

Article

Power Quality Transient Detection and Characterization Using Deep Learning Techniques

Nuno M. Rodrigues ^{1,†} , Fernando M. Janeiro ^{2,†}  and Pedro M. Ramos ^{1,*,†} ¹ Instituto de Telecomunicações, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal² Instituto de Telecomunicações, Universidade de Évora, 7000-671 Évora, Portugal

* Correspondence: pedro.m.ramos@tecnico.ulisboa.pt; Tel.: +351-218-418-485

† These authors contributed equally to this work.

Abstract: Power quality issues can affect the performance of devices powered by the grid and can, in severe cases, permanently damage connected devices. Events that affect power quality include sags, swells, waveform distortions and transients. Transients are one of the most common power quality disturbances and are caused by lightning strikes or switching activities among power-grid-connected systems and devices. Transients can reach very high magnitudes, and their duration spans from nanoseconds to milliseconds. This study proposed a deep-learning-based technique that was supported by convolutional neural networks and a bidirectional long short-term memory approach in order to detect and characterize power-quality transients. The method was validated (i.e., benchmarked) using an alternative algorithm that had been previously validated according to a digital high-pass filter and a morphological closing operation. The training and performance assessments were carried out using actual power-grid-measured data and events.

Keywords: power quality monitoring; power grid measurements; transient detection; transient characterization; deep learning; convolutional neural networks



Citation: Rodrigues, N.M.; Janeiro, F.M.; Ramos, P.M. Power Quality Transient Detection and Characterization Using Deep Learning Techniques. *Energies* **2023**, *16*, 1915. <https://doi.org/10.3390/en16041915>

Academic Editor: Miguel Jiménez Carrizosa

Received: 29 December 2022

Revised: 6 February 2023

Accepted: 13 February 2023

Published: 15 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Power quality (PQ) monitoring consists of measuring and detecting events that disturb the grid-supplied waveform and may affect the normal behavior of electrical equipment. Power quality events include overvoltages, undervoltages, sags, swells, waveform distortions, flickers and transients [1]. In particular, transients occur more frequently than other events and are short-term disturbances with amplitudes starting at 5–10 % of the rated voltage. Electrical equipment are designed to resist PQ events up to a certain level. If the PQ events exceed certain thresholds, either in duration or amplitude, permanent damage may occur. The acceptable limits are defined by the IEEE and IEC international standards, such as IEEE 1159-2019 [1] and IEC 61000-2-2 [2]. Although measurement and event detection methods have been proposed by these international standards, PQ remains an active research topic, especially in the area of signal processing, with the development and proposal of new algorithms and techniques [3–5].

Artificial neural networks (ANNs) are inspired by the brain network of neurons. The inputs of each neuron are the weighted combinations of the outputs of other neurons, together with a bias component. This combination is applied to an activation function, which provides the non-linearity required by these methods in order to determine the neuron output [6]. ANNs have been used in diverse areas of science, usually for pattern recognition [7] but also in measurements [8] and non-destructive testing [9]. An ANN is trained by examples, using a back-propagation algorithm [6] until it reaches a suitable accuracy. Recent advances in computing power, availability of data and advances on network topologies have led to the development of deep learning (DL) [10]. The main advantages of DL include the ability to use raw data, minimal human intervention and

the capacity to search for deeply embedded patterns in the training data. This concept has enabled breakthrough results in many different areas of science and technology, including self-driving cars [11], facial recognition [12], and media content suggestions [13], among many others. The use of DL for power quality event detection and characterization is no exception, with some recent results presented in the literature [14–17]. However, these have been limited only to training sets with simulated/synthetic waveforms and without any real measured data, which cannot capture the full scale of imperfections that characterize real grid-acquired waveforms.

The most frequent PQ events are transients, and many different methods have been used to detect and their amplitude and duration. These include, for example, wavelets [18], time-frequency [19] and morphology methods [20]. The latter can be complex, involving filtering that is followed by dilation and erosion operations, according to a suitable threshold to discard the low amplitude transients. This study proposed the use of deep learning to detect and estimate the amplitude and duration of transients in the electrical grid. To achieve this goal, a two-stage approach was proposed, in which a convolutional neural network (CNN) was used to detect the presence of transients, which fed a second network composed of a convolutional neural network and a bidirectional long short-term memory (BiLSTM) to estimate the transient duration and amplitude. The two networks were trained and validated with actual power grid events that were stored in a publicly available database [21]. The events stored in the database had been detected and characterized by the transient detection algorithm presented in [20]. Therefore, this paper presents the results of the development of a deep learning network, trained and validated with real acquired signals, for the detection and characterization of PQ transient events.

This paper is divided into four sections, including the Introduction and the Conclusions. Section 2 presents a characterization of the measured transients that were used to train, validate and test the proposed DL neural network. Section 3 describes the DL network used for transient detection and the subsequent network that estimated the transient amplitude and duration as well as the main results of this paper.

2. Power Quality Transient Characterization

In Figure 1, four examples of the acquired power grid voltage transients with different characteristics are presented. The amplitudes are shown in per unit (pu) values, which are typical in power quality, (e.g., [1]), to normalize the amplitudes using, in this case, the power grid voltage nominal RMS voltage, which was 230 V in Portugal. The power grid voltages were continuously acquired at 12.5 kS/s using an LEM Hall Effect sensor (LEM LV 25-P) and a 16-bit data acquisition board (DAQ) from National Instruments—NI-USB-9215. The segments were 50 ms long, which corresponded to 2.5 nominal periods at 50 Hz, with 625 samples for each segment.

Figure 1 shows the transients at different power grid voltage phases with different amplitudes, phase locations and durations. Figure 1a shows an oscillatory transient while b and c show negative transients, and the waveform in d includes a positive transient [1]. Notice that other power quality events were not considered here, as the focus was centered only on transient detection and characterization. The represented data showed, in addition to the transient, some harmonic distortion typical in low voltage (LV) power grid voltages.

Transient detection and characterization (e.g., amplitude and duration), can be achieved for example, using wavelet decomposition [22,23] or using a high-pass digital filter followed by a closing morphological operation. The high-pass filter and closing morphological operation method was presented in [24] and implemented in an embedded measurement system described in [20]. This method was used in this study as the reference method (i.e., benchmark) because it was also the process used to detect and measure the real transients in the event database [21] that was used.

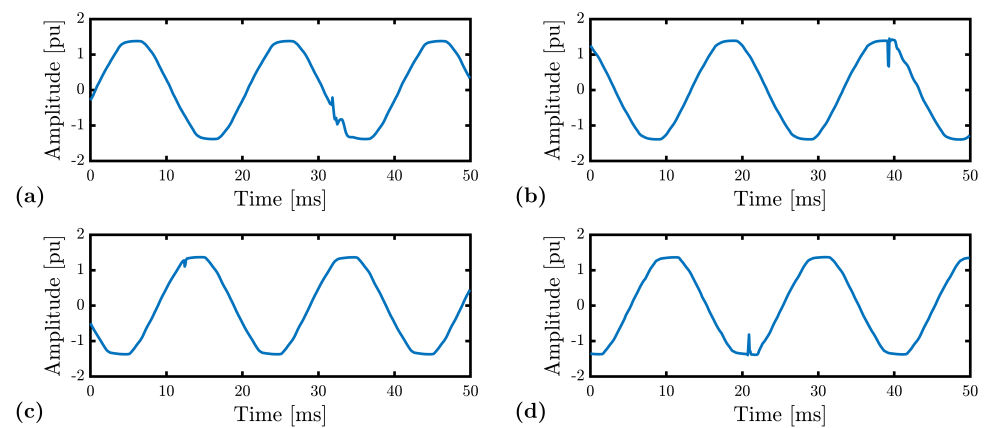


Figure 1. (a–d) Example of four acquired power grid voltage normalized segments. Each segment has 625 acquired samples with 16-bit DAQ at 12.5 kS/s, which corresponds to segments with 50 ms. The four examples highlight the difference in transient amplitudes, durations and relative phase locations.

Figure 2 represents an example of the high-pass filter and morphological algorithm operation used to detect the transients and estimate the transient characteristic parameters (i.e., amplitude and duration). Figure 2a shows an example of an oscillatory transient near 32 ms. The high-pass filter extracts the power grid voltage fundamental component, and the filter-output absolute normalized voltage is represented in Figure 2b. The closing morphological operation, which relied on the dilation operation followed by the erosion, grouped together consecutive oscillations as part of the same transient. Figure 2c shows the result of the morphology closing operation, where the multiple oscillations were joined together to define one unique transient event.

Transient detection was performed by a threshold analysis of the closing operation results (in this paper, a 0.1 pu threshold was considered). If the closing operation results exceeded the pre-defined threshold, a transient was considered detected. The transient amplitude was the maximum of the closing operation, and the transient duration was estimated by the time interval between the first and second interceptions of the closing operation results, with the threshold value using interpolation. For the example in Figure 2, the transient had a 0.119 pu amplitude and 0.72 ms duration. Notice that the harmonic distortions in Figure 2a were present at the filter output, as shown in Figure 2b, but they were not enough to trigger the transient detection process.

The complete dataset used in this work had 9768 transients, and the histogram distribution of the amplitude of these transients is shown in Figure 3. Notice that since the threshold transient detection was set at 0.1 pu, there were no transients with an amplitude below this value. The highest recorded transient in this dataset had a 1.997 pu amplitude.

Figure 4 presents the histogram of the transient durations for the complete dataset. Recall that the duration values were obtained by interpolation from the closing morphological operation interceptions, according to the 0.1 pu threshold, and have, therefore, better resolution than the inverse of the sampling rate (80 μ s). The maximum recorded transient duration was 6.6 ms. Notice that there was a significant amount of transients with duration values near 0.2 ms.

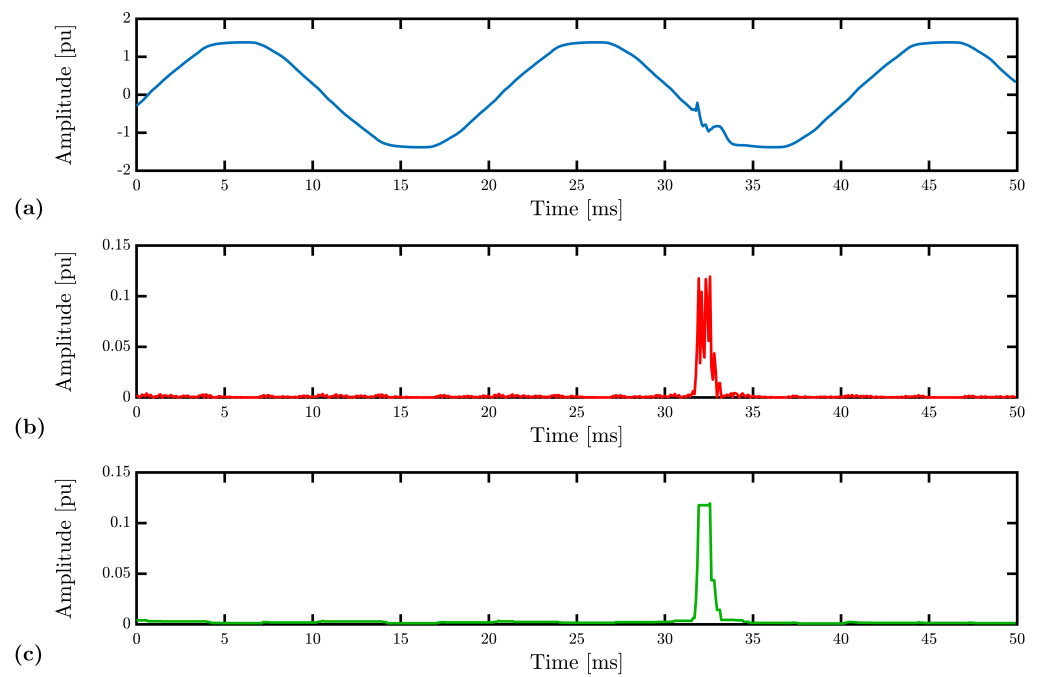


Figure 2. Example of transient detection and characterization using the high-pass filter and closing morphology operation. The input segment is represented in (a) while the absolute values of the filter output are presented in (b). The result of the closing morphological operation is depicted in (c).

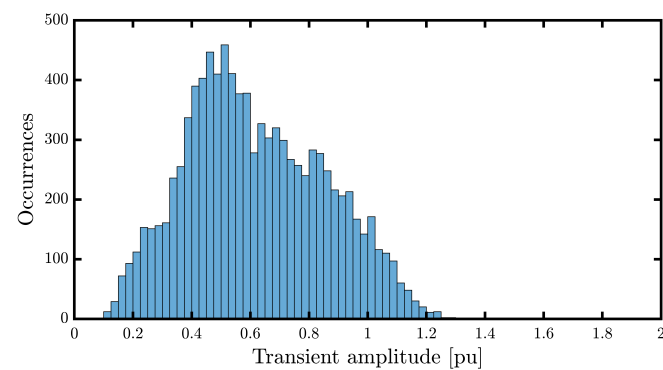


Figure 3. Histogram of the transient amplitudes in the 9768 database segments. Notice that the original transient detection used to build the database ignored events with transient amplitudes below 0.1 pu, and, therefore, there are no events below that amplitude.

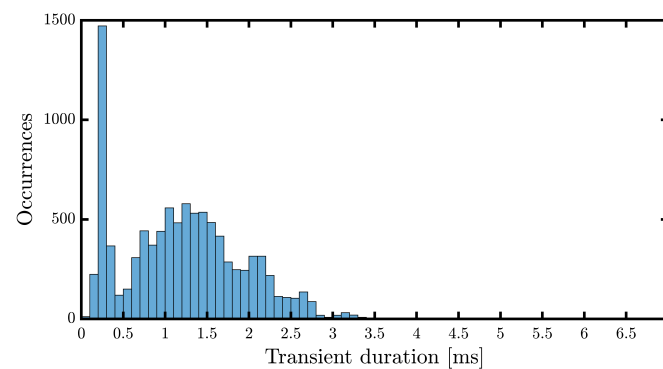


Figure 4. Transient duration histogram of the recorded transient database. The duration was obtained by interpolation of the instances where the closing operation crossed the threshold value.

Finally, Figure 5 depicts the phase location histogram of the recorded transients. Notice that the phase location was not uniform, with more transients occurring near or around 180° .

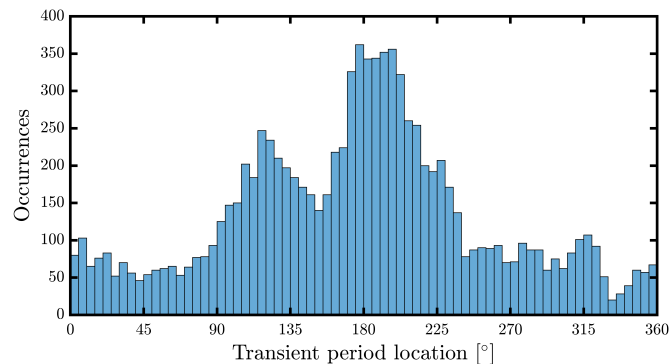


Figure 5. Histogram of the phase location of the transients included in the measured database.

3. Deep Learning for PQ Transient Detection and Characterization

While there are many different architectures for deep-learning networks, the convolutional neural network (CNN) is especially suited for image analysis and classification. These networks have the ability to extract hidden patterns in the provided information by training their internal filters. Therefore, since the problem under study corresponded to the detection and characterization of transients in acquired data segments, the proposed algorithm was based on CNNs. The theory of CNNs is beyond the scope of this work, and a good survey of their theory and applications can be found in [25].

The proposed deep-learning architecture for transient detection and characterization is depicted in Figure 6. It was a two-step process, where the purpose of the first step was to detect the presence of a transient while the second step was only used when a transient was detected, and its objective was to estimate the amplitude and duration of the transient.

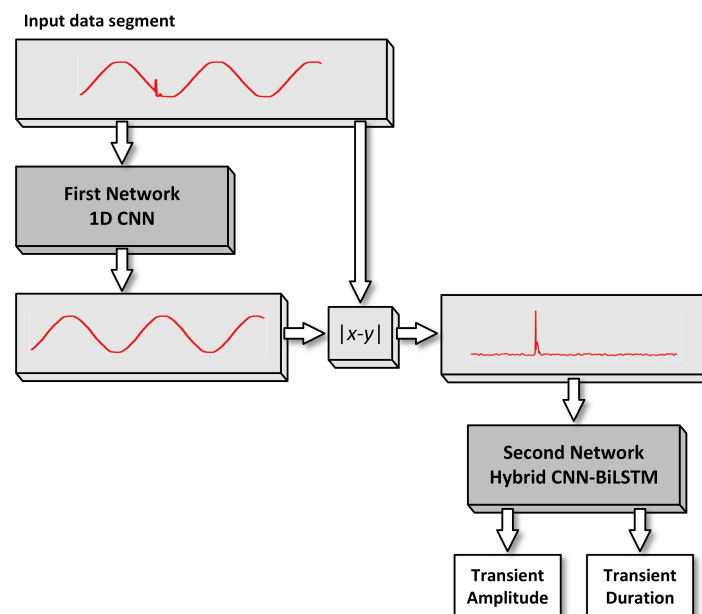


Figure 6. Topology of the proposed deep-learning architecture for the detection and characterization of power quality transients.

The objective of the first step was to extract the input voltage waveforms without transients from the 625-sample input time series. This step was obtained with a 1D CNN, as described in Section 3.1. Power quality transients appeared as spikes in the absolute

difference between the input time series and the extracted waveform. Transients were detected by comparison of the absolute difference with a pre-defined threshold.

When a transient was detected, the objective of the second step was to extract information that could be used to estimate the transient amplitude and duration. The network used in the second step was a hybrid CNN-BiLSTM, as described in Section 3.2.

3.1. Transient Detection

The objective of the first network was to obtain the waveform without the transient from the 625-sample acquired and amplitude normalized time series. The topology used in this step is depicted in Figure 7 and includes four layers.

This network was a 1D CNN where the first layer was a convolutional layer, which contains multiple filters with fixed lengths, and its objective is to extract different levels of information from the input time series. Each filter had different coefficients, and their numbers depended on the filter length. The output of each filter was the convolution of the filter with the input time series, along with a stride that was set for the whole layer. Since the objective of this step was to obtain a time series with a length that was the same as the input time series (i.e., 625 samples in this particular application), the stride was set to 1, and padding had to be used to ensure the filter outputs also had the same length. The convolutional layer had 200 filters with lengths of 240 and 200 bias parameters (one for each filter), which resulted in a total of 48 200 optimization parameters. The output of the convolutional layer were 200 time series (the convolution filter outputs), each with 625 samples.

The next three layers were fully connected layers (FCNs), in which the first received the 200×625 data from the convolutional layer and output 100×625 , for a total of 20 100 optimization parameters. The second FCN received the 100×625 from the first FCN and output 100×625 , for a total of 10 100 optimization parameters. The third and final FCN received the 100×625 from the second FCN and output the final 625 time series, with 101 optimization parameters. Overall, the 1D CNN of this first step had 78 501 parameters that were then optimized during the back-propagation training process.

In DL applications, the hyperparameters that define each layer are defined before the training process. If the desired goal is not achieved, these hyperparameters are adjusted, typically to increase the number of degrees of freedom. The hyperparameters were adjusted using a random search [26] while avoiding any overfitting of the training data. Although other methods for hyperparameter-tuning are available (e.g., grid search or Bayesian search), the use of random search was simple and fast, and it achieved good training results. In this 1D CNN network, the hyperparameters included the number of filters; their size and stride for the convolutional layer; and the number of fully connected layers as well as their distribution to obtain the 625-sample output waveforms without transients.

The training process consisted of inputting the time-series segments of the acquired power grid voltage without PQ events but with the normal voltage grid distortions and noise. Then, we trained the network to estimate the same time series. Back-propagation used the differences between the input time series and the network-estimated time series to adjust the set of optimization parameters of the network, so that the network output was similar to the input (only in the training, validation and test steps, as these datasets did not contain transients). Once the network was trained, whenever a time series with a transient was inserted, the network should output the time series without the event. The difference between the input and output time series indicated if a transient had been detected and included information about the transient for its characterization, namely, the transient amplitude and duration, which was addressed in the second step, as depicted in Figure 6.

For the training process of the 1D CNN, a dataset of 5585 time series, with 625 samples per segment without transients, was used. The validation used a different dataset with 621 segments while the test set had 470 segments. Figure 8 shows four examples of the trained network inputs and outputs, where the blue line corresponds to the network time series input (on the left-side vertical axis) and the red line is the difference between the

input and output of the network (right-side vertical axis). In this figure, only one period of the nominal power grid voltage (20 ms) is shown to improve the transient and spike visibility. The example in Figure 8a corresponded to a time series without a transient, and the difference showed the absence of any significant spikes. The difference was always below 0.05 pu, and the harmonic distortions were not registered. The examples in the remaining subplots of Figure 8 correspond to the input time series with transients, and the corresponding spikes are clearly visible and aligned with the time series transients. The situation depicted in Figure 8c had the largest represented amplitude transient, which exceeded 0.5 pu near 47 ms.

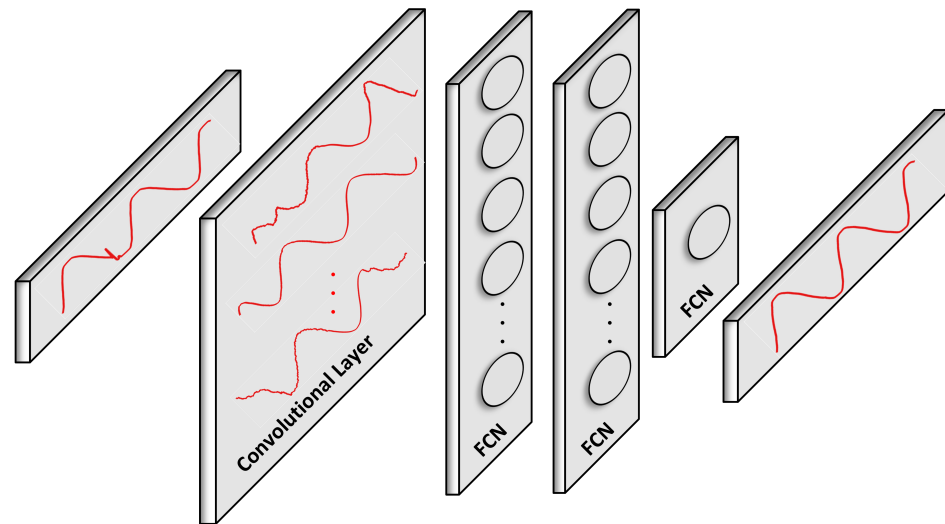


Figure 7. Internal topology of the 1D CNN that aims to extract the power grid voltage waveform without transients. This was then used to detect if a transient occurred by subtracting it from the input time series. The second network used the absolute difference to estimate the transient amplitude and duration.

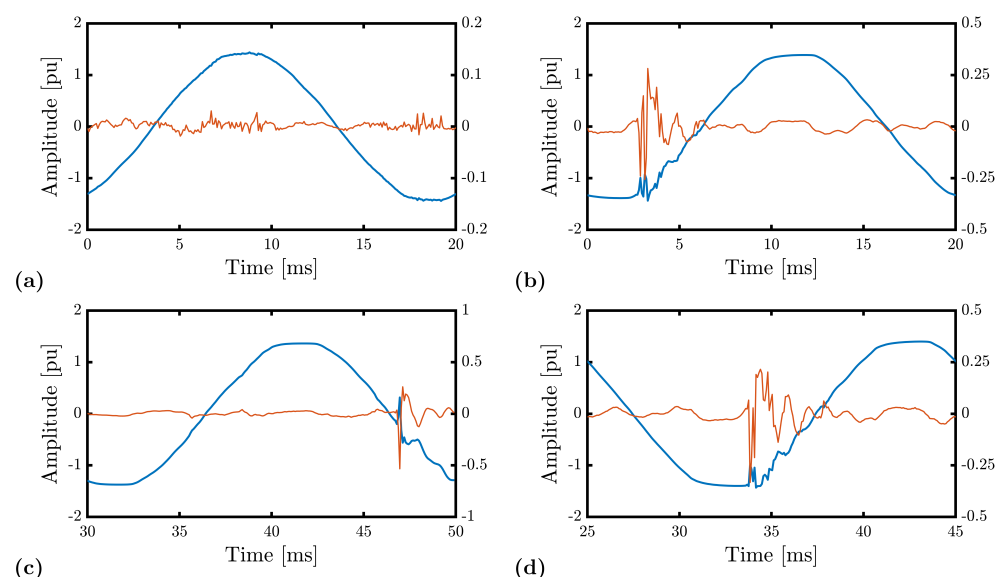


Figure 8. Examples of the operation of the 1D CNN: (b–d) depict transients while (a) is a transient-free segment. The blue lines depict the input time series (left vertical axis) while the red lines represent the difference between the input and output time series (right vertical axis).

3.2. Transient Parameter Estimation

The second step of the proposed method began with the absolute difference between the acquired segment and the output segment from the previous step, as shown in Figure 6,

and its objective was to estimate the transient amplitude ($\widehat{A}_i$) and duration ($\widehat{D}_i$). The learning process used the transient amplitude (A_i) and duration (D_i) values obtained by the high-pass filter and morphology operation, as described in Section 2. Training was performed using the mean-squared error (MSE) weighted by the average values of the transient amplitudes ($\bar{A}$) and transient durations ($\bar{D}$)

$$\text{MSE}_{\text{weighted}} = \frac{1}{N} \left[\sum_{i=1}^N \left(\frac{\widehat{A}_i - A_i}{\bar{A}} \right)^2 + \sum_{i=1}^N \left(\frac{\widehat{D}_i - D_i}{\bar{D}} \right)^2 \right] \quad (1)$$

where N is the number of data segments used. This weighted MSE was necessary because the two estimated parameters had significantly different orders of magnitude (i.e., the average transient amplitude was $\bar{A} = 0.347$ pu, and the average transient duration was $\bar{D} = 0.702$ ms). If the same weight were used, the training process would tend to give more relevancy (i.e., weight) to the highest magnitude parameter (in this case, the transient amplitude). In this situation, the lowest magnitude parameter estimation, the transient duration, would become irrelevant in the optimization process, which would result in poor transient duration estimation.

The proposed network topology for this second stage, represented in Figure 9, was a hybrid CNN–BiLSTM. It began with a convolutional layer with 64 filters, with a size of 15, a stride of 1 and no padding, as in this case, there was no need to maintain the size of the time series. The output of this layer was 64×611 with 15×64 weights and 64 biases, for a total of 1024 optimization parameters. The output of the convolutional layer was fed into a bidirectional long short-term memory (BiLSTM) layer with 100 hidden units. The BiLSTM was used because it could learn from the dependencies of the time-series sequence data. The bidirectional version considered the dependencies in both directions by adding one extra LSTM layer with the input data in the reverse order. This BiLSTM layer had 64×800 input weights, 800×100 recurrent weights and 800 output biases, for a total of 132 000 optimization parameters, and it output 200 values. The final layer was a fully connected neural (FCN) network, which output the two estimated parameters (i.e., transient amplitude and duration). This layer had 2×200 weights and 2 biases, for a total of 402 parameters, and these were then optimized during the learning process using the cost function $\text{MSE}_{\text{weighted}}$ from (1). The complete network had a total of 133 426 parameters that were optimized during the back-propagation-based training process.

This network was trained with 7326 acquired data segments with real measured transients that had been pre-processed by the first network. For validation, a set with 1466 similar data segments was used. The testing stage was conducted with a different set with 976 segments.

The results obtained from the test set are shown in Figure 10. Figure 10a depicts the estimated transient amplitudes as a function of the actual transient amplitudes (i.e., benchmarks) while Figure 10b is the corresponding representation for the transient duration. In Figure 10a, the 0.1 pu lower limit of the transient amplitude of the used segments is clearly visible. Figure 10c,d show the estimated transient amplitude and duration error histograms. The proposed method was better at estimating the transient amplitude than the transient duration. This was a reflection of the complexity in estimating the transient duration, which was tied to the threshold that had been used in the baseline algorithm (i.e., high-pass filter and morphological operation) and had never been explicitly used in the DL networks. Specifically, from the absolute difference between the input segment and the clear segment estimated by the first 1D CNN, this two-step hybrid CNN–BiLSTM network must extract the transient duration using only the target values input during the training process as a reference.

The root-mean-square error (RMSE) of the transient amplitude, as obtained using the benchmark values from the filter and closing morphology algorithm, was 0.0197 pu, and the average error was -0.00037 pu. As for the transient duration, the RMSE was 0.2024 ms, and the average error was 0.00039 ms. The estimated amplitude error interval

$[-0.045; 0.034]$ pu was based on 95 % of the situations shown in Figure 10. For the estimated duration error, the corresponding 95 % interval was $[-0.45; 0.33]$ ms. These results again highlighted that the proposed method could achieve better results when estimating the transient amplitude, as compared to the estimated transient duration.

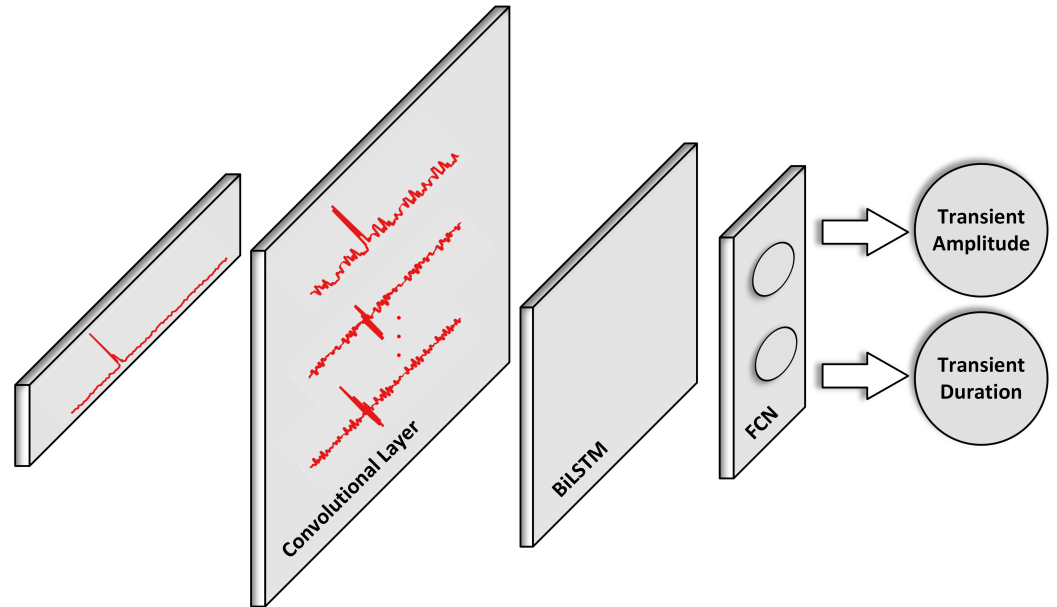


Figure 9. Hybrid CNN–BiLSTM topology for the second step, which receives the absolute difference from the output of the first 1D CNN as well as the acquired data and estimates the transient duration and amplitude.

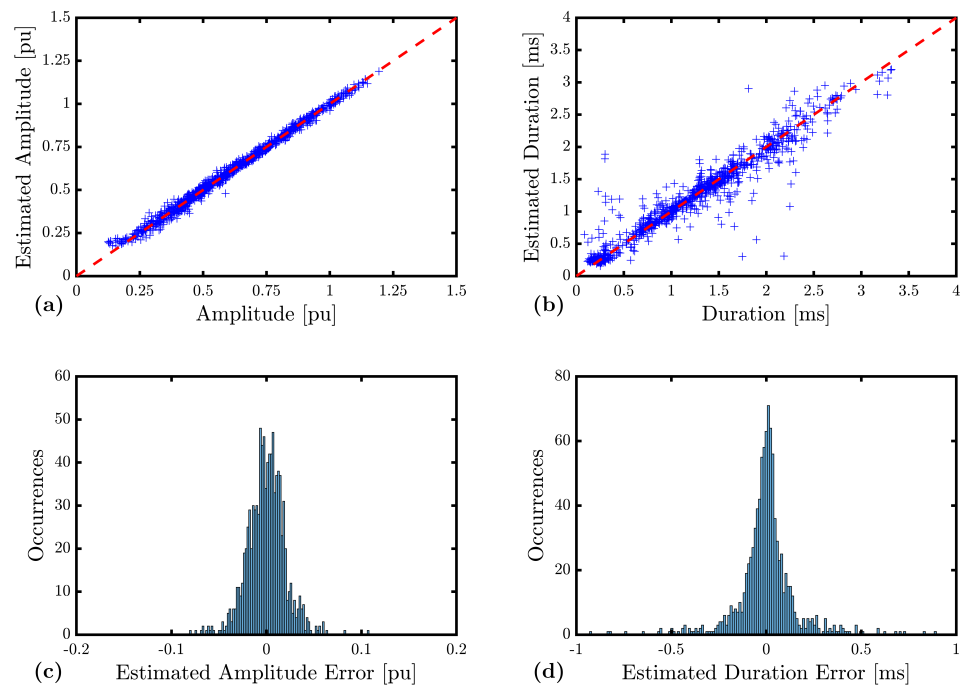


Figure 10. Test set results. In (a), the relation between the real transient amplitude (i.e., benchmark) and the proposed method’s estimated amplitude is shown while in (b), the corresponding relation for the transient duration is plotted. The histogram of the estimated transient amplitude error is shown in (c), and the estimated transient duration error is shown in (d).

4. Conclusions

This paper addressed the detection and characterization of power quality transients, which are the most common type of disturbances in the power grid. Although widely studied with different approaches, as described in the scientific literature, they usually require technical knowledge and advanced finely tuned signal processing algorithms, such as filtering, morphology operations and wavelets, among others. Previous studies available in the literature have applied deep learning methods to evaluate power quality but only with simulated/synthetic waveforms, which do not include the diverse characteristics of actual grid waveforms. The approach described in this paper used only real measured data to study the performance of deep learning on real grid transients. The strategy used for the deep-learning network divided the problem into two steps: the detection of transients and the characterization of amplitude and duration. A 1D CNN was optimized to extract the voltage waveform without transients, which was not a pure sinewave due to the presence of harmonics and waveform distortions. The second network, a hybrid CNN–BiLSTM, received (as input) the absolute difference between the original acquired signal and the output of the first network, as shown in Figure 6. This network was used to estimate transient parameters and was composed of a convolutional layer, followed by a bidirectional long short-term memory layer, and then completed with a fully connected layer.

The proposed architecture was trained with real acquired data, with and without transients, and the DL test step resulted in a transient RMSE amplitude of 0.0197 pu and 0.024 ms for the transient duration. This deep-learning approach performed better when estimating the amplitude of the transient, as compared to its duration estimation, as shown in Figure 10. These results validated the use of deep-learning algorithms for PQ transient detection and characterization, using real measured data for training and validation.

Future work will include the implementation of the trained networks in an embedded measurement system (e.g., in an STM32Cube.AI) for the continuous detection and characterization of real-time power grid transient events. The assessment of the performance of the measurement system will be compared with previously developed embedded measurement systems that use a high-pass filter and morphological operations.

Author Contributions: Conceptualization and methodology, N.M.R., F.M.J. and P.M.R.; software, N.M.R.; writing, review and editing, N.M.R., F.M.J. and P.M.R. All authors have read and agreed to the published version of the manuscript.

Funding: This work was developed under the PhD program of the Fundação para a Ciência e a Tecnologia (FCT) reference SFRH/BD/130327/2017 and is funded by FCT/MCTES through national funds and, when applicable, co-funded EU funds, under the project UIDB/EEA/50008/2020.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. *IEEE 1159-2019*; (Revision of IEEE Std 1159-2009). IEEE International Standard—IEEE Recommended Practice for Monitoring Electric Power Quality. IEEE: New York, NY, USA, 2019. [\[CrossRef\]](#)
2. *IEC 61000-2-2*; IEC International Standard—Electromagnetic Compatibility (EMC)—Environment—Compatibility Levels for Low-Frequency Conducted Disturbances and Signaling in Public Low-Voltage Power Supply Systems, 2nd ed. IEC: Geneva, Switzerland, 2018.
3. Rodrigues, N.; Janeiro, F.; Ramos, P. Implementation of Goertzel-based frequency estimation for power quality monitoring in embedded measurement systems. *Metrol. Meas. Syst.* **2022**, *29*, 455–468. [\[CrossRef\]](#)
4. Bandla, P.; Vairavasundaram, I.; Teekaraman, Y.; Kuppusamy, R.; Nikolovski, S. Real time sustainable power quality analysis of non-linear load under symmetrical conditions. *Energies* **2022**, *29*, 57. [\[CrossRef\]](#)
5. Han, Y.; Feng, Y.; Yang, P.; Xu, L.; Xu, Y.; Blaabjerg, F. Cause, classification of voltage sag, and voltage sag emulators and applications: A comprehensive overview. *IEEE Access* **2020**, *8*, 1922–1934. [\[CrossRef\]](#)
6. Haykin, S. *Neural Networks and Learning Machines*, 3rd ed.; Prentice-Hall: Upper Saddle, NJ, USA, 2009.

7. Abiodun, O.; Jantan, A.; Omolara, A.E.; Dada, K.V.; Umar, A.M.; Linus, O.U.; Arshad, H.; Kazaure, A.A.; Gana, U.; Kiru, M.U. Comprehensive review of artificial neural network applications to pattern recognition. *IEEE Access* **2019**, *7*, 158820–158846. [CrossRef]
8. Daponte, P.; Grimaldi, D. Artificial neural networks in measurements. *Measurement* **1998**, *23*, 93–115. [CrossRef]
9. Rosado, L.; Janeiro, F.; Ramos, P.; Piedade, M. Defect characterization with eddy current testing using nonlinear-regression feature extraction and artificial neural networks. *IEEE Trans. Instrumen. Meas.* **2013**, *62*, 1207–1214. [CrossRef]
10. Pouyanfar, S.; Sadiq, S.; Yan, Y.; Tian, H.; Tao, Y.; Reyes, M.; Shyu, M.-L.; Chen, S.-C.; Iyengar, S. A survey on deep learning: Algorithms, techniques, and application. *ACM Comput. Surv.* **2018**, *51*, 1–36. [CrossRef]
11. Tabernik, D.; Skočaj, D. Deep learning for large-scale traffic-sign detection and recognition. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1427–1440. [CrossRef]
12. Guo, G.; Zhang, N. A survey on deep learning based face recognition. *Comput. Vis. Image Underst.* **2019**, *189*, 102805. [CrossRef]
13. Nguyen, H.; Wistuba, M.; Grabocka, J.; Drumond, L.; Schmidt-Thieme, L. Personalized deep learning for tag recommendation. *Lect. Notes Comput. Sci.* **2017**, *10234*, 186–197. [CrossRef]
14. Wang, S.; Chen, H. A novel deep learning method for the classification of power quality disturbances using deep convolutional neural network. *Appl. Energy* **2019**, *235*, 1126–1140. [CrossRef]
15. Ma, J.; Zhang, J.; Xiao, L.; Chen, K.; Wu, J. Classification of Power Quality Disturbances via Deep Learning. *IETE Tech. Rev.* **2017**, *34*, 408–415. [CrossRef]
16. Oliveira, R.; Bollen, M. Deep learning for power quality. *Electr. Power Syst. Res.* **2023**, *214*, 108887. [CrossRef]
17. Salles, R.; Ribeiro, P. The use of deep learning and 2-D wavelet scalograms for power quality disturbances classification. *Electr. Power Syst. Res.* **2023**, *214*, 108834. [CrossRef]
18. Gaing, Z.-L. Wavelet-based neural network for power quality disturbance recognition and classification. *IEEE Trans. Power Deliv.* **2004**, *19*, 1560–1568. [CrossRef]
19. Wang, M.; Rowe, G.; Manishev, A. Classification of power quality events using optimal time-frequency representations, theory and application. *IEEE Trans. Power Deliv.* **2004**, *19*, 1496–1503. [CrossRef]
20. Radil, T.; Ramos, P.; Janeiro, F.; Serra, A. PQ monitoring system for real-time detection and classification of disturbances in a single-phase power system. *IEEE Trans. Instrumen. Meas.* **2008**, *57*, 1725–1733. [CrossRef]
21. IT-GIM Group. Smart-Monitor Power Quality Events Database. Available online: <http://smartmonitor.lx.it.pt/> (accessed on 30 January 2023).
22. Santoso, S.; Powers, E.J.; Grady, W.M.; Hofmann, P. Power quality assessment via wavelet transform analysis. *IEEE Trans. Power Deliv.* **1996**, *11*, 924–930. [CrossRef]
23. Gaouda, A.M.; Salama, M.M.A.; Sultan, M.R.; Chikhani, A.Y. Power quality detection and classification using wavelet-multiresolution signal decomposition. *IEEE Trans. Power Deliv.* **1999**, *1476*, 1469–1476. [CrossRef]
24. Radil, T.; Ramos, P.; Janeiro, F.; Serra, A. An efficient approach to detect and classify power quality disturbances. *COMPEL-Int. J. Comp. Math. Electr. Electron. Eng.* **2008**, *27*, 1178–1191. [CrossRef]
25. Kiranyaz, S.; Avci, O.; Abdeljaber, O.; Ince, T.; Gabbouj, M.; Inman, D. 1D convolutional neural networks and applications: A survey. *Mech. Syst. Signal Proc.* **2021**, *151*, 107398. [CrossRef]
26. Feurer, M.; Hutter, F. Hyperparameter Optimization. In *Automated Machine Learning*; Feurer, M., Hutter, F., Kotthoff, L., Vanschoren, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2019; pp. 3–33. [CrossRef]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.