



---

*Research article*

## **SunSpark: Fusion of time-domain and frequency-domain transformer for accurate identification of DC arc faults**

**Chunpeng Tian<sup>1</sup>, Zhaoyang Xu<sup>2</sup>, Yunjie Liu<sup>3</sup>, Lukun Wang<sup>1</sup> and Pu Sun<sup>1,\*</sup>**

<sup>1</sup> College of Intelligent Equipment, Shandong University of Science and Technology, Taian 271019, China

<sup>2</sup> University of Cambridge, Wellcome-MRC Cambridge Stem Cell Institute, Cambridge, England

<sup>3</sup> School of Communication Engineering, Taishan College of Science and Technology, Taian 271038, China

\* **Correspondence:** Email: [sunpu@sdust.edu.cn](mailto:sunpu@sdust.edu.cn).

**Abstract:** Photovoltaic (PV) power generation is pivotal to the energy strategies of various nations, yet it is plagued by significant security challenges. This paper proposes a large-scale neural network model that integrates time-domain and frequency-domain techniques for the detection of arc faults in PV systems. The algorithm leverages sequence decomposition to extract trend information from current signals, and then applies the Fourier transform to convert various encoded data into the frequency domain. Due to the sparsity of frequency-domain information, the computational cost of extracting and processing information in the frequency domain is minimal, resulting in high efficiency. The selectively extracted information is then input into a separate lightweight classifier for classification and recognition. The proposed intelligent framework not only effectively filters out high-frequency noise signals, but also demonstrates strong robustness against various disturbances, yielding exceptional recognition performance with an accuracy rate consistently surpassing 97%. Code and data are available at this repository: <https://github.com/yixizhuimeng?tab=projects>.

**Keywords:** signal processing; arc fault detection; artificial intelligence; time-domain and frequency-domain fusion

---

### **1. Introduction**

Energy is a fundamental infrastructure underpinning national development and has profound impacts on economic, environmental, and social progress. As energy consumption accelerates, traditional energy sources approach depletion [1, 2]. Moreover, greenhouse gases and environmental pollution resulting from traditional energy consumption significantly affect human quality of life. Renewable

energy sources, including solar PV and wind power, are emerging as vital solutions for energy security and sustainable development. Renewable energy grid integration systems are widely applied in both distributed power generation and residential power systems. Consequently, there is an increasing emphasis on the safety of renewable energy systems and inverters [3].

Solar PV power, characterized by its small footprint, ease of installation, and high power generation capacity, is globally popular and widely applied. In 2020, the addition of renewable energy capacity exceeded 256 gigawatts (GW), of which solar PV power contributed over half, reaching 139 GW. The total installed capacity for renewable energy reached 760 GW [4]. It is projected that by 2050, renewable energy will be a primary electricity source, providing approximately 11% of global electricity [5]. In the practical application of solar PV systems, direct current (DC) arc faults are among the most hazardous faults [6]. Electric arcs, a critical challenge in PV systems, are formed by the ionization of gases between two conductors, leading to a sustained plasma discharge [7]. This process typically occurs through a medium that is usually non-conductive, such as air. Electric arcs are not only hazardous due to their high temperature and intense brightness, but also because they can cause damage to electrical equipment and pose a fire risk [8]. In terms of electrical characteristics, electric arcs manifest as distinctive, periodic fluctuations in voltage and current, often accompanied by unique frequency components distinguishable from normal operational signals [9]. These properties of electric arcs make them both a critical area of study and a challenge for effective detection in PV systems.

Typically, PV systems commonly exhibit three types of arc faults: series arc faults (SAF), parallel arc faults (PAF), and ground arc faults (GAF). Parallel arc faults (PAF) and ground arc faults (GAF) are usually accompanied by observable signal alterations. Traditional detection apparatus can accurately capture these signal deviations to initiate appropriate responses [10]. Nevertheless, signal perturbations resulting from series arc faults (SAF) are subtle, and conventional threshold-based detection methods struggle to discern such nuanced changes. Consequently, this investigation concentrates on series arc faults (SAF) within PV systems. Furthermore, the unpredictable incidence of arcs and the pulse-width modulation (PWM) control utilized in PV inverters introduce high-frequency noise interference to the current, amplifying the complexity of arc detection tasks [11].

Researchers worldwide have conducted extensive research on direct current arc faults, yielding some results. Arc fault identification primarily involves four research directions: simulation models, arc light radiation, electrical signal fluctuations, and reasoning-based intelligent algorithms like neural networks. Most research involves modeling arc macro characteristics using physical and mathematical equations, considering arcs as variable resistors and calculating their equivalent impedance using nonlinear differential equations. Notable models include the Mayr model [12], the Cassie model [13], and the improved Schavemaker model [14]. Some researchers have improved traditional impedance models with dynamic models like diode models and hyperbolic models, elucidating arc-related effects through voltage and current relationships [15]. However, there are significant gaps in the existing literature, particularly regarding the real-world application of these models due to the complexity of equations and parameter limitations.

Physical phenomena-based detection methods, like arc light radiation, rely on capturing arc occurrences' physical phenomena using relevant instruments. Murakami et al. [16] observed arc light beams using high-speed cameras. Yue et al. [17] determined arcs by detecting high input capacitance-caused intermittent discharges at the interface. Xiong et al. [18] analyzed arc signals' electromagnetic radiation signals using a fourth-order Hilbert curve. Generally, these methods have successfully detected

arcs. However, the randomness of arc occurrence positions limits their large-scale application. Electrical signal fluctuation-based detection methods involve studying and analyzing the strong voltage and current changes occurring when an arc occurs. From a data processing perspective, these methods are intricately divided into three dimensions: time domain, frequency domain, and time-frequency domain. While these traditional methods have shown some success in detecting arcs, their practical application faces limitations due to the inherent randomness of arc occurrence and the variability in arc characteristics. Compared to previous approaches, our work stands out by providing a more holistic and robust solution for arc detection, leveraging the strengths of both time-domain and frequency-domain analysis. Our method overcomes the limitations of traditional methods by efficiently processing the unique electrical signatures of arcs, thus offering a more reliable and scalable solution for arc fault detection in PV systems.

The method based on changes in the telecommunication signal has been widely applied due to its simple implementation. Hastings et al. [19] determined arcs by comparing different current peak signals. Gu et al. [20] conducted arc frequency domain characteristic analysis using the Fast Fourier Transform (FFT). Frequency domain analysis alone cannot determine the exact arc occurrence time. Therefore, some researchers combine time-domain and frequency-domain analysis. Liu et al. [21] used Variational Mode Decomposition (VMD) to fuse time-domain and frequency-domain signals, enhancing the arc detection algorithm's resistance to interference. Wang et al. [22] and Chen et al. [23] conducted a multi-resolution analysis of arc signals in the time-frequency domain using wavelet transforms. These methods significantly improved arc detection accuracy but are limited by the need for manually set thresholds. There are no universally applicable discrimination criteria for current and voltage in different scenarios, hence the need for more effective arc fault detection methods.

Recently, with the upgrade in computing resources, neural networks have been widely applied in various fields due to their powerful learning and recognition capabilities [24,25]. Li et al. [26] proposed using a backpropagation-based neural network for arc detection. Yang et al. [27] converted filtered arc data into grayscale images as Convolutional Neural Network (CNN) inputs for arc classification. Lu et al. [28] suggested a combined approach using domain adaptation and a Deep Convolutional Generative Adversarial Network (DA-DCGAN) for arc detection, achieving promising results. Wang et al. [29] processed power spectra using a Convolutional Neural Network and proposed a lightweight Efficientnet-B1 model. Data-driven artificial intelligence algorithms that continuously collect data from new scenarios to reconfigure model parameters show great promise in the field of arc fault detection [30–32]. However, there are challenges that hinder its full potential [33, 34].

In the field of arc fault detection based on deep learning, there are three significant challenges: 1) Lack of a common dataset: currently, there is no publicly available dataset for arc fault detection. This means that researchers need to set up their own arc detection experimental platforms and collect data before conducting relevant studies. This process can be time-consuming and leads to difficulties in effectively comparing new models developed by different researchers. 2) Limited scope of learnable neural networks: existing learnable neural networks are often limited to specific scenarios, lacking a large-scale, generalized neural network architecture that can handle multiple scenarios effectively. 3) Time-series analysis for arc prediction: arc occurrences follow a timeline, and the processes before and after the arc event contain rich time-series information. This information can be utilized for predicting arc occurrence trends through time-series analysis.

The main contributions of this article, supported by subsequent sections, are as follows:

1) Provision of a large standard dataset. We create a large standard dataset comprising ten thousand data points, including various loads such as resistors, capacitors, and inductors. This dataset will assist researchers in conducting experiments and model evaluations in the field of arc fault detection.

2) Introduction of data augmentation strategy. We employ a classical time-series decomposition method. This deep-level encoding input strategy is expected to enhance the model's feature extraction capability, as demonstrated in our experiments.

3) Fusion of time and frequency domain information. Our approach combines time-domain and frequency-domain data, utilizing the Fourier transform for feature extraction. This method simplifies the attention mechanism, capturing both global and local information, as shown in the results.

4) Efficient classifier. We introduce an efficient fast mapping classifier, outperforming traditional neural networks in classification tasks, as evidenced in our comparative studies.

The rest of this article is organized as follows. Section 2 introduces related work. Section 3 presents the proposed model, with a focus on highlighting the improved model's key components. Section 4 discusses the evaluation methods and experimental tests. Finally, the experimental findings are explained, scientific contributions summarized, and future research directions indicated.

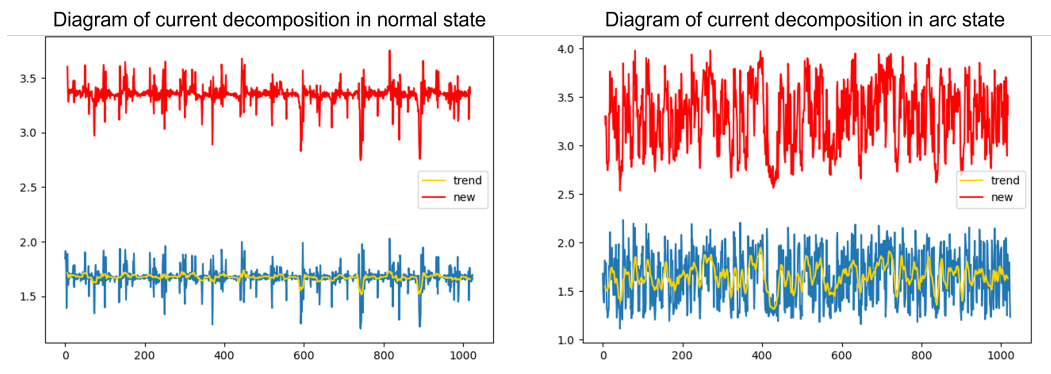
## 2. Related work

The lifecycle of an arc can be broadly divided into four phases: normal, arcing, stable burning, and extinguished states. As per the UL1699B standard, it is imperative to perform arc fault detection within the arcing phase. Throughout an arc's entire lifecycle, subsequent stages are chronologically dependent on preceding ones, sharing similar components. This process essentially encapsulates a time-series relationship. Unlike other sequential data types, such as language or video, time-series data is recorded continuously, with each point saving only some scalars. Since a single time point typically cannot provide sufficient semantic information for analysis, many studies focus on temporal variations, which are more informative and can reflect the inherent properties of time series, such as continuity, periodicity, and trend. The Transformer model [35] has shown exceptional performance in sequence processing in recent years, thanks to its self-attention mechanism. It has yielded remarkable results across various domains, including natural language processing [36], audio processing [37], and motion analysis [38]. The superior capability of the Transformer architecture primarily stems from its multi-head attention mechanism, which excels at capturing correlations in long sequences. However, it is crucial to note that self-attention mechanisms come with  $O(L^2)$  complexity in terms of both memory and time for sequences of length  $L$ , which can be highly unfriendly. LogTrans [39] employs causal convolutions to incorporate features into the attention mechanism, reducing the complexity to  $O(L(\log L)^2)$ . Informer [40] utilizes a ProbSparse self-attention mechanism based on the KL divergence to reduce the complexity to  $O(L \cdot \log L)$ . Reformer [41] replaces dot-product attention with local hashing attention, reducing the complexity from  $O(L^2)$  to  $O(L \cdot \log L)$ . Autoformer [42] introduces a sequence-based attention mechanism, achieving  $O(L \cdot \log L)$  complexity. Pyraformer [43] introduces a multi-resolution pyramid attention mechanism for modeling long-range dependencies and time series prediction, reducing both time and space complexity to  $O(L)$ . FEDformer [44] leverages frequency-domain information to represent sparsity and designs a frequency-enhanced Transformer, reducing complexity to linear.

### 3. Method

#### 3.1. The data augmentation module for sequence decomposition

In actual arc sampling data, the arc data's behavior exhibits a wider range compared to normal data. The normal data range is encompassed by the arc data, making it challenging to separate normal data for detection. Due to the evident temporal correlations in arc data, we introduce a time-series decomposition approach for handling arc recognition tasks. It is crucial to highlight that the actual arc noise, influenced by different loads, can not be ignored. Our method takes into account the noise and discrepancies introduced by these varying loads, ensuring a closer alignment with real-world arc characteristics. Time-series decomposition aids in understanding the characteristics of time-series data, enabling better detection. In the STL (Seasonal and Trend decomposition using LOESS) time series decomposition algorithm [45], the original time series data is separated into several constituent parts, each representing distinct patterns or categories that can potentially be predicted. Usually, it includes the trend component, seasonal component, and residual, representing the long-term trend, repetitive cycles, and irregular fluctuations, respectively, within a time series. Figure 1 illustrates the current decomposition in the normal and arc state.



**Figure 1.** Normal current and abnormal current decomposition schematic diagram.

$$\mathbf{X}_t = \hat{S}_t + \hat{T}_t + \hat{R}_t \quad (3.1)$$

where  $\hat{S}_t$  represents seasonal information,  $\hat{T}_t$  represents trend cycle information, and  $\hat{R}_t$  represents residual information. Inspired by Zhou et al. [44], we apply the concept of sequence decomposition to process the arc data.

In the seasonal-trend decomposition, for an input sequence of length  $L$ , the trend component of the time series  $x$  ( $x \in \mathbb{R}^{L \times d}$ ), denoted as  $t$ , is obtained by taking the average of the sum of sequence values over  $k$  periods. This process is referred to as moving average smoothing, and the procedure is as follows:

$$\hat{T}_t = \frac{1}{m} \sum_{j=-k}^k x_{t+j} \quad (3.2)$$

where  $m = 2k + 1$ ,  $\hat{T}t$  represents the trend component value at time  $t$ , and  $x_{t+j}$  represents the value of the time series at the  $j$ th position relative to time  $t$ , with  $t$  as the symmetric center.

Through the process of moving average smoothing, some of the randomness in the original time series data is eliminated, while the smoothed trend component is retained, resulting in the creation of a new sequence. He et al. [46] employed a residual approach to transmit the original information directly to deeper layers, ensuring the sufficiency of learnable information in neural networks. Inspired by this concept, we use the decomposed trend components of the original time series as carriers to transfer information to deeper layers. Compared to directly transmitting the raw sequence information, the trend components contain purer sequence information, which significantly reduces the blind feature extraction inherent in using convolutional methods.

This decomposition, especially the extraction of the trend component, plays a vital role in arc detection. The trend component helps in isolating the consistent and longer-lasting patterns in the data, which can be indicative of the arc's presence.

### 3.2. Frequency domain information processing module based on Fourier transform

The Fourier transform is an effective tool for mapping information from the time domain to the frequency domain, decomposing a time-domain signal into a superposition of different frequencies of sine or cosine waves. The mathematical formula is as follows:

$$F(\omega) = \int f(t)e^{-i\omega t} dt \quad (3.3)$$

where  $f(t)$  is the time-domain function,  $F(\omega)$  is the representation of the function in the frequency domain,  $i$  is the imaginary unit, and  $\omega$  is the frequency. In a time series, the current value not only depends on past values, but also exhibits some form of dependency on future values, showing mutual interdependence. A time series signal is transformed into its corresponding signal spectrum through the Fourier transform, where the amplitude and phase information of each frequency component collectively represent the characteristics of the original signal at that frequency. This allows us to obtain global information such as the main frequency component and frequency distribution range. On the other hand, due to the temporal correlations in time series signals, there are highly similar components between data signals. The frequency domain information matrix obtained using the Fourier transform has low rank, theoretically reducing the complexity of the input signal.

Assuming the input current signal is  $x(x \in R^{N \times D})$ , it is first subjected to a linear projection using the matrix  $W(W \in R^{D \times D})$  to obtain the  $q(q \in R^{N \times D})$  information matrix. After processing with the Fourier function, the time-domain information matrix  $q$  is transformed into the frequency-domain information matrix  $Q(Q \in R^{N \times D})$ . Due to the similarity in the frequencies of the main components in the frequency-domain information, we select a certain number of frequencies using random sampling as the main frequencies for the entire information. Compared to fixed sampling, random sampling allows for a more comprehensive consideration of both low-frequency and high-frequency component characteristics.

$$\tilde{Q} = \text{Sel}(Q) = \text{Sel}(F(q)) \quad (3.4)$$

Where  $\tilde{Q} \in C^{M \times D}$ , we limit  $M \ll N$ , significantly reducing the computational complexity. We initialize a set of random parameter matrices  $R(R \in C^{D \times D \times M})$ , where the first  $D$  represents the input

channel and the second D represents the output channel. We define Y as the result of the operation between Q and R, represented as follows:

$$\widetilde{Y} = \widetilde{Q} \odot R \quad (3.5)$$

where  $\widetilde{Y} \in C^{M^*D}$ . To restore the obtained information using zero-padding to the same length as information Y ( $Y \in R^{N^*D}$ ), we then use the inverse Fourier transform to convert the frequency-domain information back to the time domain. The specific process is represented as follows:

$$Fe(q) = F^{-1}(\text{Padding}(\widetilde{Q} \odot R)) \quad (3.6)$$

where  $F^{-1}(x)$  represents the inverse Fourier transform, and  $Fe(q)$  represents the time-domain information.

Transforming the signal to the frequency domain enables the model to detect arc-specific frequency components, which might be otherwise masked in the time-domain data. This transformation is crucial as arcs generate characteristic frequency signatures that can be isolated and detected more effectively in the frequency domain.

### 3.3. Frequency domain attention mechanism based on the Fourier transform

For the input information's queries, keys, and values matrices, they can be represented as  $q \in R^{L^*D}$ ,  $k \in R^{L^*D}$ ,  $v \in R^{L^*D}$ , where  $q$ ,  $k$ ,  $v$  are obtained by multiplying the input information  $x$  with their respective matrices. The formulas are as follows:

$$q = x \bullet w_q \quad (3.7)$$

$$k = x \bullet w_k \quad (3.8)$$

$$v = x \bullet w_v \quad (3.9)$$

where  $w_v, w_k, w_q \in R^{L^*D}$ . The standard attention is represented as:

$$\text{Atten}(q, k, v) = \text{Softmax}\left(\frac{qk^T}{\sqrt{d_q}}\right)v. \quad (3.10)$$

In the frequency domain attention based on the Fourier transform, we first perform Fourier transform on the original  $q$ ,  $k$ ,  $v$  sequence data information. Then, we randomly select M frequency components as the primary features of the information sequence for the attention mechanism calculation. The former are represented as  $\widetilde{Q} \in C^{MD}$ ,  $\widetilde{K} \in C^{MD}$ ,  $\widetilde{V} \in C^{M^*D}$  respectively. The Fourier-based frequency domain attention mechanism can be represented as:

$$\widetilde{Q} = \text{Sel}(F(q)) \quad (3.11)$$

$$\widetilde{K} = \text{Sel}(F(k)) \quad (3.12)$$

$$\widetilde{V} = \text{Sel}(F(v)) \quad (3.13)$$

$$F(q, k, v) = F^{-1}(\text{Padding}(\partial(\widetilde{Q} \bullet \widetilde{K}) \bullet \widetilde{V})) \quad (3.14)$$

where  $\partial$  is the activation function, and we use softmax as the activation function. Zero-padding is performed before executing the inverse Fourier transform.

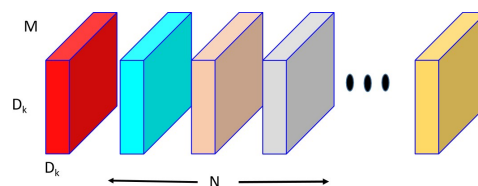
The introduction of the attention mechanism in the frequency domain helps the model to focus on specific frequency components which are more indicative of arcs. This targeted attention ensures that the model gives more importance to frequencies which are more likely to be associated with arcs, improving the detection accuracy.

### 3.4. Efficient mapping classification head

Unlike the traditional Transformer architecture used for downstream tasks, we use the Transformer solely as a feature extractor. Traditional convolutional methods for feature extraction can result in localism and redundancy in the extracted features due to variations in convolutional kernels and strides. In contrast to the traditional convolution-based feature extraction approach, Transformers inherently possess the capability to extract global features by computing importance scores for different features through attention mechanisms. This approach avoids the redundancy and locality issues associated with convolutional methods.

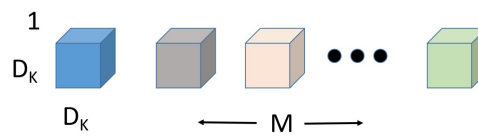
Fedformer [44] and others directly perform downstream MLP tasks after the Transformer encoding layer. This approach results in significant waste of the importance calculation performed by the Transformer layer as it places both important and unimportant features on a similar scale for downstream tasks. This suppresses the proportion of highly important features. Instead, we use a specific convolutional network to perform convolutional operations on the features extracted by the Transformer layer in order to extract important features for downstream classification tasks.

MobileNet is a type of lightweight convolutional neural network that employs depth-wise separable convolutions. It maintains high recognition accuracy while keeping the parameter count and computational load low. We drew inspiration from this approach and introduced depth-wise separable convolutions in the task of arc detection, achieving notable performance improvements. The essence of depth-wise separable convolution is to confine the receptive field of standard convolution within a single channel. In other words, all information within a specific channel is processed by a single convolutional kernel. Convolution kernels with the same number of channels are then used to filter information across multiple channels. Afterward, point-wise convolution is employed to balance and synergize information between channels, resulting in new features. As shown in Figures 2 and 3, compared to standard convolution, depth-wise convolution primarily focuses on a single channel. After aggregating the features extracted from a single channel,  $1 \times 1$  point-wise convolution (Figure 4) is used to synergize the information within the channel. This significantly contributes to reducing parameter computation and extracting feature information for the Transformer layer.

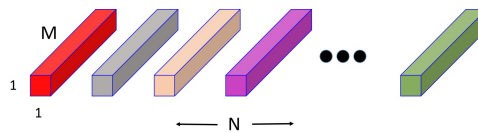


**Figure 2.** Standard convolution filters.





**Figure 3.** Depthwise convolutional filters.



**Figure 4.**  $1 \times 1$  Convolutional filters called pointwise convolution in the context of depthwise separable convolution.

This tailored convolutional approach ensures that the subtle arc-related features extracted by the Transformer are not lost and are further refined for the classification task. By focusing on channel-specific information using depth-wise convolution, we ensure that the unique characteristics of arcs are captured and used effectively for detection.

#### 4. Experiment implements

In order to evaluate the proposed SunSpark model, which is based on time-frequency fusion, we conducted experimental comparisons with several typical models, including the Transformer with self-attention mechanism [35], LogTransformer using causal convolution [39], Informer based on KL divergence [40], Reformer with local hashing attention [41], Autoformer [42], Pyraformer [43], FEDformer [44], and others.

##### 4.1. Implementation details

The experimental evaluation was conducted utilizing the PyTorch machine learning framework on a dedicated Windows workstation. The system was equipped with an Intel(R) Core(TM) i7-6800K CPU operating at a frequency of 4.2 GHz, backed by 32 GB of RAM for handling extensive data loads. The computational processes were accelerated using an NVIDIA GeForce GTX 1080 Ti GPU.

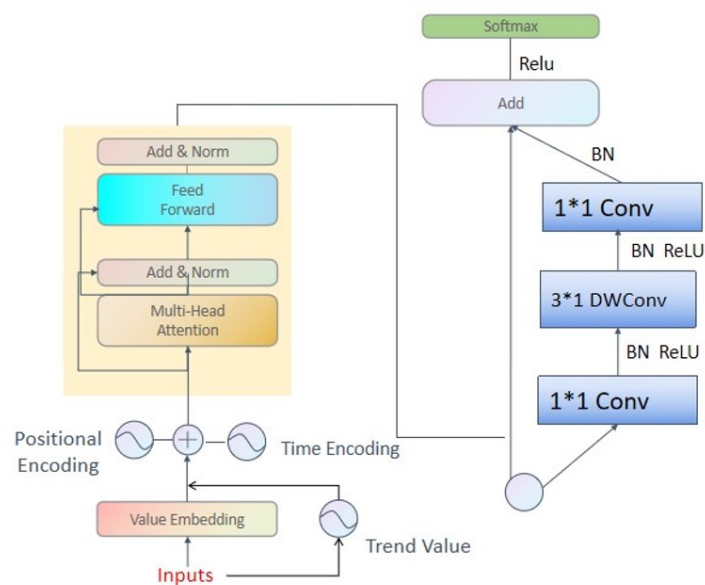
##### 4.2. Standardized unified arc dataset

The self-collected arc dataset was obtained using a self-sampling device based on electromagnetic induction principles. This device captured current data at a dense frequency of 4 million samples per second under various voltage and load conditions. It is critical to note that different loads can significantly influence the characteristics of arc faults. In our dataset, we considered a variety of loads that might mask arcs, potentially leading to false positives in arc detection. This comprehensive approach ensures that our model is trained with diverse scenarios, reflecting real-world complexities.

Due to the presence of 4 million data points in each record, the dataset became exceedingly large, posing a significant burden on existing deep learning hardware. To mitigate this, we sliced each ini-

tially collected dataset, ensuring that the accuracy of the original data was preserved while reducing the burden on data processing equipment. The entire dataset was divided into six sections based on the integrated devices in the circuit: 1uF capacitor, 10uF capacitor, 80uH inductor, one set of connected resistors, seven sets of parallel-connected resistors, and no connected resistors. We conducted arc current data collection experiments at voltages ranging from 100 V to 300 V for each load configuration. To address the concerns of low-quality data or noise, we conducted rigorous preprocessing. We promptly corrected the collected data using visualization tools. Some of the data collected at specific voltage levels were removed due to excessive external noise interference, while data with significant state transitions were retained.

### 4.3. Time-Frequency Domain Fusion Transformer Network



**Figure 5.** SunSpark model network.

Incorporating the insights from our time-series decomposition, our Time-Frequency Domain Fusion Transformer Network aims to capture the nuanced patterns in both time and frequency domains. The model integrates the decomposed trend components from the original time series as carriers to transfer information into deeper layers, enhancing the model's capability to discern intricate patterns and improving its performance.

The network (Figure 5) we designed primarily consists of a Sequence Decomposition Enhancement Module, Frequency Domain Information Processing Module, Encoder Layers, and a Classifier. The input current information, 'x', is first subjected to sequence decomposition to obtain the trend component 'T'. Subsequently, we employ a residual connection, which is a vital architectural innovation in deep learning, allowing the model to learn identity functions that expedite the training process and enable the stacking of more layers without the network performance degrading. This residual connection is employed to combine the data, encoded values, and the trend information 'T' to enhance the sequential time-domain information 'x'. This process extracts the trend information from the sequential data and accumulates it with the original information, enhancing important details while preserving the

integrity of the original data. The newly obtained information is then individually subjected to position encoding and time encoding. The resulting encoded information is fused and serves as the input to the Frequency Domain Information Processing Module. This module primarily involves filtering out unnecessary information from the input encoded data. This is achieved by performing a Fourier transform on the time-domain encoded data to obtain frequency-domain feature components corresponding to each piece of information. Due to the similarity of the frequency-domain components, we randomly select a subset of components from the feature component set and handle the remaining information through padding. The attention mechanism in the Encoder Layers primarily utilizes frequency-domain attention. It processes the input information to obtain the corresponding Q, K, and V for the attention layer through matrix operations. Similar to the Frequency Domain Information Processing Module mentioned earlier, these Q, K, and V representations are then mapped using Fourier transforms.

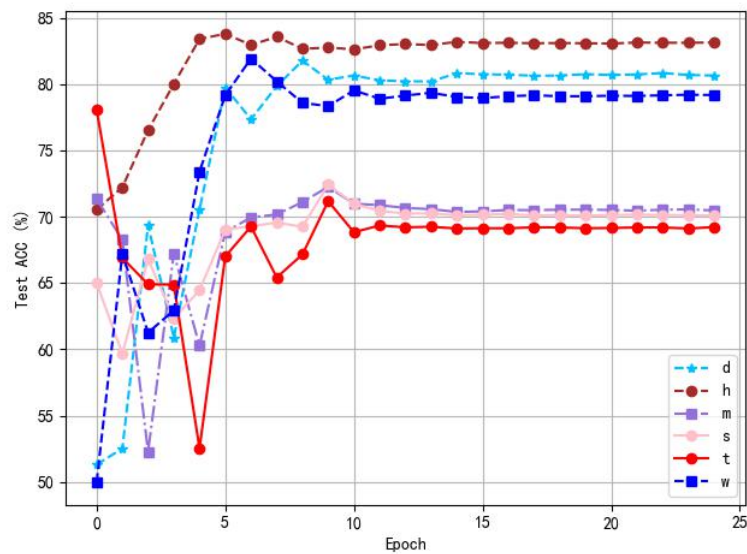
In order to reduce computational complexity and focus on essential information, a fixed number of feature components are selected at random from the mapped frequency-domain information for attention mechanism calculations. This significantly reduces unnecessary dot product computations, thereby greatly decreasing computational complexity.

After the dot product calculations of the attention mechanism in the Encoder Layers, multi-dimensional features of the current input data are extensively extracted. These extracted features are then fed into our designated classifier. The classifier primarily consists of 13 depth-wise separable convolutions, with each depth-wise separable convolution including both depth-wise and point-wise convolutions. We adapted the 3 x 3 depth-wise convolutional kernel from MobileNet to 3 x 1 to suit our current detection task. Additionally, a point-wise convolution is applied between channels before the depth-wise convolution, which helps filter inter-channel information for the subsequent depth-wise convolution. Finally, there is another point-wise convolution in the depth-wise separable convolution unit to balance local information within a single channel.

#### *4.4. Encoding experiment comparison*

We conducted experiments on different modes under time encoding as well as the trend component information obtained through sequence decomposition, separately. The purpose of conducting this experiment was to find a time representation method that is similar to the current sequence. After comparison, the time encoding method based on hourly intervals, as shown in Figure 6, aligns with the characteristics of our data. In the figure, 's' represents secondly, 't' represents minutely, 'h' represents hourly, 'd' represents daily, 'w' represents weekly, and 'm' represents monthly.

From Table 1, it is evident that hourly (h) time encoding method yields the highest accuracy of 84%. This is significantly higher compared to other methods, suggesting that hourly intervals are the most suitable time granularity for representing our data in the context of this study. The daily (d) and weekly (w) intervals also provide relatively high accuracy, indicating that these granularities capture some essential patterns in the data. In contrast, the methods based on seconds (s), minutes (t), and monthly (m) intervals perform poorly, suggesting that these time granularities might either miss critical patterns or introduce unnecessary noise into the data representation.



**Figure 6.** Comparative experiment of time encoding embedding methods.

**Table 1.** Comparative experiment results of time encoding embedding methods.

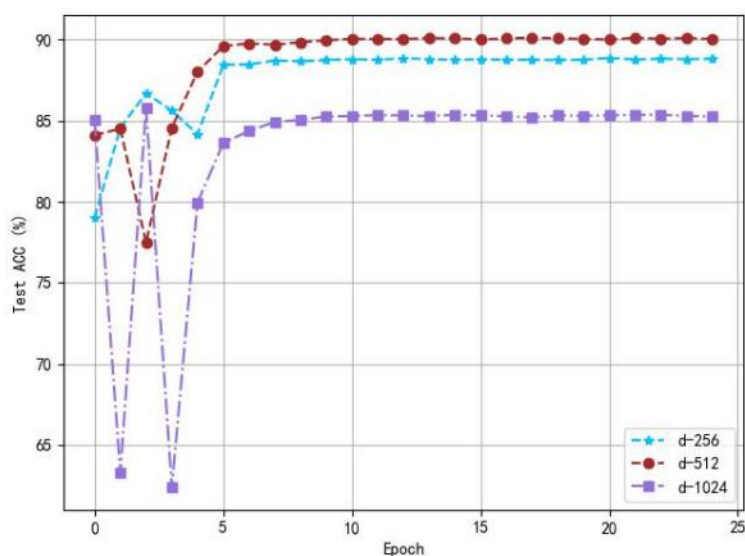
Embedding method	Test accuracy (%)
d	80.8
h	84
m	70.5
s	70
t	69
w	79

After selecting a suitable time encoding method, we made minor adjustments to the overall network architecture, including configuring the dimension of the model (Figure 7).

From Table 2, it can be observed that the model with a dimension of 512 offers the highest accuracy of 90.1%. This suggests that, while increasing the model's dimensionality can capture more intricate patterns, there is an optimal point beyond which the performance might degrade due to overfitting or increased computational complexity.

**Table 2.** Comparison experiment results of encoding dimension settings.

D_model	ACC (%)
256	88.7
512	90.1
1024	85.4

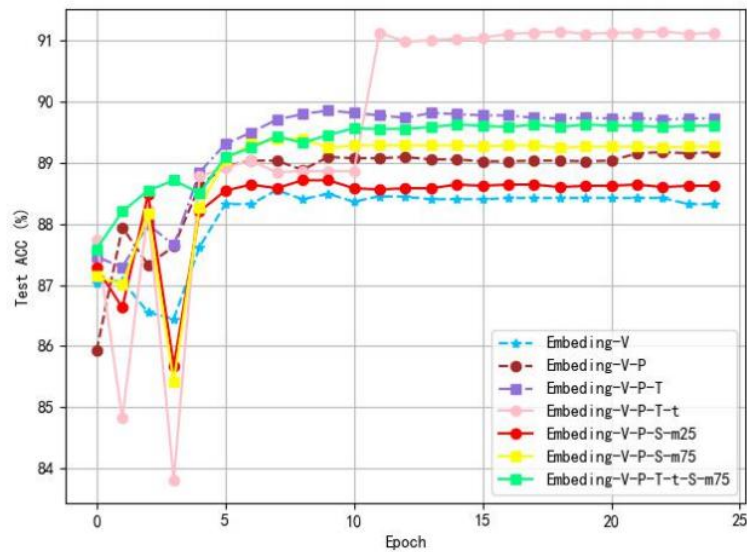


**Figure 7.** Comparison experiment of encoding dimension settings.

An essential aspect of SunSpark’s evaluation was understanding its performance across various operating points. When subjected to different environmental conditions, such as increased noise levels or varying arc types, the model consistently demonstrated exceptional robustness. Particularly, our experiments in the ‘Trend Component Decomposition’ and ‘Time-Frequency Domain Fusion Transformer Network’ highlighted SunSpark’s capability to discern intricate patterns across varying conditions. However, as depicted in our ‘Encoding Experiment Comparison’, when the encoding method was based on finer granularities like seconds (s) or minutes (t), there was a slight dip in performance. This offers valuable insights for further refinement and indicates areas where additional training or feature engineering might be beneficial.

#### 4.5. Trend component decomposition experiment

Time-series data, especially in the context of arc fault detection, often contains underlying patterns or trends that can provide valuable insights. The trend component of a time-series data captures its long-term movement. Specifically, it reflects the consistent and long-lasting increase or decrease in the data. Extracting and analyzing these trend components can offer a clearer perspective of the underlying patterns in the data, devoid of noise or short-term fluctuations. In our approach, we place significant emphasis on extracting and utilizing these trend components to enhance our model’s ability to identify arc faults accurately.



**Figure 8.** Trend component decomposition experiment.

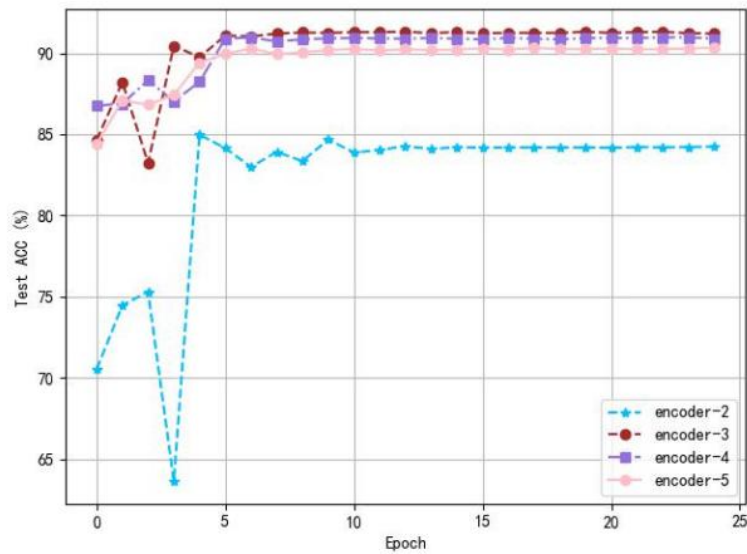
**Table 3.** Trend component decomposition experiment results.

Embedding	ACC (%)
value	88.4
Value+position	89.3
Value+position+Temporal	89.7
V+P+T+trend	91.1
V+P+T+seasonal (moving = 25)	88.6
V+P+T+seasonal (moving = 75)	89.3
V+P+T+trend+seasonal (moving = 75)	89.6

As previously elaborated, the trend component plays a pivotal role in understanding the underlying patterns of time-series data. Figure 8 illustrates that both position encoding and time encoding noticeably enhance recognition accuracy (Table 3). Incorporating trend component information results in a more substantial improvement in accuracy. Moreover, experiments that involved decomposing periodic components and adjusting the sliding factor parameter confirmed that periodic component information adversely affects recognition performance.

#### 4.6. Comparison experiment of encoding layer numbers

Prior to conducting internal experiments within the encoding layers, we assessed the number of encoding layers, as shown in Figure 9 and Table 4. To maintain a lightweight model for training and implementation, we limited the experiments to five encoding layers. This approach was adopted to avoid the excessive computational overhead associated with a higher number of layers.



**Figure 9.** Comparison experiment of encoding layer numbers.

**Table 4.** Comparison experiment results of encoding layer numbers.

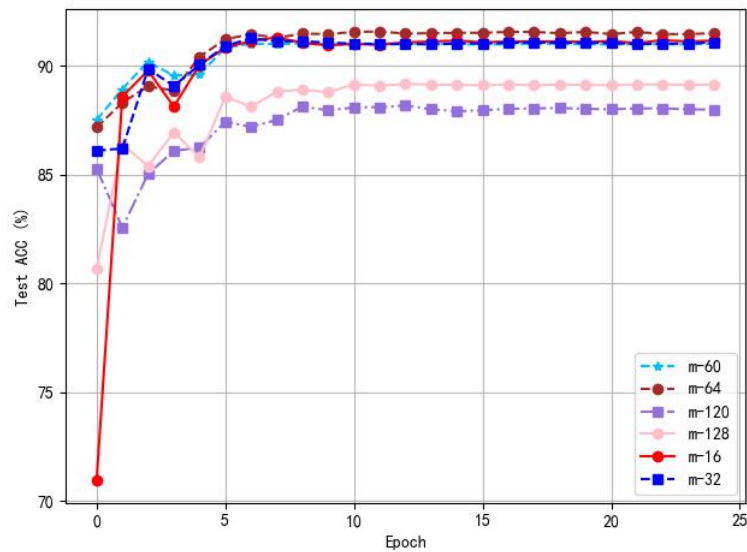
Encoder-layers	Acc (%)
Two-layers	84.6
Three-layers	91.3
Four-layers	90.8
Five-layers	90.3

#### 4.7. Selection of frequency domain components in Fourier mode

In this experiment, current signals underwent a Fourier transformation, transitioning from the time domain to frequency domain data. Given the inherent low-rank and sparsity characteristics of signals in the frequency domain, we conducted experiments to determine an optimal number of frequency domain components that would represent the majority of information while minimizing computational cost, as shown in Table 5 and Figure 10.

**Table 5.** Selection of frequency domain components in Fourier mode.

Select model	ACC (%)
Model = 60	91.0
Model = 64	91.5
Model = 120	88.0
Model = 128	89.1
Model = 16	91.1
Model = 32	91.0



**Figure 10.** Selection of frequency domain components in Fourier mode.

#### 4.8. Comparison of efficiency among different classifiers

In this paper, we introduce a comparative analysis between lightweight networks, specifically MobileNet, and traditional, more complex neural network architectures. This analysis focuses on key performance metrics including inference time, computational efficiency, and accuracy. Traditional networks, known for their deep and complex structures, offer high accuracy but often at the cost of increased computational resources and longer inference times. In contrast, MobileNet, a lightweight network, is designed to reduce computational demand while maintaining a balance between accuracy and efficiency. This makes it particularly suitable for applications where resource constraints are a critical factor.

In our comparative analysis within Section 4.8, we specifically evaluate the efficiency of different classifiers, including lightweight networks like MobileNet, in the context of real-time arc fault detection. Our findings challenge the traditional belief that increased complexity and a higher parameter count inherently translate to superior performance. Lightweight networks, particularly MobileNet, exhibit competitive performance, especially in terms of speed and resource utilization, which are critical for real-time applications. For instance, MobileNet demonstrated a precision of 0.9709, recall of 0.9613, and F1 score of 0.9661, outperforming other classifiers like Efficientnet (precision: 0.8906, recall: 0.8742, F1: 0.8729), and Squeeze Net (precision: 0.8671, recall: 0.8670, F1: 0.8670).

These metrics underscore the efficiency of MobileNet, attributable to its architecture, which is optimized for rapid processing and reduced computational demand without significantly compromising the quality of feature extraction. Moreover, when comparing inference time (IT), MobileNet showed a remarkable speed of 13 milliseconds (ms), which is faster than most of the traditional models like Informer (16 ms) and Transformer (18 ms). This efficiency makes MobileNet an ideal choice for our application, providing a balanced trade-off between speed, accuracy, and computational resource utilization.

Contrary to traditional beliefs, our findings indicate that lightweight networks, while typically less



complex and possessing fewer parameters than traditional networks, can still offer efficient feature extraction capabilities in specific contexts. In our study, lightweight networks such as MobileNet showed competitive performance, particularly in speed and resource utilization, crucial for real-time arc fault detection. This efficiency is partly due to their architecture, optimized for rapid processing and reduced computational demand, without significantly compromising the quality of feature extraction, as evidenced by the metrics in Table 6.

**Table 6.** Comparison of efficiency among different classifiers.

Classification head	Precision	Recall	F1
No addition	0.6424	0.6424	0.6420
Efficientnet	0.8906	0.8742	0.8729
Mobile Net	0.9709	0.9613	0.9661
Shuffle Net	0.9751	0.9518	0.9633
Squeeze Net	0.8671	0.8670	0.8670

#### 4.9. Comprehensive comparative experiment of large models

SunSpark demonstrates excellent recognition characteristics for time-series data, such as arc sequences. Compared to other models, its time-frequency domain transformation substantially reduces the complexity of feature extraction. To conduct a meaningful comparison between SunSpark and existing models, we performed comparative experiments with models in the relevant field. The recorded results are as follows.

**Table 7.** Comprehensive comparative experiment of large models.

Models	Precision	Recall	F1	Train_loss	Val_acc	Val_loss	IT (ms)
Autoformer	0.8921	0.8870	0.8866	0.0936	0.8869	0.491	12
Informer	0.8883	0.8599	0.8574	0.3489	0.8600	0.4398	16
Pyraformer	0.9015	0.8990	0.8989	0.1768	0.8990	0.1799	15
Reformer	0.8802	0.8560	0.8537	0.2825	0.8560	0.3179	10
TimesNet	0.7036	0.6995	0.6979	0.5838	0.6994	0.5749	11
Transformer	0.9004	0.9002	0.9002	0.1301	0.9002	0.2099	18
SunSpark (Ours)	0.9709	0.9604	0.9661	0.0039	0.9663	0.1006	13

As illustrated in Table 7, our model attained the highest scores across various evaluation metrics. Compared to the inference-focused model, Informer, our model exhibited a 9% improvement in recognition capability while significantly reducing computational complexity.

## 5. Conclusions

This article successfully outlines a cost-effective, rapid-detection, and high-accuracy universal arc fault diagnosis method by examining the time-frequency domain characteristics of PV current signals in normal and fault states. The proposed model leverages time series decomposition to extract trend information, enhancing the continuity of information throughout the time sequence. The enhanced

information undergoes transformation from the time domain to the frequency domain via the Fourier transform, and essential features are extracted in the frequency domain. This process considerably reduces the computational cost of Transformer-like large models while preserving the ability to extract robust features. Ultimately, the information transformed back into the time domain is processed by a lightweight classifier for arc fault diagnosis. Experimental results indicate that, at a high-frequency signal sampling rate, this diagnostic model effectively filters out high-frequency noise signals while maintaining an accuracy rate exceeding 97%, representing an improvement of over 7%.

As part of our future work, we aim to explore the practical deployment of our proposed technique in real-world scenarios. This includes assessing the feasibility of implementing our method on low-cost, low-power devices or integrating it with existing infrastructure. We plan to conduct comprehensive studies to evaluate the performance of our technique when deployed on various types of devices, ranging from embedded systems to portable diagnostic tools. This will not only validate the effectiveness of our technique in diverse field conditions, but will also provide insights into its adaptability and scalability in different operational environments. Moreover, the execution time of the designed model surpasses that of some traditional algorithms. Consequently, enhancing the execution speed of this algorithm is another crucial focus for future research.

### Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

### Acknowledgments

The authors declare there is no project support for this article.

### Conflict of interest

The authors declare there is no conflict of interest.

### References

1. I. Colak, H. Wilkening, G. Fulli, J. Vasiljevska, F. Issi, O. Kaplan, Analysing the efficient use of energy in a small smart grid system, in *2012 International Conference on Renewable Energy Research and Applications (ICRERA)*, IEEE, (2012), 1–4. <https://doi.org/10.1109/ICRERA.2012.6477410>
2. Y. Wang, Z. Liu, J. Xu, W. Yan, Heterogeneous network representation learning approach for ethereum identity identification, *IEEE Trans. Comput. Social Syst.*, **10** (2022), 890–899. <https://doi.org/10.1109/TCSS.2022.3164719>
3. Y. Wang, X. Lin, M. Pedram, A near-optimal model-based control algorithm for households equipped with residential photovoltaic power generation and energy storage systems, *IEEE Trans. Sustainable Energy*, **7** (2015), 77–86. <https://doi.org/10.1109/TSTE.2015.2467190>
4. D. S. Renné, Progress, opportunities and challenges of achieving net-zero emissions and 100% renewables, *Sol. Compass*, **1** (2022), 100007. <https://doi.org/10.1016/j.solcom.2022.100007>

5. S. Lu, B. T. Phung, D. Zhang, A comprehensive review on dc arc faults and their diagnosis methods in photovoltaic systems, *Renewable Sustainable Energy Rev.*, **89** (2018), 88–98. <https://doi.org/10.1016/j.rser.2018.03.010>
6. Q. Xiong, X. Liu, X. Feng, A. L. Gattozzi, Y. Shi, L. Zhu, et al., Arc fault detection and localization in photovoltaic systems using feature distribution maps of parallel capacitor currents, *IEEE J. Photovoltaics*, **8** (2018), 1090–1097. <https://doi.org/10.1109/JPHOTOV.2018.2836986>
7. B. M. Smirnov, *Theory of Gas Discharge Plasma*, Springer, 2015. <https://doi.org/10.1007/978-3-319-11065-3>
8. J. Yuventi, DC electric arc-flash hazard-risk evaluations for photovoltaic systems, *IEEE Trans. Power Delivery*, **29** (2013), 161–167. <https://doi.org/10.1109/TPWRD.2013.2289921>
9. R. F. Ammerman, P. K. Sen, Modeling high-current electrical arcs: A volt-ampere characteristic perspective for AC and DC systems, in *2007 39th North American Power Symposium*, (2007), 58–62. <https://doi.org/10.1109/NAPS.2007.4402286>
10. X. Chen, W. Gao, C. Hong, Y. Tu, A novel series arc fault detection method for photovoltaic system based on multi-input neural network, *Int. J. Electr. Power Energy Syst.*, **140** (2022), 108018. <https://doi.org/10.1016/j.ijepes.2022.108018>
11. X. Liu, A series arc fault location method for dc distribution system using time lag of parallel capacitor current pulses, in *2018 IEEE International Power Modulator and High Voltage Conference (IPMHVC)*, IEEE, (2018), 218–222. <https://doi.org/10.1109/IPMHVC.2018.8936690>
12. O. Mayr, Beiträge zur theorie des statischen und des dynamischen lichtbogens, *Arch. Elektrotech.*, **37** (1943), 588–608. <https://doi.org/10.1007/BF02084317>
13. A. M. Cassie, Arc rupture and circuit severity: A new theory, *CIGRE Rep.*, **102** (1939).
14. S. M. H. Hosseini, E. Y. Eshagh, A. R. Edalatian, The modeling of electric arc in high voltage circuit breakers with use of schavemaker model and negative feedback, *Int. J. Sci. Eng. Invest.*, **2** (2013).
15. G. Liu, S. Du, J. Su, X. Han, Research and development trend of low voltage arc fault protection technology, *Power Grid Technol.*, **1** (2017), 321–329.
16. M. Murakami, H. Ryonai, T. Kubono, J. Sekikawa, Properties of short arc phenomena on agcu electrical contact pairs for automotive electronics devices, in *Electrical Contacts - 2007 Proceedings of the 53rd IEEE Holm Conference on Electrical Contacts*, IEEE, (2007), 146–150. <https://doi.org/10.1109/HOLM.2007.4318208>
17. L. Yue, V. Le, Z. Yang, X. Yao, A novel series arc fault detection method using sparks in dc microgrids with buck converter interface, in *2018 IEEE Energy Conversion Congress and Exposition (ECCE)*, IEEE, (2018), 492–496. <https://doi.org/10.1109/ECCE.2018.8557406>
18. Q. Xiong, S. Ji, L. Zhu, L. Zhong, Y. Liu, A novel dc arc fault detection method based on electromagnetic radiation signal, *IEEE Trans. Plasma Sci.*, **45** (2017), 472–478. <https://doi.org/10.1109/TPS.2017.2653817>
19. J. K. Hastings, J. C. Zuercher, B. Pahl, B. T. Pier, E. T. Gisske, Direct current arc fault circuit interrupter, direct current arc fault detector, noise blanking circuit for a direct current arc fault circuit interrupter, and method of detecting arc faults, 2012.

20. J. C. Gu, D. S. Lai, J. M. Wang, J. J. Huang, M. T. Yang, Design of a dc series arc fault detector for photovoltaic system protection, *IEEE Trans. Ind. Appl.*, **55** (2019), 2464–2471. <https://doi.org/10.1109/TIA.2019.2894992>
21. S. Liu, L. Dong, X. Liao, X. Cao, X. Wang, B. Wang, Application of the variational mode decomposition-based time and time–frequency domain analysis on series dc arc fault detection of photovoltaic arrays, *IEEE Access*, **7** (2019), 126177–126190. <https://doi.org/10.1109/ACCESS.2019.2938979>
22. Z. Wang, R. S. Balog, Arc fault and flash signal analysis in DC distribution systems using wavelet transformation, *IEEE Trans. Smart Grid*, **6** (2015), 1955–1963. <https://doi.org/10.1109/TSG.2015.2407868>
23. S. Chen, X. Li, Y. Meng, Z. Xie, Wavelet-based protection strategy for series arc faults interfered by multicomponent noise signals in grid-connected photovoltaic systems, *Sol. Energy*, **183** (2019), 327–336. <https://doi.org/10.1016/j.solener.2019.03.008>
24. H. Su, W. Qi, C. Yang, J. Sandoval, G. Ferrigno, E. De Momi, Deep neural network approach in robot tool dynamics identification for bilateral teleoperation, *IEEE Rob. Autom. Lett.*, **5** (2020), 2943–2949. <https://doi.org/10.1109/LRA.2020.2974445>
25. H. Su, Y. Hu, H. R. Karimi, A. Knoll, G. Ferrigno, E. De Momi, Improved recurrent neural network-based manipulator control with remote center of motion constraints: Experimental results, *Neural Networks*, **131** (2020), 291–299. <https://doi.org/10.1016/j.neunet.2020.07.033>
26. T. Li, Q. Xiong, R. Li, H. Liu, S. Ji, J. Li, Dc arc fault risk degree evaluation based on back propagation neural network, in *2021 Power System and Green Energy Conference (PSGEC)*, IEEE, (2021), 655–659. <https://doi.org/10.1109/PSGEC51302.2021.9541696>
27. K. Yang, R. Chu, R. Zhang, J. Xiao, R. Tu, A novel methodology for series arc fault detection by temporal domain visualization and convolutional neural network, *Sensors*, **20** (2020), 162. <https://doi.org/10.3390/s20010162>
28. S. Lu, T. Sirojan, B. Phung, D. Zhang, E. Ambikairajah, DA-DCGAN: An effective methodology for DC series arc fault diagnosis in photovoltaic systems, *IEEE Access*, **7** (2019), 45831–45840. <https://doi.org/10.1109/ACCESS.2019.2909267>
29. Y. Wang, C. Bai, X. Qian, W. Liu, C. Zhu, L. Ge, A DC series arc fault detection method based on a lightweight convolutional neural network used in photovoltaic system, *Energies*, **15** (2022), 2877. <https://doi.org/10.3390/en15082877>
30. H. Su, A. Mariani, S. E. Ovrur, A. Menciassi, G. Ferrigno, E. De Momi, Toward teaching by demonstration for robot-assisted minimally invasive surgery, *Trans. Autom. Sci. Eng.*, **18** (2020), 484–494. <https://doi.org/10.1109/TASE.2020.3045655>
31. H. Su, W. Qi, Y. Hu, H. R. Karimi, G. Ferrigno, E. De Momi, An incremental learning framework for human-like redundancy optimization of anthropomorphic manipulators, *IEEE Trans. Ind. Inf.*, **18** (2020), 1864–1872. <https://doi.org/10.1109/TII.2020.3036693>
32. J. Zhao, Y. Lv, Output-feedback robust tracking control of uncertain systems via adaptive learning, *Int. J. Control Autom. Syst.*, **21** (2023), 1108–1118. <https://doi.org/10.1007/s12555-021-0882-6>

33. W. Qi, H. Fan, H. R. Karimi, H. Su, An adaptive reinforcement learning-based multimodal data fusion framework for human–robot confrontation gaming, *Neural Networks*, **164** (2023), 489–496. <https://doi.org/10.1016/j.neunet.2023.04.043>
34. W. Qi, H. Su, A cybertwin based multimodal network for ecg patterns monitoring using deep learning, *IEEE Trans. Ind. Inf.*, **18** (2022), 6663–6670. <https://doi.org/10.1109/TII.2022.3159583>
35. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., Attention is all you need, in *Advances in Neural Information Processing Systems*, **30** (2017). Available from: [https://proceedings.neurips.cc/paper\\_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf).
36. J. Devlin, M. W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, preprint, arXiv:1810.04805.
37. C. Z. A. Huang, A. Vaswani, J. Uszkoreit, N. Shazeer, I. Simon, C. Hawthorne, et al., Music transformer, preprint, arXiv:1809.04281.
38. Y. Liu, J. Zhang, L. Fang, Q. Jiang, B. Zhou, Multimodal motion prediction with stacked transformers, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2021), 7577–7586.
39. S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y. X. Wang, et al., Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting, in *Advances in Neural Information Processing Systems*, **32** (2019). Available from: [https://proceedings.neurips.cc/paper\\_files/paper/2019/file/6775a0635c302542da2c32aa19d86be0-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2019/file/6775a0635c302542da2c32aa19d86be0-Paper.pdf).
40. H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, et al., Informer: Beyond efficient transformer for long sequence time-series forecasting, in *Proceedings of the AAAI Conference on Artificial Intelligence*, **35** (2021), 11106–11115. <https://doi.org/10.1609/aaai.v35i12.17325>
41. N. Kitaev, Ł. Kaiser, A. Levskaya, Reformer: The efficient transformer, preprint, arXiv:2001.04451.
42. H. Wu, J. Xu, J. Wang, M. Long, Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting, in *Advances in Neural Information Processing Systems*, **34** (2021), 22419–22430. Available from: [https://proceedings.neurips.cc/paper\\_files/paper/2021/file/bcc0d400288793e8bdcd7c19a8ac0c2b-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2021/file/bcc0d400288793e8bdcd7c19a8ac0c2b-Paper.pdf)
43. S. Liu, H. Yu, C. Liao, J. Li, W. Lin, A. X. Liu, et al., Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting, in *International Conference on Learning Representations*, 2022. Available from: [https://dsg.tuwien.ac.at/team/sd/papers/ICLR\\_2022\\_SD\\_Pyraformer.pdf](https://dsg.tuwien.ac.at/team/sd/papers/ICLR_2022_SD_Pyraformer.pdf).
44. T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, R. Jin, Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting, in *Proceedings of the 39th International Conference on Machine Learning*, PMLR, **162** (2022), 27268–27286. Available from: <https://proceedings.mlr.press/v162/zhou22g.html>.

- 
45. Q. Wen, J. Gao, X. Song, L. Sun, H. Xu, S. Zhu, Robuststl: A robust seasonal-trend decomposition algorithm for long time series, in *Proceedings of the AAAI Conference on Artificial Intelligence*, **33** (2019), 5409–5416. <https://doi.org/10.1609/aaai.v33i01.33015409>
  46. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016), 770–778.



AIMS Press

©2024 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)