wang and G. Zheng, "Residential appliances identification and monitoring by a nonintrusive method," *IEEE Trans. Smart Grids*, vol. 3, no. 1, pp. 80–92, Mar. 2012.
- [25] M. Zeifman, "Disaggregation of home energy display data using probabilistic approach," *IEEE Trans. Consum. Electron.*, vol. 58, no. 1, pp. 23–31, Feb. 2012.
- [26] R. Machlev, J. Belikov, Y. Beck, and Y. Levron, "MO-NILM: A multi-objective evolutionary algorithm for NILM classification," *Energy Buildings*, vol. 199, pp. 134–144, Sep. 2019.
- [27] R. Machlev, Y. Levron, and Y. Beck, "Modified cross-entropy method for classification of events in NILM systems," *IEEE Trans. Smart Grid*, vol. 10, no. 5, pp. 4962–4973, Sep. 2019.
- [28] M. A. Mengistu, A. A. Girmay, C. Camarda, A. Acquaviva, and E. Patti, "A cloud-based on-line disaggregation algorithm for home appliance loads," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 3430–3439, May 2018.
- [29] A. Moradzadeh, O. Sadeghian, K. Pourhossein, B. Mohammadi-Ivatloo, and A. Anvari-Moghaddam, "Improving residential load disaggregation for sustainable development of energy via principal component analysis," *Sustainability*, vol. 12, no. 8, p. 3158, Apr. 2020.
- [30] L. Pereira and N. Nunes, "An empirical exploration of performance metrics for event detection algorithms in non-intrusive load monitoring," *Sustain. Cities Soc.*, vol. 62, Nov. 2020, Art. no. 102399.
- [31] D. Hua, F. Huang, L. Wang, and W. Chen, "Simultaneous disaggregation of multiple appliances based on non-intrusive load monitoring," *Electr. Power Syst. Res.*, vol. 193, Apr. 2021, Art. no. 106887.
- [32] D. Baptista, S. Mostafa, L. Pereira, L. Sousa, and F. Morgado-Dias, "Implementation strategy of convolution neural networks on field programmable gate arrays for appliance classification using the voltage and current (VI) trajectory," *Energies*, vol. 11, no. 2460, pp. 1–18, 2018.
- [33] L. De Baets, J. Ruyssinck, C. Develder, T. Dhaene, and D. Deschrijver, "Appliance classification using VI trajectories and convolutional neural networks," *Energy Buildings*, vol. 158, pp. 32–36, Jan. 2018.
- [34] D. L. Cavalca and R. A. S. Fernandes, "Recurrence plots and convolutional neural networks applied to nonintrusive load monitoring," in *Proc. IEEE Power Energy Soc. Gen. Meeting (PESGM)*, Aug. 2020, pp. 1–5.
- [35] J. Z. Kolter and M. J. Johnson, "REDD: A public data set for energy disaggregation research," in *Proc. Workshop Data Mining Appl. Sustainability (SIGKDD)*, San Diego, CA, USA, vol. 25, 2011, pp. 59–62.
- [36] M. J. Johnson and A. S. Willsky, "Bayesian nonparametric hidden semi-Markov models," *J. Mach. Learn. Res.*, vol. 14, no. 1, pp. 673–701, 2013.
- [37] S. Makonin, F. Popowich, I. V. Bajic, B. Gill, and L. Bartram, "Exploiting HMM sparsity to perform online real-time nonintrusive load monitoring," *IEEE Trans. Smart Grid*, vol. 7, no. 6, pp. 2575–2585, Nov. 2016.
- [38] F. Luo, G. Ranzi, W. Kong, Z. Y. Dong, S. Wang, and J. Zhao, "Non-intrusive energy saving appliance recommender system for smart grid residential users," *IET Gener., Transmiss. Distrib.*, vol. 11, no. 7, pp. 1786–1793, May 2017.
- [39] Z. Xu, W. Chen, and Q. Wang, "A new non-intrusive load monitoring algorithm based on event matching," *IEEE Access*, vol. 7, pp. 55966–55973, 2019.
- [40] S. M. Tabatabaei, S. Dick, and W. Xu, "Toward non-intrusive load monitoring via multi-label classification," *IEEE Trans. Smart Grid*, vol. 8, no. 1, pp. 26–40, Jan. 2017.
- [41] J. Kim, T.-T.-H. Le, and H. Kim, "Nonintrusive load monitoring based on advanced deep learning and novel signature," *Comput. Intell. Neurosci.*, vol. 2017, pp. 1–22, Oct. 2017.
- [42] M. Gaur and A. Majumdar, "Disaggregating transform learning for non-intrusive load monitoring," *IEEE Access*, vol. 6, pp. 46256–46265, 2018.
- [43] S. Singh and A. Majumdar, "Deep sparse coding for non-intrusive load monitoring," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 4669–4678, Sep. 2017.
- [44] D. Li and S. Dick, "Residential household non-intrusive load monitoring via graph-based multi-label semi-supervised learning," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 4615–4627, Jul. 2019.
- [45] W. Kong, Z. Y. Dong, J. Ma, D. J. Hill, J. Zhao, and F. Luo, "An extensible approach for non-intrusive load disaggregation with smart meter data," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3362–3372, Jul. 2018.
- [46] P. beim Graben, K. K. Sellers, F. Fröhlich, and A. Hutt, "Optimal estimation of recurrence structures from time series," *EPL Europhys. Lett.*, vol. 114, no. 3, p. 38003, May 2016.
- [47] J.-P. Eckmann, S. O. Kamphorst, and D. Ruelle, "Recurrence plots of dynamical systems," *Europhys. Lett.*, vol. 4, no. 9, pp. 973–977, Nov. 1987.
- [48] N. Marwan, M. C. Romano, M. Thiel, and J. Kurths, "Recurrence plots for the analysis of complex systems," *Phys. Rep.*, vol. 438, nos. 5–7, pp. 237–329, 2007.
- [49] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [50] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–14.



DIEGO L. CAVALCA was born in Pirajui, Brazil, in 1988. He is a technologist in system analysis from the Centro Universitário de Lins—UNILINS, Lins, Brazil, in 2013, and the M.Sc. degree in computer science from the Federal University of São Carlos (UFSCar), São Carlos, Brazil, in 2018. He is currently pursuing the Ph.D. degree in computer science with UFSCar. His research interests include deep learning and optimization algorithms.



RICARDO A. S. FERNANDES (Member, IEEE) was born in Barretos, Brazil, in 1984. He received the B.Sc. degree in electrical engineering from the Educational Foundation of Barretos, Barretos, in 2006, and the M.Sc. and Ph.D. degrees in electrical engineering from the University of São Paulo, São Carlos, Brazil, in 2009 and 2011, respectively. In 2015 and 2017, he was a Visiting Professor with the Polytechnic Institute of Porto. He is currently an Associate Professor with the Federal University of São Carlos. His research interests include smart grids, smart homes and buildings, power quality, and machine learning.