





Article

Deep Learning for High-Impedance Fault Detection: Convolutional Autoencoders

Khushwant Rai , Farnam Hojatpanah , Firouz Badrkhani Ajai  and Katarina Grolinger * 

Department of Electrical and Computer Engineering, The University of Western Ontario, London, ON N6A 5B9, Canada; kkhushwa@uwo.ca (K.R.); fhajatpa@uwo.ca (F.H.); fajaei@uwo.ca (F.B.A.)

* Correspondence: kgroling@uwo.ca; Tel.: +1-519-661-2111 (ext. 81407)

Abstract: High-impedance faults (HIF) are difficult to detect because of their low current amplitude and highly diverse characteristics. In recent years, machine learning (ML) has been gaining popularity in HIF detection because ML techniques learn patterns from data and successfully detect HIFs. However, as these methods are based on supervised learning, they fail to reliably detect any scenario, fault or non-fault, not present in the training data. Consequently, this paper takes advantage of unsupervised learning and proposes a convolutional autoencoder framework for HIF detection (CAE-HIFD). Contrary to the conventional autoencoders that learn from normal behavior, the convolutional autoencoder (CAE) in CAE-HIFD learns only from the HIF signals eliminating the need for presence of diverse non-HIF scenarios in the CAE training. CAE distinguishes HIFs from non-HIF operating conditions by employing cross-correlation. To discriminate HIFs from transient disturbances such as capacitor or load switching, CAE-HIFD uses kurtosis, a statistical measure of the probability distribution shape. The performance evaluation studies conducted using the IEEE 13-node test feeder indicate that the CAE-HIFD reliably detects HIFs, outperforms the state-of-the-art HIF detection techniques, and is robust against noise.



Citation: Rai, K.; Hojatpanah, F.; Badrkhani Ajai, F.; Grolinger, K. Deep Learning for High-Impedance Fault Detection: Convolutional Autoencoders. *Energies* **2021**, *14*, 3623. <https://doi.org/10.3390/en14123623>

Academic Editors: Djaffar Ould-Abdeslam, Igor Simone Stievano and Riccardo Trincherio

Received: 11 May 2021
Accepted: 16 June 2021
Published: 18 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 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/>).

Keywords: high-impedance fault; power system protection; unsupervised learning; deep learning; convolutional autoencoder; convolutional neural network

1. Introduction

High-impedance faults (HIF) typically occur when a live conductor contacts a highly resistive surface [1–4]. They commonly have characteristics such as asymmetry of the current waveform, randomness, and non-linearity of the voltage-current relationship. These characteristics are diverse and are affected by multiple factors including surface types and humidity conditions [4]. As a result, the HIF current magnitude typically ranges from 0 to 75 A [1]. Such low current magnitudes compared to normal load current levels together with a high diversity of characteristics and patterns make HIFs difficult to detect. Specifically, the conventional overcurrent relays fail to discriminate most HIFs from load unbalance [1,2,5]. Nevertheless, the arcing ignition caused by an HIF is a safety hazard [4,6]. Moreover, undetected HIFs have been reported to cause instability of renewable energy systems [7]. As a result, reliably detecting and clearing HIFs in a timely manner are crucial to ensure the safety of personnel and maintain the power system integrity [4–6].

Various HIF detection techniques have been proposed. The harmonic-based detection is one of the most common techniques [1,8–12]. It operates well when the measured signals contain a large amount of harmonics, but it loses sensitivity when the HIF is far from the relay [1]. An HIF detection scheme based on a power line communication system is proposed by Milioudis et al. [13]. This method is successful in detecting and locating HIFs, but requires costly communication systems and is not suitable for large and complex networks [14].

The discrete wavelet transformation-based HIF detection technique [15] operates by examining the measured signals in both the time and frequency domains. Although this

method reduces the HIF detection delay, its performance is highly dependent on the choice of the mother wavelet and is adversely affected by the presence of unbalanced loads and noise [16].

HIF detection by Kalman Filtering [17] is based on tracking randomness in the current waveform. Because the small HIF current does not cause significant variations in the current waveforms seen by the relay at the substation [18], this approach exhibits low sensitivity.

The approach based on mathematical morphology [2] detects close-in HIFs but is not sufficiently reliable in presence of non-linear loads and under remote HIFs [16]. Empirical mode decomposition (EMD) [19] and variational mode decomposition (VMD) [5,6] have been recently proposed for HIF detection. These techniques operate by extracting HIF-related features from the signals. However, the EMD-based method suffers from modal mixing and is sensitive to noise [20]. On the other hand, the VMD-based technique is more robust against noise, nevertheless, it requires a large number of decomposition modes, as well as a high sampling frequency to perform well, which results in a large computational burden. Moreover, the VMD technique utilized in [5] for HIFs detection cannot reliably detect HIFs when a large non-linear load is in operation.

In recent years, machine learning (ML) approaches have been gaining popularity. A support vector machine (SVM)-based model is used with features extracted by VMD for HIF detection in distribution lines incorporating distributed generators [6]. This method prevents false detection of switching events as HIFs; however, it is sensitive to noise for signal to noise ratios (SNR) below 30 dB. Moreover, the transformations associated with the VMD require high computational capability [4]. An artificial neural network (ANN) is combined with discrete wavelet transforms (DWT) for HIF detection in medium-voltage networks. The DWTs are responsible for extracting the relevant features from the current waveforms, whereas the ANN acts as a classifier using those extracted features [21]. Probabilistic neural networks and various fuzzy inference systems have also been combined with wavelet transform (WT) for HIF detection [22,23]. In another study [24], a modified Gabor WT is applied on the input signal to extract features in the form of two dimensional (2-D) scalograms. The classifiers such as adaptive neuro-fuzzy inference system and SVM are utilized along with DWT-based feature extraction to detect HIFs and to discriminate them from other transients in medium-voltage distribution systems [25]. The neural networks are sensitive to frequency changes, whereas the WT faces challenges in terms of selecting the mother wavelet. The decision tree algorithm with the fast Fourier transform is also an alternative approach, but it is sensitive to noise [26].

Recurrent neural network (RNN)-based long short term memory (LSTM) approach is used with features obtained by DWT analysis to detect the HIFs in the solar photovoltaic integrated power system [27]. However, this method detects HIFs with a success rate of only 92.42% and, moreover, this study uses DWT-based features which are sensitive to noise [16]. In another study, the HIFs were detected and classified with 2-D convolutional neural network (CNN) which employs supervised learning to learn from the extracted features [24]. This method was accurate in the presented experiments; however, as it is a supervised learning-based method, it may fail to reliably detect HIF and non-HIF scenarios that are not present in the training data.

All reviewed ML studies apply a supervised approaches to learn the mapping from the input to the output based on a limited set of HIF and non-HIF scenarios present in the training set. Here, the non-HIF scenario refers to normal steady-state operation and any transient response to non-HIF disturbance, such as capacitor switching and load variations. A supervised learning system may fail to reliably identify any fault scenario or any non-fault disturbance which is not present in the training set. However, there is a wide range of non-fault operating conditions and it is difficult to include them all in the training set. Therefore, a different way of training ML models is required. Moreover, the existing technical literature on ML-based HIF detection relies on features extracted by resource intensive signal processing and data transformation techniques [28].

Consequently, this paper proposes the convolutional autoencoder (CAE) framework for HIF detection (CAE-HIFD), which utilizes an unsupervised approach that learns solely from the fault data, thus avoiding the need to take into account all possible non-fault scenarios in the learning stage. The ability of the CAE to model the relationship between the data points constituting a signal enables the CAE-HIFD to learn complex HIF patterns. The CAE discriminates steady-state conditions from HIFs by identifying deviations from the learned HIF patterns using cross-correlation (CC). The security against false detection of non-HIF disturbances (e.g., capacitor and load switching) as HIFs is achieved through kurtosis analysis.

The performance of the proposed protection strategy is evaluated through comprehensive studies conducted on the IEEE 13-node test feeder taking into account various HIF conditions involving seven different fault surfaces, as well as diverse non-HIF scenarios. The results indicate that the proposed CAE-HIFD (i) reliably detects HIFs regardless of the type of the surface involved and the fault distance, (ii) accurately discriminates between HIFs, steady-state operating conditions, and non-HIF disturbances, (iii) achieves higher accuracy than the state-of-the-art HIF detection techniques, and (iv) is robust against noise.

The paper is organized as follows: Section 2 provides an overview of autoencoders, Section 3 presents the proposed CAE-HIFD, Section 4.5 evaluates the performance of the CAE-HIFD, and finally Section 5 concludes the paper.

2. Unsupervised Learning with Autoencoders

A supervised machine learning model learns the mapping function from the input to output (label) based on example input–output pairs provided in the training dataset. In contrast, an unsupervised learning model discovers patterns and learns from the input data on its own, without the need for labeled responses, which makes this approach a go-to solution when labels are not available. Autoencoders are trained through unsupervised learning where the model learns data encoding by reconstructing the input [29]. They are commonly used for dimensionality reduction [29,30], but the non-linear feature learning capability has made them also successful in denoising and anomaly detection [31–33].

As shown in Figure 1, a conventional autoencoder is a feed-forward neural network consisting of an input, an output, and one or more hidden layers. The encoder part of the autoencoder reduces dimensionality. The input x of dimension f is multiplied by the weights W and, together with the bias b , is passed through the activation function σ to produce representation z of dimension m , $m < f$ [29], as follows.

$$z = \sigma(Wx + b) \quad (1)$$

Next, the decoder attempts to reconstruct the input x from the encoded value z . The product of weights W' and z is added to biases c , and the resultant is utilized as an input to the activation function σ to generate the reconstructed signal y as follows:

$$y = \sigma(W'z + c) \quad (2)$$

Over a number of iterations (epochs), the autoencoder optimizes the weights and biases by minimizing an objective function, such as mean squared error (MSE) [29]:

$$MSE = \frac{1}{f} \sum_{i=1}^f (x_i - y_i)^2 \quad (3)$$

The described feed-forward autoencoder ignores the spatial structure present in data and, to address this issue, the CAE was introduced [34]. A CAE replaces the fully connected layers of the feed-forward autoencoder with convolutional and deconvolutional layers. The convolutional layer performs the convolution operation on local spatial regions of the input. The two-dimensional (2D) and one dimensional (1D) convolutions have led to significant advancements in image processing [35] and sensor data processing [36–38],

respectively. In this study, the CAE-HIFD employs a 1D-CAE for HIF detection. In addition to the convolutional layer, the encoder typically has a max-pooling layer(s) which perform(s) down-sampling to reduce dimensionality. The decoder in the CAE reconstructs the input with the help of transposed convolution or up-sampling layer(s).

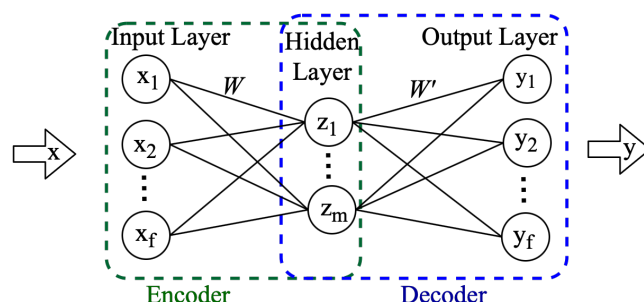


Figure 1. Conventional Autoencoder structure.

3. Convolutional Autoencoder for HIF Detection

Traditionally, for fault detection, autoencoders are trained with normal data and then used to detect abnormal operating conditions by identifying deviations from the learned normal data [32]. On the contrary, in the proposed CAE-HIFD, the CAE is trained using fault data only and recognizes non-fault operating conditions by detecting deviations from the learned fault scenarios. The spatial feature learning and generalization capability of the CAE assist the CAE-HIFD to detect new HIF scenarios that are not present in the training set. Furthermore, since the CAE is only trained on the fault data, any non-fault cases will not be identified as HIF, which increases the security of the proposed protection strategy.

As depicted in Figure 2, the CAE-HIFD is comprised of offline training and online HIF detection. The analog three-phase voltage and current signals are sampled and converted to digital signals using A/D converters. Training happens in the offline mode using a dataset prepared from multivariate time series consisting of three-phase voltage and current signals. The online HIF detection uses the weights and thresholds obtained from the offline training. As the data preprocessing step is the same for both training and detection, the preprocessing is described first, followed by the explanation of training and detection.

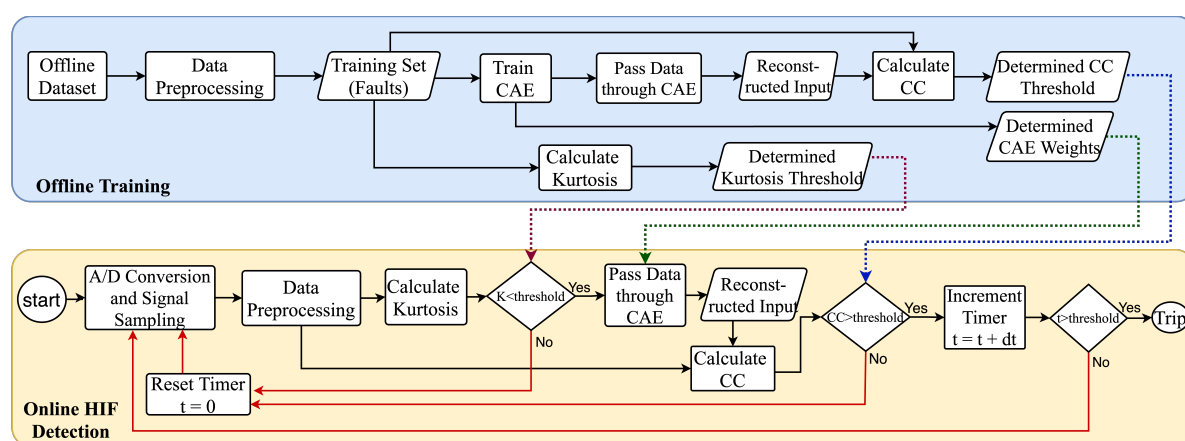


Figure 2. CAE Framework for HIF Detection (CAE-HIFD).

3.1. Data Preprocessing

The first step of data preprocessing entails the sliding window approach applied to time-series data to transform it into a representation suitable for the CAE. As illustrated in Figure 3, the first n samples (readings) make the first data window; thus, the data window dimension is n time steps \times f number of features. In each iteration of the algorithm,

the data window slides for s time steps, where s is referred to as a *stride*, to create the second data window and so on. Note that Figure 3 illustrates a case where $s = n$.

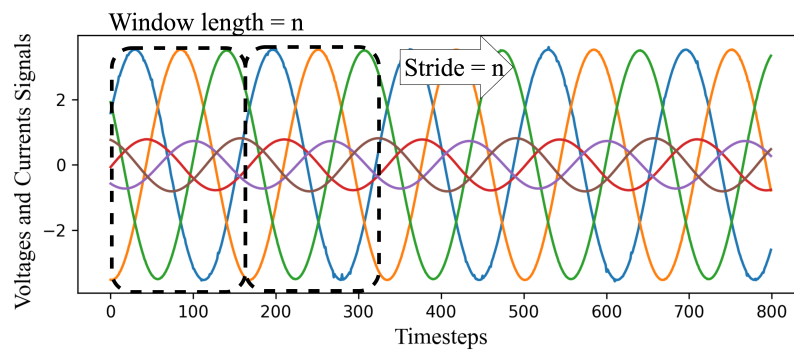


Figure 3. Sliding window approach.

Each voltage and current phase signal (each feature) in the sliding window is processed individually by differencing. The first-order d^1 and second-order d^2 differencing of signal $y(t)$, e.g., phase A voltage, are as follows:

$$d^1(t) = y(t) - y(t - 1) \quad (4)$$

$$d^2(t) = d^1(t) - d^1(t - 1) \quad (5)$$

The HIF causes small distortions in the voltage and current waveforms. The second-order differencing helps the CAE to learn and detect the HIF pattern by amplifying these distortions and suppressing the fundamental frequency component of each input signal. Differencing also amplifies noise; nevertheless, the generalization and spatial feature extraction capabilities of the CAE make the CAE-HIFD robust against noise as demonstrated in Section 4.5.

3.2. Offline Training

The CAE is trained solely with the fault data, and the non-fault data are only utilized for the system validation. As illustrated in Figure 2, the preprocessed data are passed to the CAE as one data window, $n \times f$ matrix, at a time. As shown in Figure 4, the CAE is composed of two main components: encoder and decoder [29].

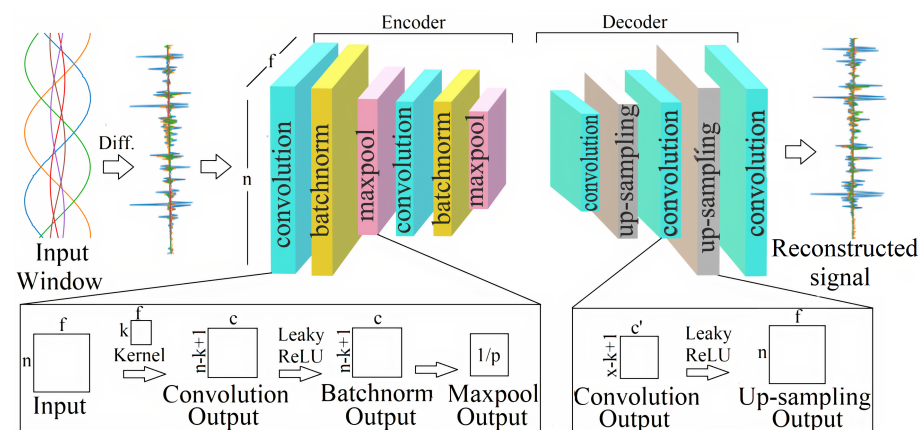


Figure 4. Convolutional autoencoder (CAE) structure.

The first layer in the encoder performs the 1D convolution operation on the $n \times f$ input matrix with the kernel of size $k \times f$. This kernel moves across the time steps of the input and interacts with k time steps (here $k < n$) of the input window at a time; thus, during the CAE training, the kernel learns the local spatial correlations in the input samples. There are

m kernels in the first layer and each kernel convolves with input to generate an activation map. Consequently, the output of the first layer has a dimension of $(n - k + 1) \times m$, and every column of this output matrix corresponds to the weights of one kernel. These kernel weights are learned during the CAE training process. Rectified linear unit (ReLU) activation function is often used to introduce non-linearity after the convolution. However, here LeakyReLU, a leaky version of ReLU, is used instead because ReLU discards the negative values in the sinusoidal wave [29]. Next, the batch normalization layer re-scales and re-centers data before passing them to the next layer in order to improve the training convergence. The batch normalized data are passed to the max-pooling layer to reduce data dimensionality and the associated computational complexity. The size of the max-pooling operation is p ; therefore, the output of the pooling layer is $\frac{1}{p}$ of the convolved input. As illustrated in Figure 4, the convolution, batch normalization, and max-pooling layers are repeated two times to extract features on different levels of abstraction. These encoder layers create an encoded representation of the input signal which is passed to the decoder.

Although the encoder decreases the dimensionality of the input, the decoder re-constructs the original signal from these encoded values. In the decoder, as illustrated in Figure 4, the convolutional layer first generates the activation map, and then the up-sampling operations increase the dimensionality of the down-sampled feature map to the input vector size. During up-sampling, the dimensionality of the input is scaled by repeating every value along the time steps in the signal with the scaling factor set according to the max-pooling layer size in the encoder. Similar to the encoder, in the decoder, the convolutional and up-sampling layers are repeated twice (Figure 4).

The CAE optimizes the weights and the biases using the back-propagation process in which the gradient descent is applied based on the loss function, typically MSE. In the proposed CAE-HIFD, the MSE is utilized as the loss function for training the CAE using fault data. In an autoencoder, the MSE is also referred to as a reconstruction error as it evaluates the similarity between the input signal and the reconstructed signal given by the autoencoder output. As the objective of the gradient descent algorithm is to minimize the MSE for training data, the MSE is expected to be low for the training data and high for any deviations from the training patterns.

In the CAE-HIFD, the CAE sees only the fault data during training, and consequently, the trained CAE is expected to fail in reconstructing the non-fault data input. Therefore, the MSE for the non-fault data is expected to be higher than the MSE for the learned fault data. Traditionally, in autoencoders, the separation between fault and non-fault data is done based on a threshold which is determined using the MSEs of the training dataset. However, in HIF detection, when CAE is trained with fault data, MSE is not a reliable metric for calculating the threshold. As illustrated in Figure 5, the differentiated fault data forms a complex pattern with a high number of fluctuations causing the dissimilarities between the CAE output and input. The magnitude of these fluctuations varies from -2.0 to 1.0 and, as a result, even a small mismatch between input and CAE output leads to high MSE: for example, in Figure 5a, MSE for fault data window is 0.0244 . On the other hand, in Figure 5b, MSE for steady state data window is 0.0002 because of a relatively simpler pattern compared to HIFs and small amplitudes of differentiated signal oscillations varying from -0.04 to 0.04 . Consequently, the MSE is not a reliable indicator to discriminate between HIF and non-fault cases.

In signal processing, a metric commonly used to evaluate the similarity between signals is the cross-correlation (CC) [39] which is defined as:

$$CC = (f * g)(\tau) = \int_{-\infty}^{\infty} \overline{f(t)} g(t + \tau) dt \quad (6)$$

where f and g are two signals and τ is a time shift in the signal.

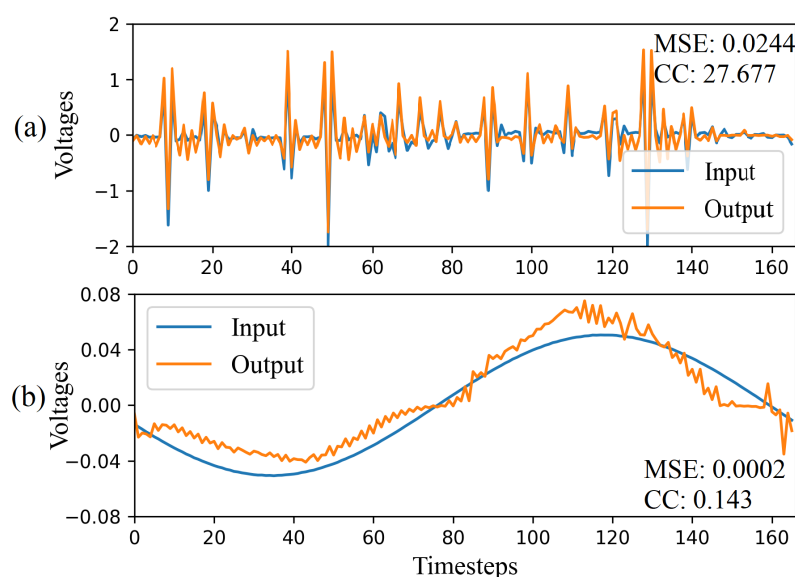


Figure 5. Input and output of the CAE for (a) HIF and (b) Non-HIF scenario.

In CAE-HIFD, CC is used to measure the similarity between the CAE input and output signals. As illustrated in Figure 2, after the CAE training is completed, the trained CAE reconstructs all data windows from the training set and obtains reconstructed signals. Next, for each window in the training set, the CC value is calculated for the input signal and the corresponding CAE output. As seen in Figure 5a, the HIF data window has a CC value of 27.677 because the input and output signals of the CAE are similar. On the contrary, a normal steady-state operating condition data window, Figure 5b, has a low CC value of 0.143 as the output deviates from the input. As the minimum CC value from the training set represents the least similar input–output pair from the training set, this minimum CC value serves as the CC threshold for separating HIF and non-HIF cases.

The CAE perceives the responses to disturbances, such as capacitor and load switching, to be HIFs because these disturbances cause waveform distortions. However, these disturbances usually occur for a shorter duration of time than HIFs making their statistical distribution different from those of HIFs and steady-state operation. Figure 6 shows that the disturbances and HIFs both exhibit Gaussian behavior, but the disturbances have a thinner peak and flatter tails on the probability density function (PDF) plot. In contrast, steady-state operation data (sinusoidal waveforms) have an arcsine distribution.

To distinguish disturbances from HIFs, the statistical metrics kurtosis is used. The kurtosis provides information about the tailedness of the distribution relative to the Gaussian distribution [39]. For univariate data $y_1, y_2, y_3, \dots, y_n$ with standard deviation s and mean \bar{y} , the kurtosis is:

$$K = \frac{\sum_{i=1}^n (y_i - \bar{y})^4 / n}{s^4} \quad (7)$$

As Figure 6 shows, flatter tails and thinner peaks results in higher kurtosis values. For example, the distribution of the differentiated capacitor switching disturbance in Figure 6b has a kurtosis value of $K = 76.6$ which is higher than the $K = 1.9$ for the HIF distribution in Figure 6f.

The kurtosis is calculated from the training set individually for each data window after applying differencing. To prevent misinterpretation of the K values and avoid treating HIFs as non-fault disturbances, the kurtosis threshold must be higher than every K value present in the training set. Accordingly, the kurtosis threshold is the value below which all the K values of the training data lie.

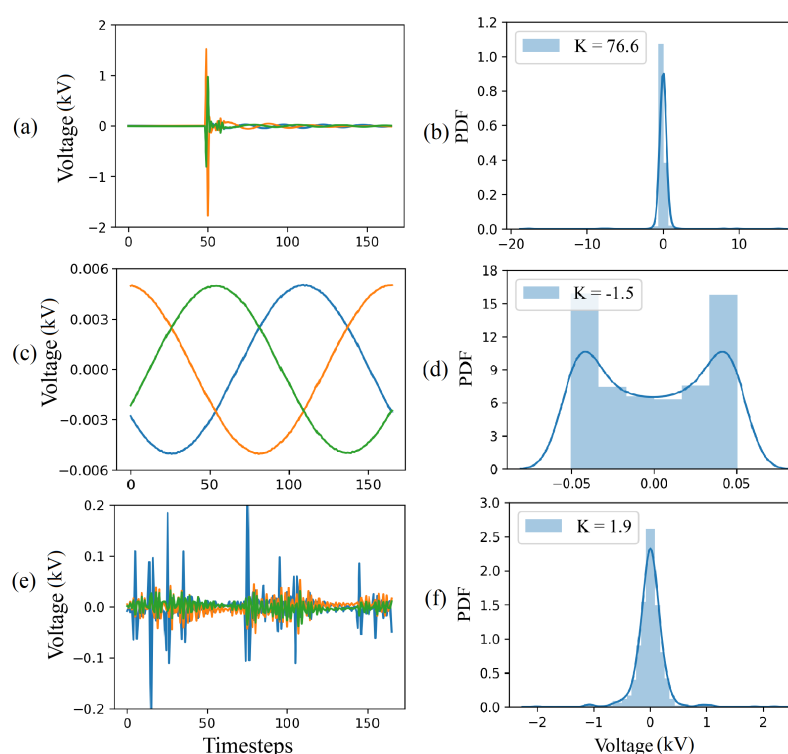


Figure 6. Kurtosis analysis: (a) differentiated voltage signal corresponding to a capacitor switching disturbance, (b) PDF of the voltage signal corresponding to a capacitor switching disturbance (c) differentiated normal steady-state voltage, (d) PDF of the normal steady-state voltage, (e) differentiated HIF voltage, and (f) PDF of the HIF voltage.

The artifacts of the offline training are the CC threshold, the learned CAE weights, and the kurtosis threshold. These artifacts are used for online HIF detection.

3.3. HIF Detection

The online HIF detection algorithm uses the artifacts generated by offline training as illustrated in Figure 2. First, the analog input signal is converted to digital by the A/D converter and the data preprocessing module generates data windows which proceed through the remaining HIF detection components, one window at the time.

The value of kurtosis is calculated for each data window and compared with the corresponding threshold obtained from the offline training. Any data window with the kurtosis value above the threshold is identified as a non-fault disturbance case for which the CAE is disabled because there is no need for additional processing as the signal is already deemed to be a disturbance. Next, the timer is reset for processing the next input signal segment.

If the kurtosis value is less than the threshold, the data window is sent to the trained CAE which encodes and reconstructs the signal. As the CAE is trained with fault data, for HIFs, the reconstructed signal is similar to the original signal. This similarity is evaluated by calculating the CC between the reconstructed signal and the original signal. If the CC value of the data window is greater than the CC threshold determined in the training process, the signal is identified to be corresponding to a HIF.

Under transient disturbances, such as capacitor switching, the value of CC may exceed the corresponding threshold for a short time period immediately after the inception of disturbance. False identification of disturbances as HIFs is prevented using a pick-up timer. The timer is incremented when the CC exceeds its threshold and is reset to zero whenever the CC or K indicates a non-HIF condition, as shown in Figure 2. A tripping (HIF detection) signal is issued when the timer indicates that the time duration of the HIF exceeds a predetermined threshold.

4. Evaluation

This section first describes the study system and the process of obtaining data for the performance verification studies. Next, the details of CAE-HIFD model training and the effects of different CAE-HIFD components are presented. Furthermore, the response of the CAE-HIFD to different case studies is demonstrated. Finally, the CAE-HIFD performance is compared with other HIF detection approaches, and its sensitivity to noise is examined.

4.1. Study System

The dataset utilized for model training and evaluation is obtained through time-domain simulation studies performed in the PSCAD software. The study system is the IEEE 13 node test feeder of Figure 7, a realistic 4.16 kV distribution system with a significant load unbalance. This test feeder was selected in order to examine the system behavior under challenging load unbalance conditions and because of its common use in HIF studies [40]. Detailed information regarding the line and the load data are provided in the Appendix A, and further information about this benchmark system can be found in [40]. For an accurate representation of the HIF behavior, the antiparallel diode model of Figure 8, [2,3,5,6,41] is utilized. The HIF model parameters representing seven different faulted surface types are given in Table 1 [2,42]. These parameters lead to effective fault impedances as high as 208 ohms in 4.16 kV distribution system.

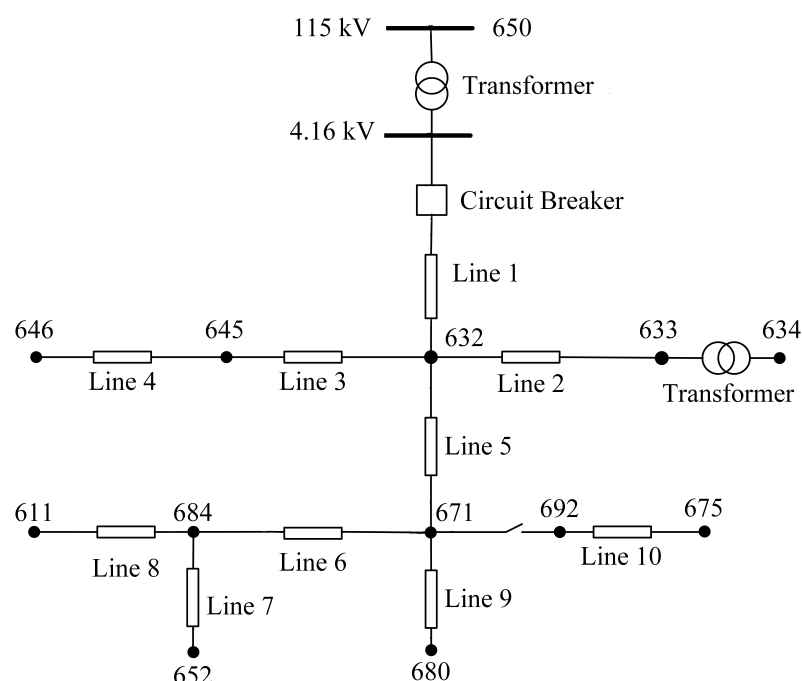


Figure 7. IEEE 13 node test feeder.

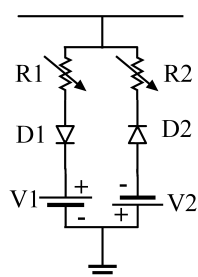


Figure 8. Nonlinear HIF model utilized for time-domain simulation studies.

In total, 210 faulty cases were simulated: 7 different surfaces, 10 fault locations, and 3 phases. After the windowing technique, this resulted in 1372 HIF data windows.

Additionally, the dataset obtained from simulations contained 272 non-fault data windows. Using data obtained from simulation studies enables considering diverse fault types, locations, and surfaces while obtaining such diverse data from real-world experiments would be difficult or even impossible. Of the fault data, 80% are assigned for the model training and the rest for testing. As CAE-HIFD requires only fault data for training, all non-fault data are assigned for testing. Of the training set, 10% are used as a validation set for the hyperparameter optimization.

Table 1. HIF Model parameters For different surfaces.

Surfaces	R1 (Ω)	R2 (Ω)	V1 (V)	V2 (V)
Wet Sand	138 \pm 10%	138 \pm 10%	900 \pm 150	750 \pm 150
Tree Branch	125 \pm 20%	125 \pm 20%	1000 \pm 100	500 \pm 50
Dry Sod	98 \pm 10%	98 \pm 10%	1175 \pm 175	1000 \pm 175
Dry Grass	70 \pm 10%	70 \pm 10%	1400 \pm 200	1200 \pm 200
Wet Sod	43 \pm 10%	43 \pm 10%	1550 \pm 250	1300 \pm 250
Wet Grass	33 \pm 10%	33 \pm 10%	1750 \pm 350	1400 \pm 350
Rein. Concrete	23 \pm 10%	23 \pm 10%	2000 \pm 500	1500 \pm 500

4.2. CAE-HIFD Model Training

The performance of the CAE is evaluated using accuracy (Acc), security (Sec), dependability (Dep), safety (Saf), and sensibility (Sen) [43].

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (8)$$

$$Sec = \frac{TN}{TN + FP} \times 100\% \quad (9)$$

$$Dep = \frac{TP}{TP + FN} \times 100\% \quad (10)$$

$$Saf = \frac{TN}{TN + FN} \times 100\% \quad (11)$$

$$Sen = \frac{TP}{TP + FP} \times 100\% \quad (12)$$

Here, true positives (TP) and true negatives (TN) are the numbers of correctly identified fault and non-fault cases, and false negatives (FN) and false positives (FP) are the numbers of miss-classified fault and non-fault cases. The accuracy is the percentage of overall correctly identified states, the security is the healthy state detection precision, the dependability is the fault state detection precision, the safety is resistance to faulty tripping, and the sensibility is resistance to unidentified faults [43]. Note that dependability is also referred to as true positive rate (TPR) or sensitivity, while security is referred to true negative rate (TNR) or specificity [32].

To achieve high accuracy, the CAE hyperparameters must be tuned; this includes the number and size of kernels, learning rate, optimizer, and batch size. Hyperparameter tuning is performed with grid search cross-validation (GSCV), wherein an exhaustive search is conducted over pre-specified parameter ranges. The GSCV determines the best performing parameters based on the scoring criteria provided by the model, in our case accuracy. The tuned CAE has 256-128-128-256 filters of size 3×3 in the four convolution layers, the optimizer is Adam, the learning rate is 0.001, and the batch size is 16.

The window size after differencing is 166 voltage/current samples, which corresponds to one cycle of the 60 Hz power frequency signal sampled at a rate of 10 kHz. The proposed method does not operate based on the fundamental frequency components of the input signal and, thus, is not sensitive to frequency deviations as shown in Section 4.4.7. The sliding window stride during training impacts the CAE-HIFD performance: its value is determined on the training dataset. Once the system is trained, it is used with the stride

of one. The upper bound for the stride value is 166 as a higher value would lead to skipped samples. The performance metrics for varying stride values are shown in Figure 9. It can be observed that the safety and dependability are not affected by the change in the stride value as none of the non-HIF cases are misclassified as a HIF case. The accuracy, security, and sensibility are 100% for the stride size of 166, and for shorter strides, these metrics are slightly lower. As the stride decreases, a few data windows are mistakenly identified as faults resulting in decrease in security, sensibility, and accuracy. Hence, the stride value of 166 is selected for the sliding window in the preprocessing of the training dataset.

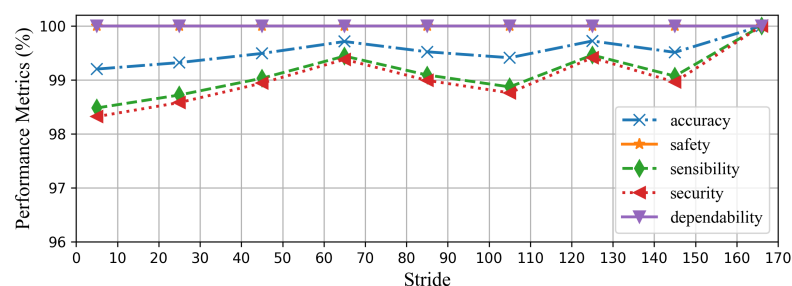


Figure 9. Impact of the sliding window stride on the performance metrics of the CAE-HIFD.

The CAE-HIFD prevents false detection of disturbances as HIFs using kurtosis value. As the distributions of the HIFs and disturbances both exhibit Gaussian behavior, an HIF window can have a K value close to the K value of a disturbance. The kurtosis threshold is determined starting from the kurtosis values for which all HIFs scenarios in the training data lie below this threshold: in this case 10. Next, accuracy on the training data is examined with thresholds close to this initial threshold. As illustrated in Figure 10, the accuracy is 100% when kurtosis thresholds are between 9.5 and 10.5. The accuracy decreases when threshold is below 9.5 because some of the HIF scenarios are mistakenly detected as non-HIF scenarios ($FN > 0$). Furthermore, the threshold above 10.5 leads to low accuracy as some non-HIF scenarios are falsely declared as HIFs ($FP > 0$). Consequently, the kurtosis threshold of 10 is selected to discriminate the non-fault disturbances from the HIFs.

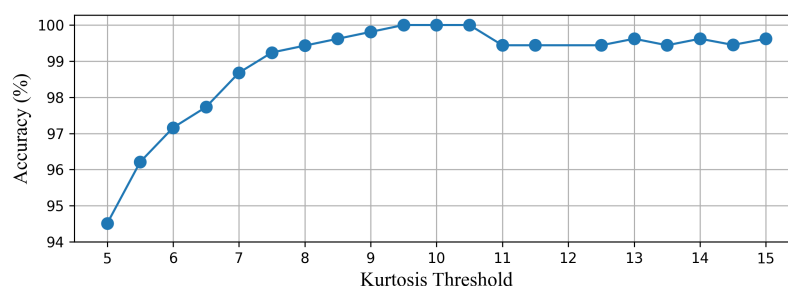


Figure 10. Impact of the kurtosis threshold on the accuracy of the CAE-HIFD.

4.3. Effects of CAE-HIFD's Components

The proposed CAE-HIFD uses differencing and cross-correlation in addition to the main component, the CAE, to increase various performance metrics. Furthermore, kurtosis is utilized to improve the security of the proposed method. Consequently, as depicted in Table 2, the CAE-HIFD achieves 100% performance in all five considered metrics regardless of the surface type, inception angle, and fault locations.

Additionally, Table 2 includes variants of the CAE-HIFD with only some of the three components included. With only CC and kurtosis, the accuracy and sensibility drop to nearly 51%, and the security decreases by 99.6%. In the absence of differencing, the CAE cannot learn patterns to distinguish between the HIF and the non-HIF data windows and, as a result, a large number of the non-HIF data windows are falsely classified as the HIF which means high FP, thus low security.

To examine the impact of CC, the traditional MSE is used in place of CC to measure the similarity of the input and reconstructed signal. As shown in Table 2, in the absence of CC, the values of accuracy and sensibility drop to nearly 50%. This happens because the MSEs calculated for the HIF and non-HIF data windows are similar and, thus, non-HIFs are falsely detected as HIFs. Furthermore, the security value is low (0.40%), whereas the dependability value is high (92.67%) as there are only a few TN and FN compared to TP and FP.

Table 2. Impact of differencing, cross-correlation, and kurtosis on CAE-HIFD performance.

Model	Acc	Saf	Sen	Sec	Dep
Proposed CAE-HIFD	100	100	100	100	100
CAE with CC, K	51.19	100	51.10	0.37	100
CAE with K, Diff	48.29	4.76	50.10	0.40	92.67
CAE with CC, Diff	96.38	100	93.31	92.70	100
CAE with CC	51.10	100	50.64	1.84	100
CAE with Diff	48.9	37.93	49.51	4.04	93.43
CAE with K	89.64	96.67	84.16	82.77	96.35

Omitting the kurtosis evaluation results in only a small increase in the number of FP cases which are the non-fault disturbances falsely declared as HIFs. Therefore, as shown in Table 2, all the performance metrics values decrease by less than 8%.

Finally, only one out of the three components is included in the CAE-HIFD framework. Whereas the simultaneous use of differencing and cross-correlation achieves relatively high performance metrics, with only one of the two components, there is a major decrease in security (more than 95%). With kurtosis only, all metrics are between 82% and 97% in comparison to 100% obtained in the presence of all three components. This is caused by the absence of differencing which assists in amplifying sign wave distortions and omission of cross-correlation which facilitates signal comparisons. The results shown in Table 2 highlight the necessity of each CAE-HIFD component and the contribution of each component to the HIF detection performance.

4.4. CAE-HIFD Response to Different Case Studies

In this section, seven case studies have been conducted to depict the response of the proposed CAE-HIFD.

4.4.1. Case Study I—Close-in HIF

Figure 11 illustrates the performance of the CAE-HIFD under both normal and HIF conditions. In this case study, the HIF is applied at Node 632 starting at 0.05 s, as seen in Figure 11a. The input voltage and current signals observed at the substation relay are shown in Figure 11b,c, and the kurtosis calculated from those voltages and currents is displayed in Figure 11d. During normal operation, the kurtosis is below the threshold; upon the HIF inception, it raises over the threshold for approximately 8–10 ms returning quickly back to below threshold values. The HIF causes the CC value to rise above the threshold, Figure 11e, and, therefore, a trip signal is issued approximately 60 ms after the HIF inception as seen in Figure 11f.

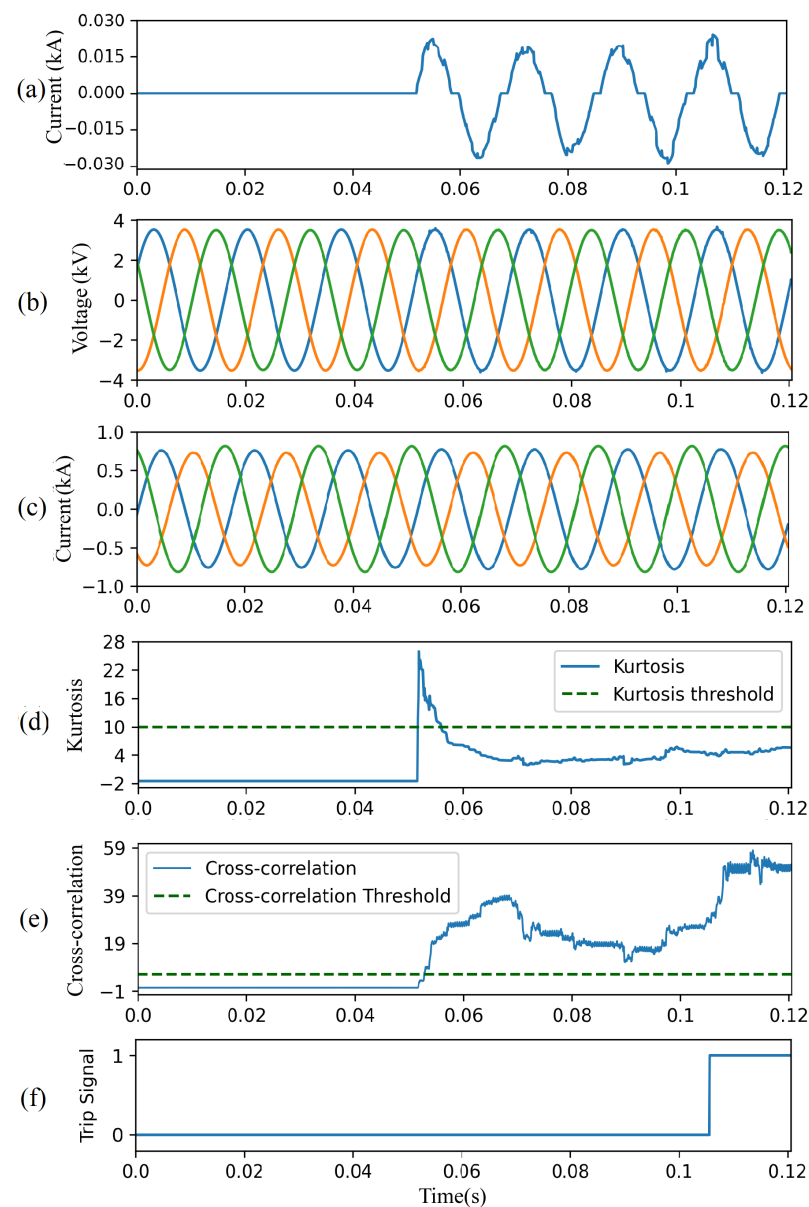


Figure 11. CAE-HIFD performance under normal and close-in HIF conditions: (a) HIF current, (b) three-phase input voltages, (c) three-phase input currents, (d) kurtosis, (e) cross-correlation, and (f) trip signal.

4.4.2. Case Study II—Remote HIF

Figure 12 depicts the result of the CAE-HIFD in presence of a remote HIF: the HIF is applied at Node 652 starting at 0.05 s, as seen in Figure 12a. The input voltage and current signals are observed in Figure 12b,c, and the calculated kurtosis is shown in Figure 12d. Due to the remote location of HIF, the HIF influence on the voltages and current signals is highly attenuated. As a result, the kurtosis surpasses the threshold for shorter duration of time (approximately 1–2 ms) remove as compared to Case study I. As shown in Figure 12e, the CC value raises above the threshold after the inception of HIF. Consequently, a trip signal is issued approximately 50 ms after the HIF inception as seen in Figure 12f.

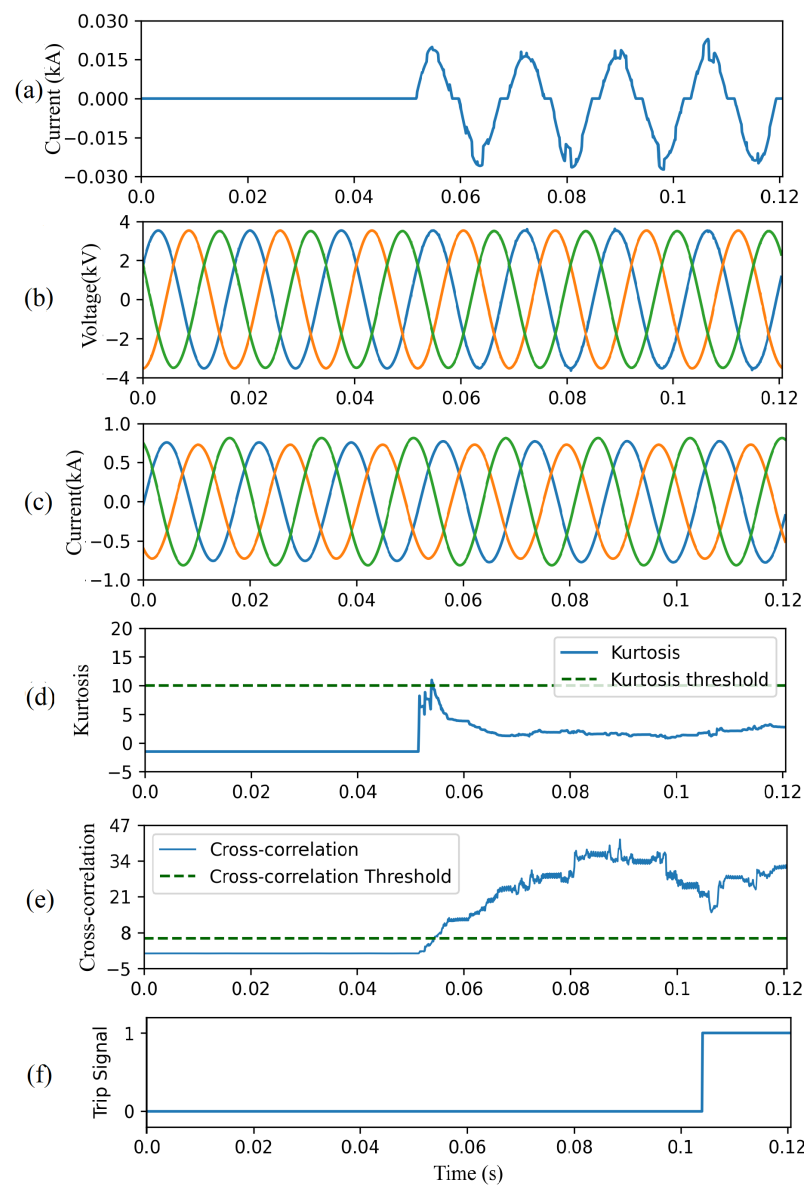


Figure 12. CAE-HIFD performance under normal and remote HIF conditions: (a) HIF current, (b) three-phase input voltages, (c) three-phase input currents, (d) kurtosis, (e) cross-correlation, and (f) trip signal.

4.4.3. Case Study III—Capacitor Switching

The proposed HIF detection method successfully discriminates HIFs from switching events as demonstrated in Figure 13 with a three-phase capacitor bank located at node 675. Figure 13a depicts phase A current caused by the capacitor energization at $t = 0.05$ s. The current and voltage signals seen by the relay at the substation exhibit significant oscillations, as shown in Figure 13b,c. This switching event causes sudden increase in the kurtosis for a short duration of time, approximately 15 ms (Figure 13d). Although the CC for the switching event is higher than its threshold, Figure 13e, this disturbance is not falsely identified as an HIF, due to the high kurtosis value. Moreover, the CC for the remaining non-HIF signal is below the threshold. Consequently, a trip signal is not issued throughout the switching event, Figure 13f.

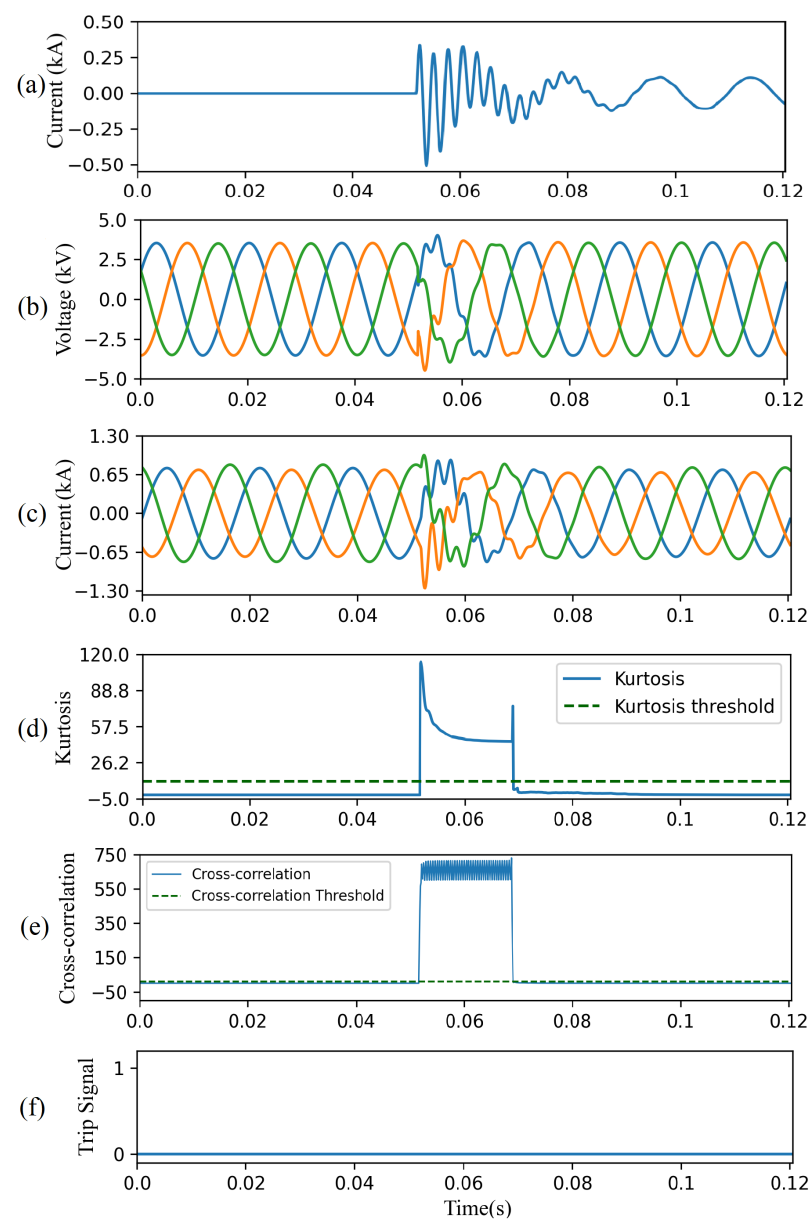


Figure 13. CAE-HIFD performance under a capacitor switching scenario: (a) capacitor current, (b) three-phase input voltages, (c) three-phase input currents, (d) kurtosis, (e) cross-correlation, and (f) trip signal.

4.4.4. Case Study IV—Non-linear Load

Figure 14 shows the performance of the proposed CAE-HIFD in presence of a non-linear load which causes significant harmonics. The load at node 634 is replaced by a DC motor fed by a six-pulse thyristor rectifier. The motor is started at $t = 0.05$ s. Figure 14a illustrates the phase A current of the non-linear load, while Figure 14b,c show voltages and currents measured by the relay at the substation. Although the CC is higher than its threshold (Figure 14e), the trip signal (Figure 14f) is not issued because the kurtosis surpasses its threshold (Figure 14d).

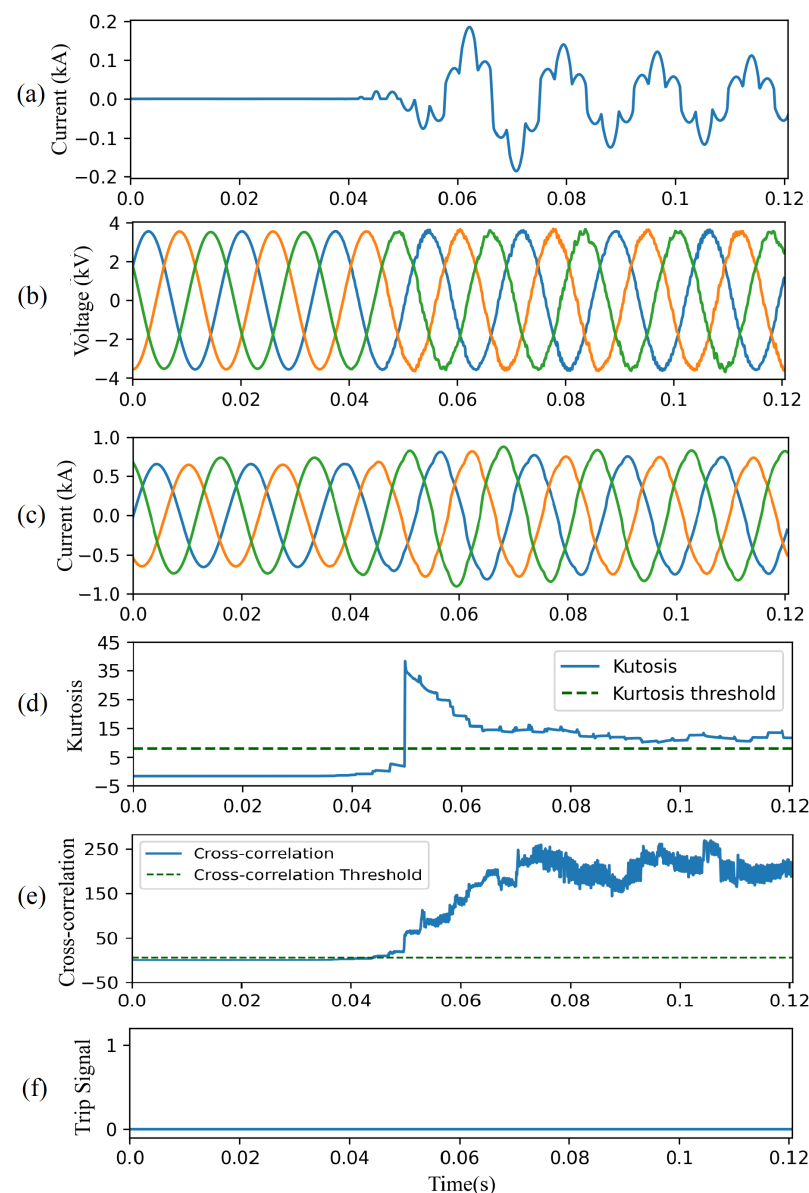


Figure 14. CAE-HIFD performance under non-linear load switching: (a) non-linear load current, (b) three-phase input voltages, (c) three-phase input currents, (d) kurtosis, (e) cross-correlation, and (f) trip signal.

4.4.5. Case Study V—Transformer Energization

This case study investigates the performance of the CAE-HIFD under a transformer energization scenario: the transformer at node 633 is energized at $t = 0.05$ s. The inrush current for phase-A is shown in Figure 15a while Figure 15b,c display voltages and currents measured at the substation. Both the resulting kurtosis shown in Figure 15d, and the CC shown in Figure 15e are below their corresponding thresholds. As a result, the proposed protection strategy does not cause any unnecessary tripping (Figure 15f) under the transformer energization scenario.

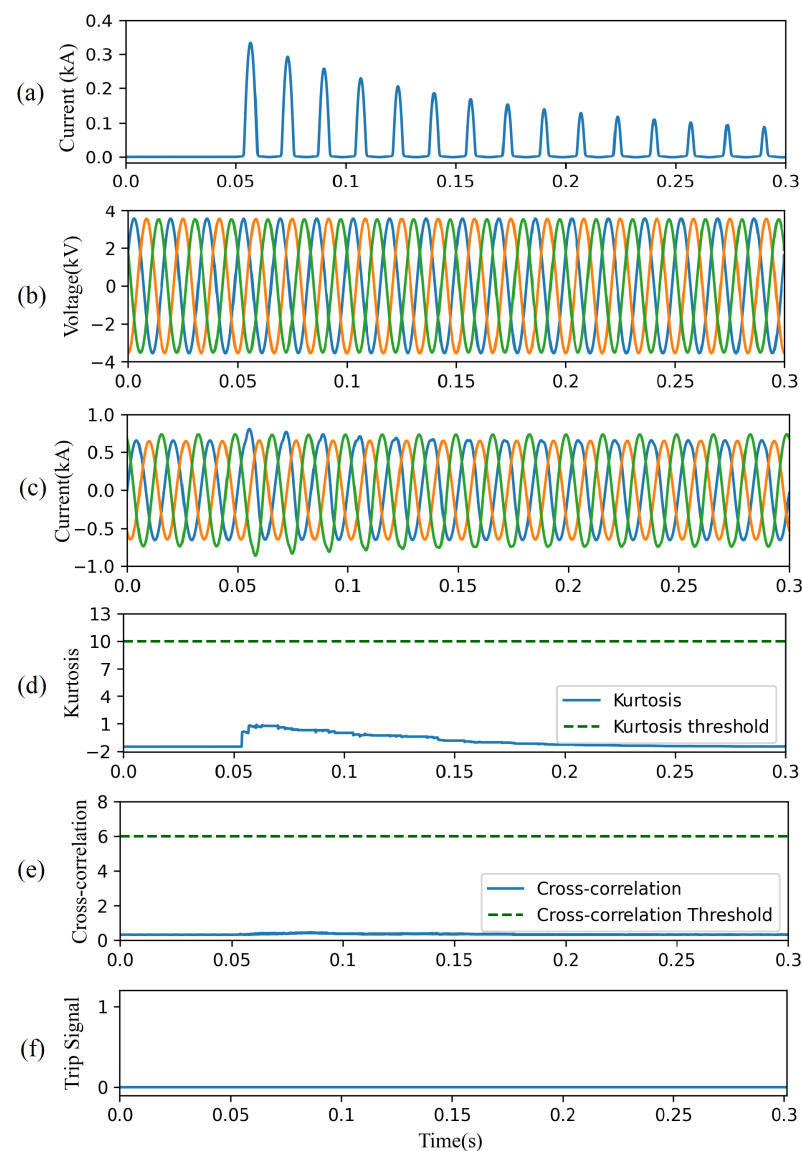


Figure 15. CAE-HIFD performance under inrush currents: (a) Inrush current, (b) three-phase input voltages, (c) three-phase input currents, (d) kurtosis, (e) cross-correlation, and (f) trip signal.

4.4.6. Case Study VI—Intermittent HIFs

This case study demonstrates the effectiveness of the proposed CAE-HIFD in detecting intermittent HIFs. A tree branch momentarily connects the phase-A to the ground for approximately 3.5 cycles (55 ms) as illustrated in Figure 16a. The voltage and current signals shown in Figure 16b,c are measured by the relay at the substation. As depicted in Figure 16d, the kurtosis does not exceed the threshold. The CC in Figure 16e crosses the threshold during the intermittent faults. As shown in Figure 16f, the trip signal is issued after 50 ms. The trip signal is reset after the intermittent fault is cleared.

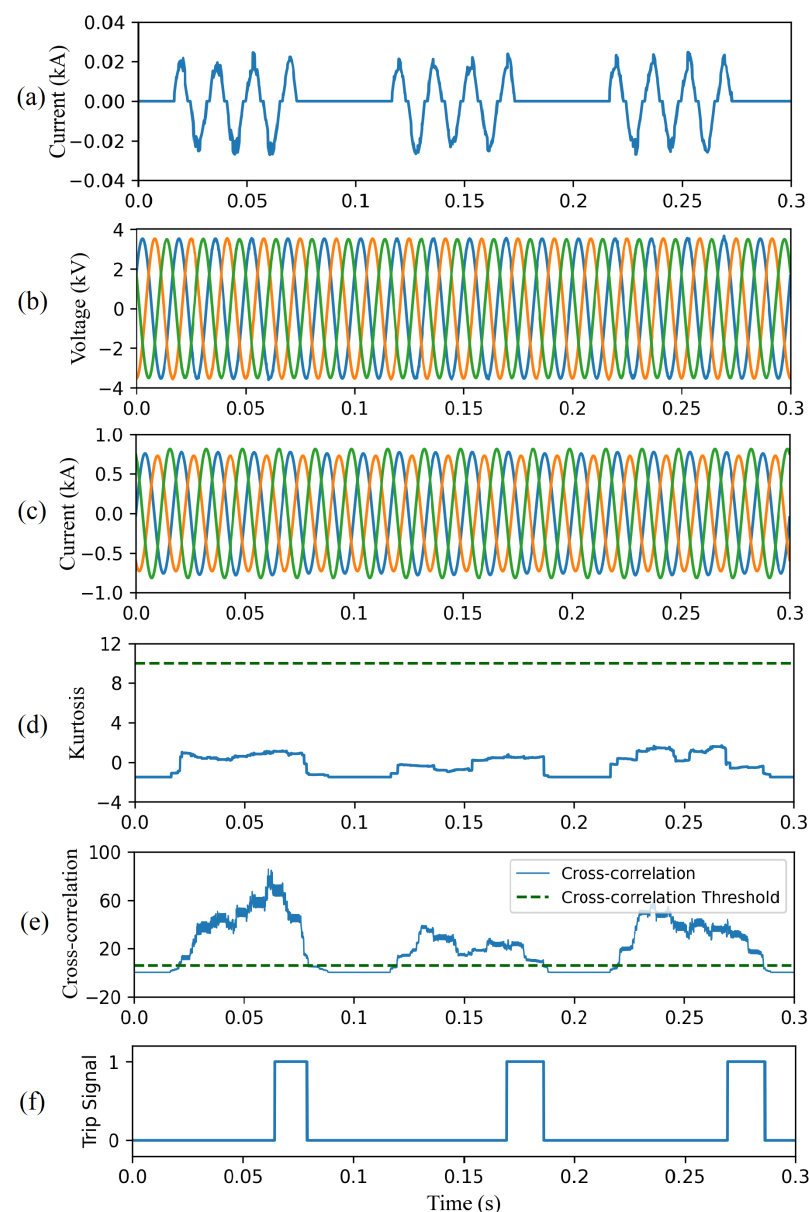


Figure 16. CAE-HIFD performance under intermittent HIF condition: (a) fault current, (b) three-phase input voltages, (c) three-phase input currents, (d) kurtosis of the input, (e) cross-correlation, and (f) trip signal.

4.4.7. Case Study VII—Frequency Deviations

To demonstrate the effectiveness of the proposed method in presence of frequency deviations, the system frequency is increased to 61 Hz in this case study. The HIF is initiated at $t = 0.05$ s. Figure 17a,b represent currents and voltages measured by the relay at the substation. As shown in Figure 17c, before the HIF takes place, the kurtosis is below the threshold. As the the HIF samples enters the sliding window, the kurtosis exceeds the threshold because the distribution suddenly changes during the transition. Next, the kurtosis returns to values below the threshold. The CC in Figure 17d is above the CC threshold; therefore, the system trips within three cycles of the HIF inception.

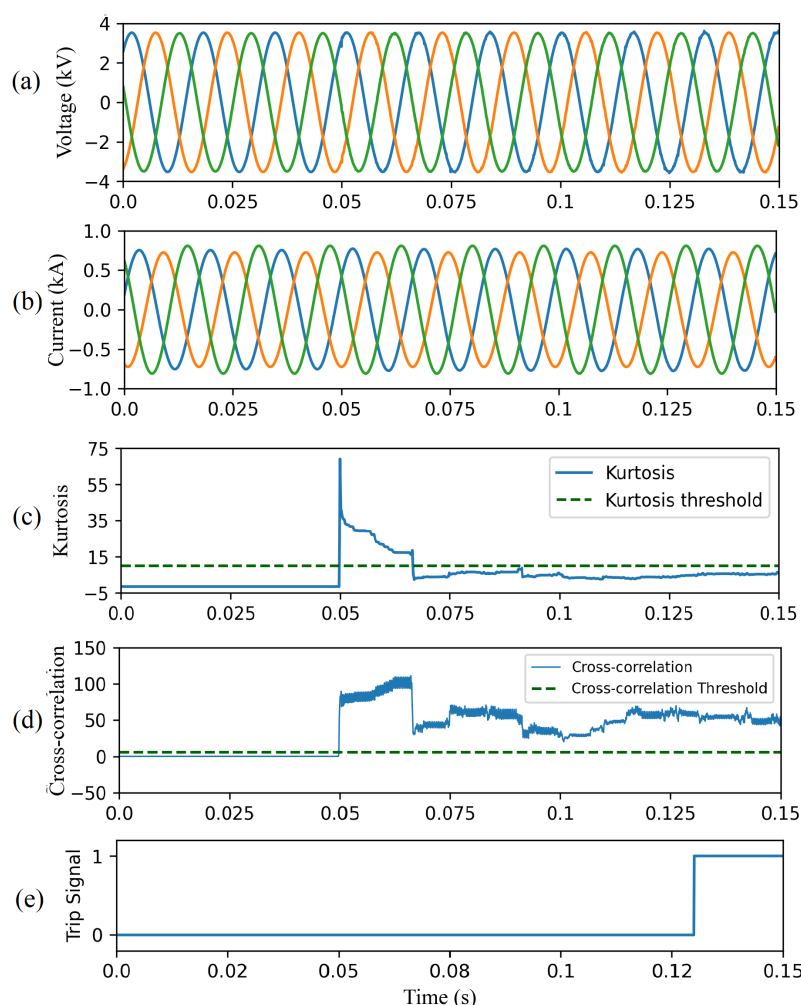


Figure 17. CAE-HIFD performance under 61 Hz fundamental frequency: (a) three-phase input voltages, (b) three-phase input currents, (c) kurtosis, (d) cross-correlation, and (e) trip signal.

4.5. Comparison with Other Approaches

This section first compares the proposed CAE-HIFD with other supervised and unsupervised learning algorithms. The two supervised models selected for the comparison are: support vector machine (SVM) [6] and artificial neural network (ANN) [21]. As supervised models require the presence of both, HIF and non-HIF data in the training set, these models are trained with a dataset containing an equal number of the HIF and non-HIF instances. Moreover, as those models have originally been used with the DWT applied on the current waveform [6,21], DWT is used here too. DWT extracts features by decomposing each phase current into seven detail level coefficients and one approximate level coefficient using the db4 mother wavelet. The features are formed by computing the standard deviation of coefficients at each level; therefore, eight standard deviations from each phase form a new input sample with 24 elements [21]. As with CAE-HIFD, the SVM and ANN hyperparameters are tuned using GDCV. The SVM kernel is RBF with γ of 0.05. The ANN has three layers with 24-18-1 neurons, the activation function for input and hidden layers is ReLU, and binary cross-entropy is the loss function.

As other studies have only used supervised learning, to examine unsupervised learning techniques, variations of the proposed approach are considered in this evaluation. Figure 18 shows the flowchart for the unsupervised ML models. Preprocessing, kurtosis, and CC calculation components are exactly the same as in the proposed CAE-HIFD, while two options are considered for the autoencoder algorithm and the training dataset. As the autoencoder, the proposed CAE-HIFD uses CAE while here we consider a variant of recurrent neural network, gated recurrent units autoencoder (GRU-AE). GRU-AE is selected

because it is successful in extracting patterns from time-series data such as those present in the current and voltage signals. The GRU-AE tuned with GDCV has two hidden layers, each one with 32 GRU cells and the ReLU activation function. For both, CAE-HIFD and GRU-AE, two types of studies are conducted: training on HIFs data only and training on non-HIFs data only.



Figure 18. Flowchart for the alternative unsupervised HIF detection models.

The results of the comparison between CAE-HIFD and the other approaches are shown in Table 3. It can be observed that CAE-HIFD outperforms other approaches and is the only one not susceptible to false tripping as indicated by the security metrics. In addition, the CAE-HIFD trained only with HIF data is highly efficient in discriminating non-HIF instances by detecting deviations from the learned HIF patterns. This prevents the algorithm from false tripping in the case of a new non-HIF pattern not present in the training set. The results of the studies presented in Table 3 indicate that the CAE-HIFD achieves equally good results regardless of whether it is trained on the HIF data or non-HIF data; however, when trained with non-HIF data, there is a risk of identifying new non-HIF patterns as HIFs. The supervised learning-based approaches category can only recognize the pattern present in the training set and, thus, may recognize new non-HIF events as HIFs. Overall, the proposed CAE-HIFD achieves better performance than the other approaches.

Table 3. Comparison of CAE-HIFD with other HIF detection approaches.

Model	Acc	Saf	Sen	Sec	Dep
Other Approaches-Supervised					
DWT+SVM [6]	97.97	100	97.78	78.99	100
DWT+ANN [21]	97.72	100	97.55	76.47	100
Variants of our approach-Unsupervised					
Train on non-faults					
GRU-AE	99.92	100	96.92	99.92	100
Proposed CAE-HIFD	100	100	100	100	100
Train on faults					
GRU-AE	34.61	32.24	83.22	97.52	5.66
Proposed CAE-HIFD	100	100	100	100	100

4.6. Robustness of the Proposed CAE-HIF against Noise

To examine CAE-HIFD robustness against noise, studies are conducted by introducing different levels of noise. The white Gaussian noise is considered because it covers a large frequency spectrum. The noise is added to the current signals because current waveforms are more susceptible to noise [1]. As shown in Figure 19, the proposed CAE-HIFD approach is immune to noise when signal to noise ratio (SNR) value is higher than 40 dB. In a case of high noise, SNR below 40 dB, the accuracy reduces to 97%. Thus, more than one consecutive window is needed to be processed before making a tripping decision in order to avoid undesired tripping and to ensure accurate HIF detection. Therefore, three consecutive windows are utilized in all performance evaluation studies in order to accurately detect all HIFs. Increasing the timer threshold improves the resiliency against unnecessary tripping, but prolongs HIF detection time. Even with the extremely noisy condition of 1 dB SNR, the accuracy, security, and sensibility do not fall below 97.65%, 94.03%, 96.28%, respectively. The reason behind this robustness is the CAE de-noising ability and strong pattern learning capability. The inherent de-noising nature of the autoencoders assists the CAE to generalize the corrupted input. Additionally, the CAE-HIFD learns the complex HIF patterns because of spatial feature learning proficiency of the CAE. The model accurately detects non-HIF

scenarios under considered noise levels; hence, the safety and dependability remains at 100% even with high levels of noise. Moreover, the values of other performance metrics are also greater than 90% throughout the SNR range of 5dB to 50dB demonstrating noise robustness of the CAE-HIFD.

Figure 20 shows CAE-HIFD performance in presence of noise of 20 dB SNR. Before the HIF inception at 0.05 s, despite the significant noise, the kurtosis (Figure 20b) and the CC (Figure 20c) remain below their thresholds. The CC surpasses the threshold upon HIF inception and, as a result, the designed protection system issues trip signal (Figure 20d).

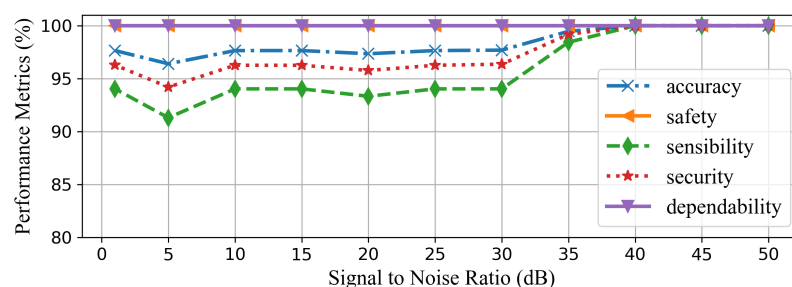


Figure 19. Effect of noise on the CAE-HIFD performance.

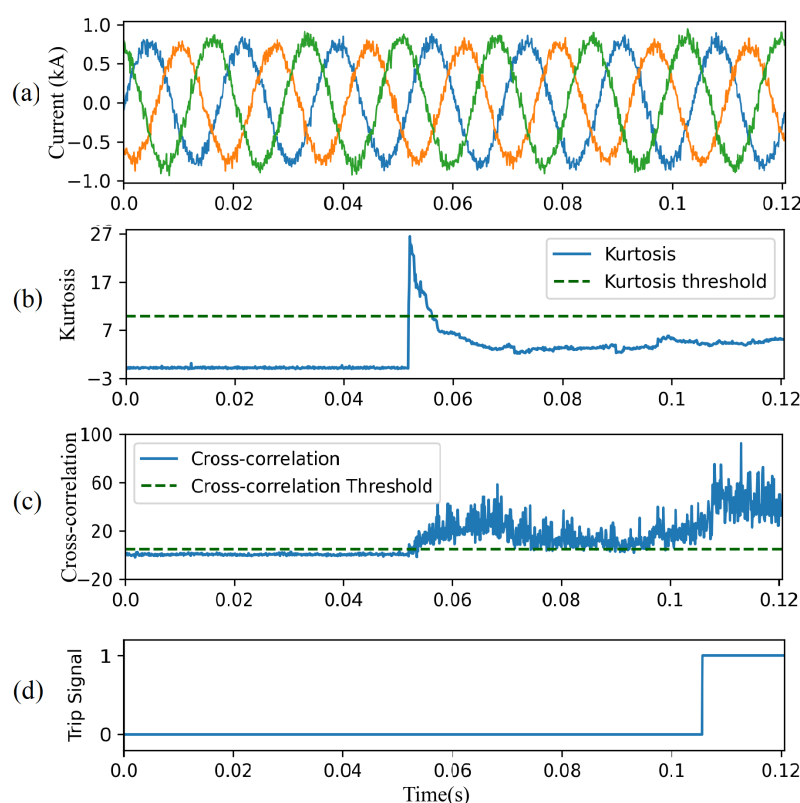


Figure 20. CAE-HIFD performance under 20 dB SNR: (a) three-phase input currents, (b) kurtosis, (c) cross-correlation, and (d) trip signal.

4.7. Discussion

The evaluation results demonstrate that the proposed CAE-HIFD achieves 100% HIF detection accuracy irrespective of the surface type, fault phase, and fault location. All metrics, including accuracy, safety, sensibility, security, and dependability are at 100% as shown in Table 2. Moreover, for all considered scenarios, the system trips within three cycles after the HIF inception.

The challenging part of machine learning for HIF detection is in the diversity of non-fault and fault signals together with similarities between non-HIF disturbances and HIFs.

By training on faults only, the proposed approach does not require simulation of non-fault scenarios for training. Distinguishing HIFs from the non-HIF steady-state operation can take advantage of the smoothness of non-HIF steady-state signal; however, non-HIF disturbances, such as capacitor and load switching, share many characteristics (e.g., randomness and non-linearity) with the HIF signals making it difficult for a neural network (in our case CAE) to distinguish between them. To address this challenge, the proposed approach takes advantage of differences in data distributions between non-fault disturbances and HIFs and employs kurtosis to differentiate between the two.

In experiments, 210 fault cases were considered corresponding to 1372 fault data windows as described in Section 4.1. Signals corresponding to these faults are different from each other as simulations included different surfaces, fault locations, and fault phases. From these fault cases, 80% is selected randomly for training, therefore, some cases are present only in testing. Moreover, all the case studies presented in Section 4.4 are conducted with data that are not seen by the proposed CAE-HIFD in training. The proposed system successfully distinguished between fault and non-fault signals for all scenarios, which demonstrates its abilities to detect previously unseen HIF and non-HIF scenarios.

Frequency deviations, as well as noise, impose major challenges for the HIF detection. Approaches that operate on the fundamental frequency components risk failures in presence of frequency deviations. However, CAE-HIFD does not operate based on the fundamental frequency components of the input signals and, consequently, is not sensitive to frequency deviations as shown in Section 4.4.7. As noise is common in distribution systems, it is important to consider it in HIF detection evaluation. HIF detection in presence of noise is difficult as noisy signals are accompanied by randomness and have characteristics that resemble HIFs. Nonetheless, experiments from Section 4.6 show that CAE-HIFD remains highly accurate even in presence of significant noise.

5. Conclusions

Recently, various machine learning-based methods have been proposed to detect HIFs. However, these methods utilize supervised learning; thus, they are prone to misclassification of HIF or non-HIF scenarios that are not present in the training data.

This paper proposes the CAE-HIFD, a novel deep learning-based approach for HIF detection capable of reliably discriminating HIFs from non-HIF behavior including diverse disturbances. The convolutional autoencoder in CAE-HIFD learns from the fault data only, which eliminates the need of considering all possible non-HIF scenarios for the training process. The MAE commonly used to compare autoencoder input and output is replaced by cross-correlation in order to discriminate HIFs from disturbances such as capacitor and load switching. To distinguish switching events from HIFs, the CAE-HIFD employs kurtosis analysis.

The results show that CAE-HIFD achieves 100% performance in terms of all five metrics of protection system performance, namely accuracy, security, dependability, safety, and sensitivity. The proposed CAE-HIFD outperforms supervised learning approaches, such as the SVM with DWT and the ANN with DWT, as well as the unsupervised GRU-based autoencoder. The CAE-HIFD performance is demonstrated on case studies including steady-state operation, close-in and remote HIFs, capacitor switching, non-linear load, transformer energization, intermittent faults, and frequency deviations. The studies on the effect of different noise levels demonstrate that the proposed CAE-HIFD is robust against noise for SNR levels as low as 40 dB and provides acceptable performance for higher noise levels.

Future work will examine HIF detection from only voltage or on current signals in order to reduce computational complexity. Furthermore, HIF classification technique will be developed to determine the phase on which the fault occurred.

Author Contributions: Conceptualization, K.R.; methodology, K.R.; software, K.R. and F.H.; validation, K.R. and F.H.; formal analysis, K.R., F.H., K.G. and F.B.A.; investigation, K.R. and F.H.; resources, K.G. and F.B.A.; data curation, F.H.; writing—original draft preparation, K.R. and F.H.;

writing—review and editing, K.G., F.B.A., K.R., F.H.; visualization, K.R.; supervision, K.G. and F.B.A.; project administration, K.G. and F.B.A.; funding acquisition, K.G. and F.B.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research has been supported by NSERC under grants RGPIN-2018-06222 and RGPIN-2017-04772.

Data Availability Statement: The data are not publicly available due to privacy.

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

Abbreviations

The following abbreviations are used in this manuscript:

CAE-HIFD	Convolutional Autoencoder framework for HIF Detection
CAE	Convolutional Autoencoder
HIF	High-Impedance Fault
EMD	Empirical Mode Decomposition
VMD	Variational Mode Decomposition
ML	Machine Learning
SVM	Support Vector Machine
SNR	Signal to Noise Ratio
DWT	Discrete Wavelet Transform
WT	Wavelet Transform
CNN	Convolutional Neural Network
CC	Cross-Correlation
MSE	Mean Squared Error
ReLU	Rectified Linear Unit
K	Kurtosis
Acc	Accuracy
Dep	Dependability
Saf	Safety
Sen	sensibility
TP	True Positives
TN	True Negatives
FN	False Negatives
FP	False Positives
GSCV	Grid Search Cross-Validation
ANN	Artificial Neural Network
GRU-AE	Gated Recurrent Units Autoencoder

Appendix A

Table A1. IEEE 13 node test feeder Load Information.

Node	Load Model	Phase 1 (kW)	Phase 1 (kVar)	Phase 2 (kW)	Phase 2 (kVar)	Phase 3 (kW)	Phase 3 (kVar)
634	Y-PQ	160	110	120	90	120	90
645	Y-PQ	0	0	170	125	0	0
646	D-Z	0	0	230	132	0	0
652	Y-Z	128	86	0	0	0	0
671	D-PQ	385	220	385	220	385	220
675	Y-PQ	485	190	68	60	290	212
692	D-I	0	0	0	0	170	151
611	Y-I	0	0	0	0	170	80
	Total	1158	606	973	627	1135	735

Table A2. IEEE 13 node test feeder Line Length and phasing.

Node A	Node B	Length (ft)	Phasing
632	645	500	C, B, N
632	633	500	C, A, B, N
633	634	0	Transformer
645	646	300	C, B, N
650	632	2000	B, A, C, N
684	652	800	A, N
632	671	2000	B, A, C, N
671	684	300	A, C, N
671	680	1000	B, A, C, N
671	692	0	Switch
684	611	300	C, N
692	675	500	A, B, C, N

References

- Wang, B.; Geng, J.; Dong, X. High-Impedance Fault Detection Based on Nonlinear Voltage–Current Characteristic Profile Identification. *IEEE Trans. Smart Grid* **2018**, *9*, 3783–3791. [\[CrossRef\]](#)
- Gautam, S.; Brahma, S.M. Detection of High Impedance Fault in Power Distribution Systems Using Mathematical Morphology. *IEEE Trans. Power Syst.* **2013**, *28*, 1226–1234. [\[CrossRef\]](#)
- Cui, Q.; El-Arroudi, K.; Weng, Y. A Feature Selection Method for High Impedance Fault Detection. *IEEE Trans. Power Deliv.* **2019**, *34*, 1203–1215. [\[CrossRef\]](#)
- Ghaderi, A.; Ginn, H.L., III; Mohammadpour, H.A. High impedance fault detection: A review. *Electr. Power Syst. Res.* **2017**, *143*, 376–388. [\[CrossRef\]](#)
- Wang, X.; Gao, J.; Wei, X.; Song, G.; Wu, L.; Liu, J.; Zeng, Z.; Kheshti, M. High Impedance Fault Detection Method Based on Variational Mode Decomposition and Teager–Kaiser Energy Operators for Distribution Network. *IEEE Trans. Smart Grid* **2019**, *10*, 6041–6054. [\[CrossRef\]](#)
- Chaitanya, B.K.; Yadav, A.; Pazoki, M. An Intelligent Detection of High-Impedance Faults for Distribution Lines Integrated With Distributed Generators. *IEEE Syst. J.* **2020**, *14*, 870–879. [\[CrossRef\]](#)
- Wang, B.; Ni, J.; Geng, J.; Lu, Y.; Dong, X. Arc flash fault detection in wind farm collection feeders based on current waveform analysis. *J. Mod. Power Syst. Clean Energy* **2017**, *5*, 211–219. [\[CrossRef\]](#)
- Lee, I. *High-Impedance Fault Detection Using Third-Harmonic Current*; Final Report. Technical Report; Hughes Aircraft Co.: Malibu, CA, USA, 1982.
- Emanuel, A.E.; Cyganski, D.; Orr, J.A.; Shiller, S.; Gulachenski, E.M. High impedance fault arcing on sandy soil in 15 kV distribution feeders: Contributions to the evaluation of the low frequency spectrum. *IEEE Trans. Power Deliv.* **1990**, *5*, 676–686. [\[CrossRef\]](#)
- Jeerings, D.I.; Linders, J.R. Unique aspects of distribution system harmonics due to high impedance ground faults. *IEEE Trans. Power Deliv.* **1990**, *5*, 1086–1094. [\[CrossRef\]](#)
- Jeerings, D.I.; Linders, J.R. A practical protective relay for down-conductor faults. *IEEE Trans. Power Deliv.* **1991**, *6*, 565–574. [\[CrossRef\]](#)
- Yu, D.C.; Khan, S.H. An adaptive high and low impedance fault detection method. *IEEE Trans. Power Deliv.* **1994**, *9*, 1812–1821. [\[CrossRef\]](#)
- Milioudis, A.N.; Andreou, G.T.; Labridis, D.P. Detection and Location of High Impedance Faults in Multiconductor Overhead Distribution Lines Using Power Line Communication Devices. *IEEE Trans. Smart Grid* **2015**, *6*, 894–902. [\[CrossRef\]](#)
- Mortazavi, S.H.; Moravej, Z.; Shahrtash, S.M. A Searching Based Method for Locating High Impedance Arcing Fault in Distribution Networks. *IEEE Trans. Power Deliv.* **2019**, *34*, 438–447. [\[CrossRef\]](#)
- Sedighi, A.R.; Haghifam, M.R.; Malik, O.P.; Ghassemian, M.H. High impedance fault detection based on wavelet transform and statistical pattern recognition. *IEEE Trans. Power Deliv.* **2005**, *20*, 2414–2421. [\[CrossRef\]](#)
- Soheili, A.; Sadeh, J. Evidential reasoning based approach to high impedance fault detection in power distribution systems. *IET Gener. Transm. Distrib.* **2017**, *11*, 1325–1336. [\[CrossRef\]](#)
- Girgis, A.A.; Chang, W.; Makram, E.B. Analysis of high-impedance fault generated signals using a Kalman filtering approach. *IEEE Trans. Power Deliv.* **1990**, *5*, 1714–1724. [\[CrossRef\]](#)
- Chakraborty, S.; Das, S. Application of Smart Meters in High Impedance Fault Detection on Distribution Systems. *IEEE Trans. Smart Grid* **2019**, *10*, 3465–3473. [\[CrossRef\]](#)
- Xue, S.; Cheng, X.; Lv, Y. High Resistance Fault Location of Distribution Network Based on EEMD. In Proceedings of the 9th International Conference on Intelligent Human-Machine Systems and Cybernetics, Hangzhou, China, 26–27 August 2017; Volume 1, pp. 322–326.

20. Yao, W.; Gao, X.; Wang, S.; Liu, Y. Distribution High Impedance Fault Detection Using the Fault Signal Reconstruction Method. In Proceedings of the IEEE 3rd Conference on Energy Internet and Energy System Integration, Changsha, China, 8–10 November 2019; pp. 2573–2577.
21. Baqui, I.; Zamora, I.; Mazón, J.; Buigues, G. High impedance fault detection methodology using wavelet transform and artificial neural networks. *Electr. Power Syst. Res.* **2011**, *81*, 1325–1333. [\[CrossRef\]](#)
22. Tonelli-Neto, M.S.; Decanini, J.G.M.S.; Lotufo, A.D.P.; Minussi, C.R. Fuzzy based methodologies comparison for high-impedance fault diagnosis in radial distribution feeders. *Int. J. Electr. Power Energy Syst.* **2017**, *11*, 1557–1565. [\[CrossRef\]](#)
23. Moravej, Z.; Mortazavi, S.H.; Shahrtash, S.M. DT-CWT based event feature extraction for high impedance faults detection in distribution system. *Int. Trans. Electr. Energy Syst.* **2015**, *25*, 3288–3303. [\[CrossRef\]](#)
24. Wang, S.; Dehghanian, P. On the Use of Artificial Intelligence for High Impedance Fault Detection and Electrical Safety. *IEEE Trans. Ind. Appl.* **2020**, *56*, 7208–7216. [\[CrossRef\]](#)
25. Veerasamy, V.; Wahab, N.I.A.; Ramachandran, R.; Thirumeni, M.; Subramanian, C.; Othman, M.L.; Hizam, H. High-impedance fault detection in medium-voltage distribution network using computational intelligence-based classifiers. *Neural Comput. Appl.* **2019**, *31*, 9127–9143. [\[CrossRef\]](#)
26. Sheng, Y.; Rovnyak, S.M. Decision tree-based methodology for high impedance fault detection. *IEEE Trans. Power Deliv.* **2004**, *19*, 533–536. [\[CrossRef\]](#)
27. Veerasamy, V.; Wahab, N.I.A.; Othman, M.L.; Padmanaban, S.; Sekar, K.; Ramachandran, R.; Hizam, H.; Vinayagam, A.; Islam, M.Z. LSTM Recurrent Neural Network Classifier for High Impedance Fault Detection in Solar PV Integrated Power System. *IEEE Access* **2021**, *9*, 32672–32687. [\[CrossRef\]](#)
28. L’Heureux, A.; Grolinger, K.; Elyamany, H.F.; Capretz, M.A.M. Machine Learning With Big Data: Challenges and Approaches. *IEEE Access* **2017**, *5*, 7776–7797. [\[CrossRef\]](#)
29. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
30. Ghosh, A.; Grolinger, K. Edge-Cloud Computing for IoT Data Analytics: Embedding Intelligence in the Edge with Deep Learning. *IEEE Trans. Ind. Inform.* **2020**. [\[CrossRef\]](#)
31. Fan, C.; Xiao, F.; Zhao, Y.; Wang, J. Analytical investigation of autoencoder-based methods for unsupervised anomaly detection in building energy data. *Appl. Energy* **2018**, *211*, 1123–1135. [\[CrossRef\]](#)
32. Araya, D.B.; Grolinger, K.; Elyamany, H.F.; Capretz, M.A.; Bitsuamlak, G. An ensemble learning framework for anomaly detection in building energy consumption. *Energy Build.* **2017**, *144*, 191–206. [\[CrossRef\]](#)
33. Li, S.; He, H.; Li, J. Big data driven lithium-ion battery modeling method based on SDAE-ELM algorithm and data pre-processing technology. *Appl. Energy* **2019**, *242*, 1259–1273. [\[CrossRef\]](#)
34. Masci, J.; Meier, U.; Cireşan, D.; Schmidhuber, J. Stacked convolutional auto-encoders for hierarchical feature extraction. In Proceedings of the International Conference on Artificial Neural Networks, Espoo, Finland, 14–17 June 2011; Springer: Berlin/Heidelberg, Germany, 2011; pp. 52–59.
35. Hossain, M.S.; Al-Hammadi, M.; Muhammad, G. Automatic Fruit Classification Using Deep Learning for Industrial Applications. *IEEE Trans. Ind. Inform.* **2019**, *15*, 1027–1034. [\[CrossRef\]](#)
36. Ince, T.; Kiranyaz, S.; Eren, L.; Askar, M.; Gabbouj, M. Real-Time Motor Fault Detection by 1-D Convolutional Neural Networks. *IEEE Trans. Ind. Electron.* **2016**, *63*, 7067–7075. [\[CrossRef\]](#)
37. Wang, K.; Qi, X.; Liu, H. A comparison of day-ahead photovoltaic power forecasting models based on deep learning neural network. *Appl. Energy* **2019**, *251*, 113315. [\[CrossRef\]](#)
38. Gholamiangonabadi, D.; Kiselov, N.; Grolinger, K. Deep Neural Networks for Human Activity Recognition With Wearable Sensors: Leave-One-Subject-Out Cross-Validation for Model Selection. *IEEE Access* **2020**, *8*, 133982–133994. [\[CrossRef\]](#)
39. Chi, C.Y.; Feng, C.C.; Chen, C.H.; Chen, C.Y. *Blind Equalization and System Identification: Batch Processing Algorithms, Performance and Applications*; Springer Science and Business Media: Berlin/Heidelberg, Germany, 2006.
40. Kersting, W.H. Radial distribution test feeders. *IEEE Trans. Power Syst.* **1991**, *6*, 975–985. [\[CrossRef\]](#)
41. Soheili, A.; Sadeh, J.; Bakhshi, R. Modified FFT based high impedance fault detection technique considering distribution non-linear loads: Simulation and experimental data analysis. *Int. J. Electr. Power Energy Syst.* **2018**, *94*, 124–140. [\[CrossRef\]](#)
42. Kavi, M.; Mishra, Y.; Vilathgamuwa, M.D. High-impedance fault detection and classification in power system distribution networks using morphological fault detector algorithm. *IET Gener. Transm. Distrib.* **2018**, *12*, 3699–3710. [\[CrossRef\]](#)
43. Ghaderi, A.; Mohammadpour, H.A.; Ginn, H.L.; Shin, Y. High-Impedance Fault Detection in the Distribution Network Using the Time-Frequency-Based Algorithm. *IEEE Trans. Power Deliv.* **2015**, *30*, 1260–1268. [\[CrossRef\]](#)