

## Convolutional Neural Networks for Fault Detection in Grid-Connected Photovoltaic Panels

K.V.B. Saraswathi Devi\*<sup>ID</sup>, Muktevi Srivenkatesh<sup>ID</sup>

Department of Computer Science, GITAM Deemed to be University, Visakhapatnam 530045, Andhra Pradesh, India

Corresponding Author Email: [121962504003@gitam.in](mailto:121962504003@gitam.in)

Copyright: ©2023 IIETA. This article is published by IIETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/isi.280619>

### ABSTRACT

**Received:** 21 June 2023

**Revised:** 23 August 2023

**Accepted:** 13 October 2023

**Available online:** 23 December 2023

#### **Keywords:**

*fault detection, photovoltaic panel, deep neural networks, binary classification, multiclass classification, resilience*

The ongoing surveillance of solar panel output power is a robust technique for identifying solar panel malfunctions. In this study, any divergences from the anticipated power output are meticulously analyzed to discern the origin of the fault. A method based on a trained convolutional neural network (CNN) is suggested for defect detection, designed to segment images of photovoltaic modules, thereby enhancing the resilience of the solar power system. The proposed deep sequential model bifurcates the input images of photovoltaic cells into two categories, i.e., defective and normal, facilitating binary classification. The defective photovoltaic cells are further classified into subgroups, such as dim, fractured, or dusty cells, allowing for multiclass classification. This further evaluation elucidates the network's capacity. Remarkably, the proposed model demonstrated a fault diagnosis accuracy of 99.947% in solar panels, surpassing other models.

## 1. INTRODUCTION

Fault detection in solar panels, typically conducted through the analysis of output power data, is an established technique for diagnosing malfunctions within these renewable energy systems. This method necessitates the continuous monitoring of the solar panel's output power, wherein any deviations from the anticipated power output are scrutinised for determining the origin of the fault [1]. The process initiates with the establishment of a baseline for the expected output power of the solar panel, a parameter derived through a combination of manufacturer specifications, simulations, and measurements. Subsequent to the determination of the expected output power, the actual output power is continuously monitored. Deviations from the expected output power may signify a fault within the solar panel system.

For instance, consistently lower output power than expected may point towards issues such as shading or a malfunctioning cell within the panel. Conversely, consistently higher output power than expected might imply the existence of a faulty voltage regulator or a wiring issue. The identification of the specific cause of the fault could necessitate additional diagnostics, such as thermal imaging to locate hot spots on the solar panel, which could indicate a malfunctioning cell, or electrical tests to detect issues with the wiring or voltage regulators. By and large, fault detection in solar panels using output power data is a valuable technique for diagnosing issues, facilitating the swift detection and diagnosis of faults, thereby minimising downtime and enhancing the efficiency of the solar panel system [2].

The rapid proliferation of grid-integrated renewable energy sources has led to the evolution of hybrid grids. However, the inconsistent characteristics of renewable energy power generation have resulted in challenges associated with power

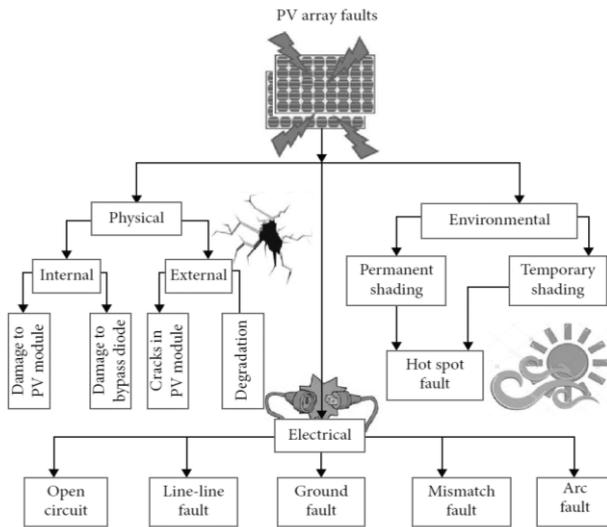
quality (PQ), system reliability, and stability, adversely impacting the operation of protective relays in real-time. Furthermore, high penetration of renewable energy sources into the utility grid could be achieved by employing multi-tapped transmission lines, particularly for the integration of wind and solar powers [3]. Such a development has posed substantial challenges for power system engineers in the design of suitable protection schemes. Different faults and incidents occurring on the power system network can engender problems relating to power quality, power system protection, and power network stability. The significance and severity of power system faults vary corresponding to the frequency of occurrence and magnitude of power flow in the network, as illustrated in Figure 1.

The penetration of renewable energy (RE) into utility networks has been increasing steadily, bringing into focus issues concerning energy security, equipment protection, grid security, power reliability, and power quality. The integration of RE near load centres has further exacerbated problems of false and delayed tripping within the protection system, primarily due to the uncertain and variable nature of RE generation. RE sources are known to cause considerable voltage transients and high current variations during outages and synchronization with the grid.

Short-circuiting of the modules composing a panel can lead to internal disruptions, resulting in decreased current and voltage readings, thereby reducing the total power output. This reduction can also be triggered by naturally existing external elements that diminish the amount of sunlight (irradiance) impinging on the surface of the solar panels, leading to dips in current and voltage readings and subsequently affecting the overall power output. A critical issue arises when changes in readings induced by internal defects are frequently conflated with those caused by external disturbances. It is crucial to

address only the internal issues; the external disturbances, which could be as benign as a passing cloud, can often be overlooked.

The primary objective of this work is to isolate the internal (Line-to-Line or LL) faults that may be present within a panel and dismiss the decrease in power output triggered by external (shading) disturbances. This focus on internal faults allows for a more precise understanding and troubleshooting of issues that directly impact the efficiency and functionality of the solar panels.



**Figure 1.** Fault types in Solar panels

### 1.1 Implications of faults in solar panels

Faults within solar panels typically result in a reduction of maximum power generation. Instead of identifying the Global Maximum Power Point (GMPP), only the Local Maximum Power Point (LMPP) is detected, thereby diminishing the efficiency of the entire solar PV system [4]. Herein,  $I_{pv}$  and  $V_{pv}$  represent the PV array current measurement and voltage measurement, respectively, while  $I_a$ ,  $I_b$ , and  $I_c$  denote current measurements of Phase\_A, Phase\_B, and Phase\_C, respectively. Similarly,  $V_a$ ,  $V_b$ , and  $V_c$  correspond to voltage measurements of phases A, B, and C, respectively. These complexities highlight the necessity for swift fault detection and classification by PV installers or consumers to facilitate effective rectification.

In light of such challenges, a range of contemporary methods, including neural networks, fuzzy logic, and machine learning algorithms, has been successfully employed in the literature and has gained increasing popularity in the research field. Furthermore, various strategies, such as Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Genetic algorithms, and advanced neuro-fuzzy methods, have been proposed and examined for monitoring the maximum power point under partially shaded conditions. These advanced methodologies aim to enhance the reliability and efficiency of solar PV systems by providing more accurate and timely fault detection and classification.

## 2. RELATED WORK

Bhadla Solar Park, one of the world's largest solar power parks, is situated in the sandy region of Bhadla, Rajasthan.

Comprising 10 million solar panels across 14,000 acres, it boasts a total capacity of 2245MW. The growth of the PV industry has been found to increase exponentially every three years according to recent surveys, with solar power being utilized in various applications, such as electric vehicles and smart grids [5].

Solar energy is primarily harnessed through photovoltaic (PV) panels, generating DC power. The PV panels available in the market are diverse, including amorphous silicon (a-Si), mono-crystalline Silicon, Polycrystalline Silicon, Cadmium Telluride (CdTe), and Copper Indium Gallium Selenide (CIGS), allowing consumers and installers to select based on their respective requirements. However, despite the benefits of PV utilization, it faces challenges such as partial shading, low conversion efficiency, aging faults, and high installation costs [6].

Partial shading, a phenomenon where some parts of the PV panels do not receive full irradiation or when irradiation is unequal across a PV array, can significantly reduce power generation [7]. This can be due to a variety of reasons, including tree shadows, proximity to tall buildings, bird nests, sand deposition, and passing clouds. This results in multiple peaks in power output, which has led researchers to propose numerous maximum power point tracking algorithms. Techniques like Particle Swarm Optimization-based MPPT algorithms and Grey Wolf Optimization (GWO) algorithm-based MPPT have been implemented for their simplicity and reduced hardware requirements.

The performance of PV array under various shading patterns has been evaluated, considering patterns such as uneven row, uneven column, diagonal, random short and narrow, short and wide, long and narrow, and long and wide, among others [8]. The PV array performance is typically assessed under three conditions: normal operating, partial shading, and faulty. Aging and mismatch, two interrelated factors, can lead to panel failures. A single panel failure can impact the entire array's performance. Aging occurs due to prolonged exposure to wind and weather.

Several solutions have been proposed to address these issues. Kellil et al. [9] suggested a reconfiguration arrangement based on properties such as overheating and damage of the aged module. Mismatch conditions, often resulting from non-uniform aging, occur when the cells are uneven [10]. The efficacy of detection can be improved through hybrid models. In view of this, the authors of the present study have designed a new hybrid model aimed at detecting the aforementioned faults.

## 3. METHODOLOGY

### 3.1 Deep sequential model

A deep sequential model, also known as a deep sequential neural network, is a type of artificial neural network that is designed to process data in a sequential or temporal manner. These models are commonly used for tasks involving sequences of data, such as time series analysis, natural language processing, speech recognition, and more. Deep sequential models are capable of learning complex patterns and dependencies within sequential data, which makes them suitable for a wide range of applications.

Key components and characteristics of deep sequential models include:

**Sequential Processing:** Deep sequential models process input data one step at a time, maintaining a sense of order or sequence. This is in contrast to feedforward neural networks, which take static inputs and produce static outputs.

**Deep Architecture:** Deep sequential models have multiple layers of processing units (neurons or cells) stacked on top of each other. These layers can be recurrent, convolutional, or a combination of both, depending on the specific architecture.

**Recurrent Layers:** Recurrent neural networks (RNNs) are a common choice for modeling sequential data. RNNs have connections that loop back on themselves, allowing them to maintain an internal state and capture dependencies over time.

**Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU):** These are specialized recurrent layers that are designed to address the vanishing gradient problem in traditional RNNs. LSTMs and GRUs can capture long-term dependencies in data.

**Convolutional Layers:** In some cases, deep sequential models may also include convolutional layers, which are well-suited for tasks involving sequences with spatial components, such as image sequences or sensor data.

**Variable-Length Inputs:** Deep sequential models can handle input sequences of variable length. This flexibility is important for many real-world applications where the length of the sequence may vary.

**Training:** Training deep sequential models typically involves backpropagation through time (BPTT) for RNNs, which is a variant of the standard backpropagation algorithm. Training deep sequential models can be computationally intensive and may require careful initialization and regularization techniques.

Deep sequential models have been successfully applied to a wide range of tasks, including natural language understanding and generation, speech recognition, machine translation, sentiment analysis, video analysis, and more.

Common deep sequential model architectures include:

**Recurrent Neural Networks (RNNs):** Basic RNNs are the simplest form of deep sequential models. However, they have limitations in capturing long-range dependencies.

**Gated Recurrent Unit (GRU) Networks:** GRUs are another variant of RNNs, which are computationally more efficient than LSTMs while still being able to capture long-term dependencies as shown in Figure 2.

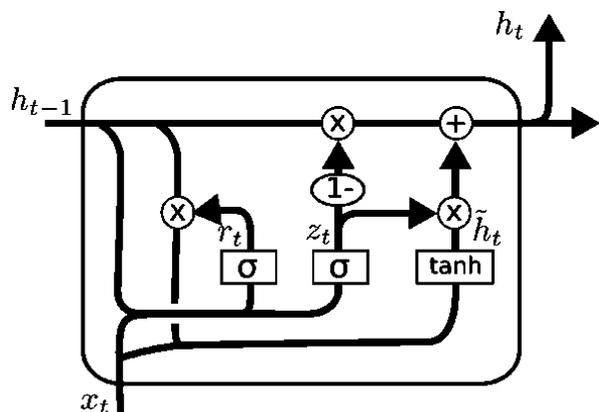


Figure 2. Gated Recurrent Unit (GRU)

**Transformer-Based Models:** Transformers have gained popularity for natural language processing tasks and sequence-to-sequence tasks. They use self-attention mechanisms to capture dependencies in input sequences.

Deep sequential models have significantly advanced the state of the art in many areas of artificial intelligence and machine learning. They are a fundamental tool for modeling and making predictions on sequential data. Deep sequential models are implemented when the input and output are both data sequences as shown in Figure 3. The points of data can be arranged into sequences so that significant data about observation performed at subsequently points in the collection may be deduced from measurements performed at a single instance in the sequence. The sequence learning issue can occur when a parameter is a sequence and an output is a single data point, as in the instances of video action identification, sentiment classification, and stock price predictions.

For sequence data, managing ongoing supervised learning processes is necessary [11]. The combined length of the sequences used for input and output may vary and may appear the same or different in instances where both the input and the result are sequences, such as recognition of speech, natural language understanding, and sequence DNA analysis. Neural networks based on deep learning have mostly been applicable to speech recognition, natural language processing, and picture analysis since they demonstrate the capacity to capture extremely complex input and output mapping [12]. This approach ultimately led to the development of multiple deep-learning time series forecasting structures that exceed conventional methods in terms of precision and efficacy.

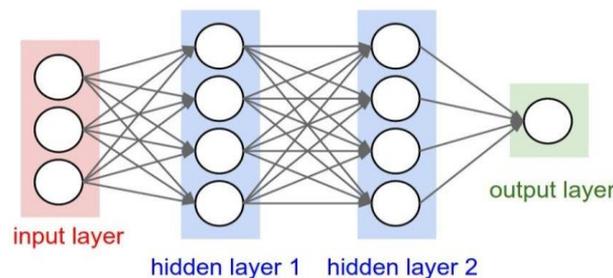


Figure 3. Deep sequential model

### 3.1.1 Long short-term memory networks

Recurrent neural networks serve in the analysis of input neural network categories in sequence. Because of these hidden states, previously predicted amounts can be used as inputs as shown in Figure 4.

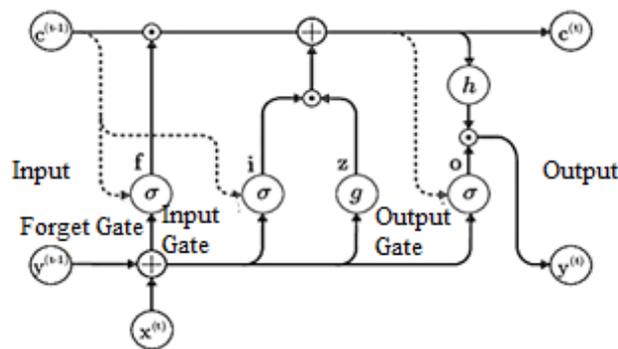


Figure 4. Recurrent neural networks model

An RNN comprising input, output, and a hidden layer make up a multilayer perceptron [13]. The n layers that make up a multi-layered perceptron are established by the particular order in which they are generated. The Keras software

platform was employed to create the deep learning algorithm. As data shows up, is stored by the algorithm, and becomes available points, a structure must be maintained. An input gate, a memory gate, and an output gate were the three gates that LSTMs utilize. Each of the three of these gates, that compose the vast majority of the LSTM model shown in Figure 5, are in the position of regulating the monitoring of everything. Eq. 1 indicates that the recollecting component of the gate utilizes a combination of the previously concealed state and the data item being processed by the sequencing gate to identify which components of the LSTM should now be removed [14].

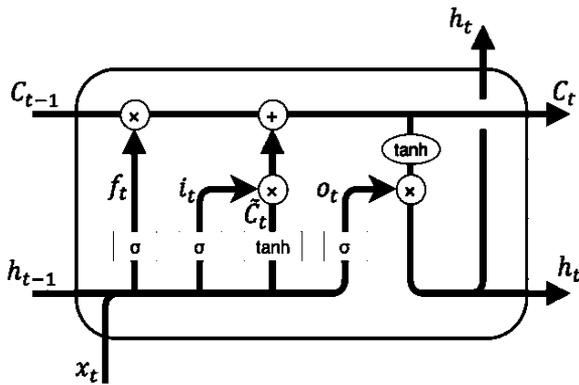


Figure 5. Long short-term memory model

Equation is the input component of the computational Eq. 2 with the gate.

$$F(t) = \sigma(W_f [h_{t-1}, X_t] + b_f) \quad (1)$$

$$I(t) = \sigma(W_i [h_{t-1}, X_t] + b_i) \quad (2)$$

$$f(x) = \frac{1}{1 + e^{ax}} \quad (3)$$

$$\tanh(x) = \frac{2}{1 + e^{-2x}} - 1 \quad (4)$$

The newly developed concealed state includes to be generated through the output gates. The most recently updated input data, the least current concealed state, and the most current real state of cells were all taken into consideration when choosing the option that was made. The remember gate, input gate, and output gate weight values were  $W_f$ ,  $W_i$ , and  $W_o$ , correspondingly. The response to Eq. 4 indicates that the formula operator sigmoid used data from both the most current input and the previously provided hidden layer input [15]. The  $(\tanh)$  function has been utilized to communicate data regarding the parameters connected to the input to the gate, the current source, and its previously concealed state. Whereas the range of a sigmoid  $(\sigma)$  function's value is 0 to 1, those of the  $\tanh$  function is -1 to 1.

### 3.1.2 Multilayer perceptron

An artificial neural network with multiple layers of connected neurons or nodes is referred to as a feedforward neural network, or multilayer perceptron (MLP) as shown in Figure 6. Information passes from the input layer to the final layer of output through the hidden layers.

Machine learning tasks including pattern recognition, regression, and classification belong to several applications for MLPs.

Key characteristics of a Multilayer Perceptron as shown in Figure 6:

**Input Layer:** The nodes in this layer represent in all the variables that have been entered. The number of nodes in this layer is decided by the dimensionality of the input data, each node indicating a feature.

**Hidden Layers:** There could be one or more hidden layers in between the input and output layers. Nodes in these categories conduct out computations and transform the input data into a format appropriate for the output. Hyper parameters that can be modified are the amount of nodes in each layer and the number of hidden stages.

**Output Layer:** The algorithm's final predictions or outputs are produced by the output layer. The type of analysis being performed (e.g., regression, multi-class classification, binary classification) affects how many nodes belong to this layer.

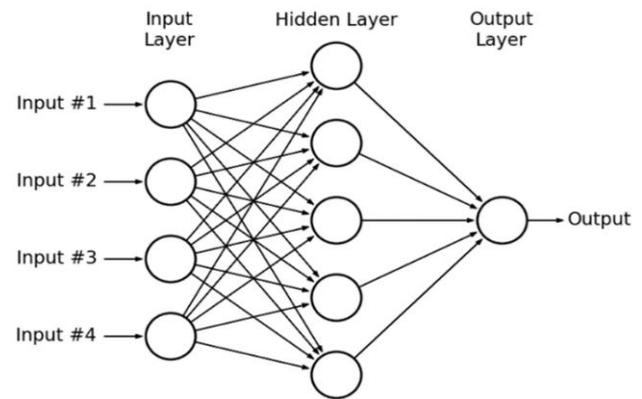


Figure 6. Multi-layer perceptron

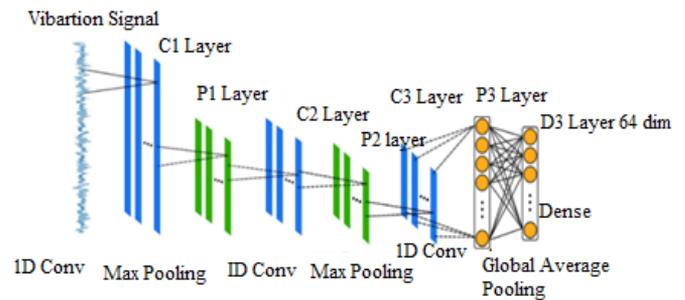


Figure 7. The progression of the process

Another type of artificial neural network that is frequently used for image and video recognition applications is a convolutional neural network (CNN). They were designed to process input data having a structure resembling a grid, like an image.

An MLP can process the contextual data, such as maintenance logs or historical operating conditions, while an LSTM can capture temporal dependencies in sensor data. The combined model can predict equipment failures more accurately.

Hybrid models that combine Multilayer Perceptrons (MLPs) and Long Short-Term Memory (LSTM) networks can be powerful tools in machine learning, especially when dealing with complex data that includes both structured and sequential

components. The progression of the process is as shown in Figure 7.

#### 4. DATA PREPROCESSING

Data on Grid-connected PV System malfunctions (GPVS faults) are gathered through laboratory tests of PV microgrid system malfunctions. There are 16 data files '.csv' files-each representing a single experiment scenario and containing information on defects in photovoltaic arrays, inverters, grid anomalies, feedback sensors, and MPPT controllers of varying severity [16]. For the purpose of reactive maintenance and PV system protection, GPVS-Faults data can be used to create, validate, and evaluate various fault detection, diagnosis, and classification methods. The faults were manually introduced in the middle of the experiments. Low-magnitude fault identification is negatively impacted by MPPT/IPPT modes; high-frequency data are noisy; temperature and insolation are disrupted and fluctuate both during and between testing. The issue is to find the flaws before a complete failure since with significant faults, the operation is interrupted and the system may shut down. Time: Time of real measurement in seconds. The average sampling is = 9.9989ms.

If: Positive-sequence estimated current frequency.

Vdc: DC voltage measurement.

Vabc: Positive-sequence estimated voltage magnitude.

Iabc: Positive-sequence estimated current magnitude.

Vf: Positive-sequence estimated current frequency.

Data preprocessing is a crucial step in preparing a dataset for analysis or machine learning. Without proper preprocessing, the data might contain inconsistencies, noise, or missing values that can negatively affect the quality of the results.

Missing values in the dataset were removed using techniques such as mean, median, or machine learning algorithms. Data is split into train data, test data, and validation data [17].

In this work standardScaler and labelEncoder methods are used for data preprocessing. StandardScaler is a popular method for standardizing (or z-score scaling) numerical features in your dataset. Standardization transforms the data such that it has a mean of 0 and a standard deviation of 1. This can be important for some machine learning algorithms that are sensitive to the scale of features.

LabelEncoder encodes the values in the target column of the solar\_data\_lim\_power DataFrame. The categorical labels will be transformed into numerical values.

#### 5. EXPERIMENTATION RESULTS

##### 5.1 Performance metrics

It is essential to choose the appropriate performance criteria for the purpose of assessing machine learning and deep learning algorithms successfully. They mainly used the performance statistics of recall (R), accuracy (A), precision (P), and F1-score (F1) to satisfy the objective of this research.

$$precision = \frac{\text{True positive } (Tp)}{\text{True positive } (Tp) + \text{False positive } (Fp)} \quad (5)$$

$$Recall = \frac{\text{True positive } (Tp)}{\text{True positive } (Tp) + \text{False Negative } (Fn)} \quad (6)$$

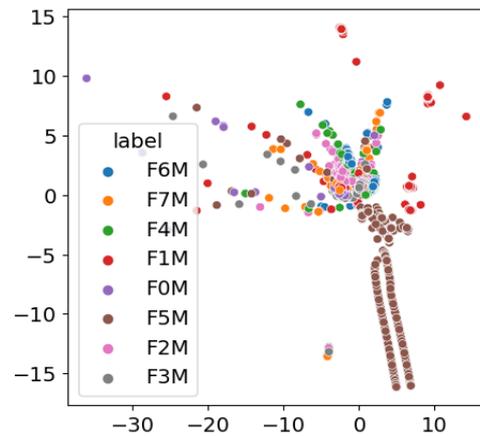
$$Accuracy = \frac{Tp + TN}{Tp + TN + FP + FN} \quad (7)$$

$$F1score = 2 * \frac{Recall * precision}{Recall + precision} \quad (8)$$

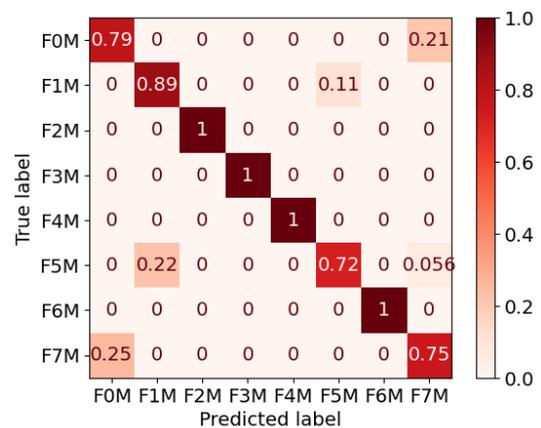
Table 1 presents the accuracy, precision, recall, and F1 score results for the Random Forest classifier, K-nearest neighbor, SVM classifier, ensemble learning, and proposed model results. The technique to evaluate a machine learning model's performance based on multiple subsets of the training data is known as cross-validation. The standard deviation of the performance metrics across these subsets can give you an idea of how stable the model's performance is. A lower standard deviation generally indicates more consistent and stable performance.

**Table 1.** Assessment of model output parameters

Model	Accuracy	Standard Deviation
Random Forest Classifier	0.9854	0.00215
K-nearest Neighbor	0.94967	0.003434
SVM Classifier	0.941301	0.00465
Proposed Model	0.99947	0.99942



**Figure 8.** Plotting the reduced dimensionality data with the corresponding fault label



**Figure 9.** Confusion matrix of prediction

The standard deviation of a machine learning model's performance metric is a measure of the model's consistency. A low standard deviation indicates stable and predictable performance, while a high standard deviation suggests less stable and inconsistent performance. When comparing different models or evaluating the same model under various conditions, it's often preferable to choose the model with lower standard deviation, as it is more likely to generalize well to new, unseen data. However, it's crucial to consider other factors and conduct a comprehensive evaluation of model performance, as standard deviation alone may not capture all aspects of a model's behavior.

Our proposed model got very good standard deviation value 0.99942, which shows the stability of the model.

The faults were manually introduced in the middle of the experiments. The readings at high frequencies are erratic. Between the scenarios and throughout them, there are disruptions. Temperature and insolation change throughout the trials and between them. The issue is to find and diagnose the defects before they result in such a complete failure because in critical fault scenarios, the operation is halted and the system may shut down. Furthermore, MPPT/ IPPT modes impair the ability to identify low-magnitude faults. The results of the prediction are shown in Figures 7 and 8.

Figure 9 shows the Confusion matrix of the model, is a valuable for understanding a model's strengths and weaknesses, particularly in situations where one type of error (false positives or false negatives) is more costly or critical than the other. Analyzing the confusion matrix helps you fine-tune the proposed machine-learning models and make informed decisions about model improvements or adjustments.

## 6. CONCLUSION

Photovoltaic systems are susceptible to a variety of errors and failures, prompt fault identification is crucial for their safety and effectiveness. Our proposed model deep sequential model-based fault detection is trained to utilize data, as well as the resulting prediction results are extremely accurate. The proposed model gets a good accuracy of 99.95%, which is better than other machine learning models. we can implement transfer learning to improve accuracy in the future.

## REFERENCES

- [1] Ji, D., Zhang, C., Lv, M., Ma, Y., Guan, N. (2017). Photovoltaic array fault detection by automatic reconfiguration. *Energies*, 10(5): 699. <https://doi.org/10.3390/en10050699>
- [2] Haque, A., Bharath, K.V.S., Khan, M.A., Khan, I., Jaffery, Z.A. (2019). Fault diagnosis of photovoltaic modules. *Energy Science & Engineering*, 7(3): 622-644. <https://doi.org/10.1002/ese3.255>
- [3] Ali, M.H., Rabhi, A., El Hajjaji, A., Tina, G.M. (2017). Real time fault detection in photovoltaic systems. *Energy Procedia*, 111: 914-923. <https://doi.org/10.1016/j.egypro.2017.03.254>
- [4] Wang, K., Qi, X., Liu, H. (2019). A comparison of day-ahead photovoltaic power forecasting models based on deep learning neural network. *Applied Energy*, 251: 113315. <https://doi.org/10.1016/j.apenergy.2019.113315>
- [5] Takashima, T., Yamaguchi, J., Otani, K., Kato, K., Ishida, M. (2006). Experimental studies of failure detection methods in PV module strings. In 2006 IEEE 4th World Conference on Photovoltaic Energy Conference, Waikoloa, HI, USA, pp. 2227-2230. <https://doi.org/10.1109/WCPEC.2006.279952>
- [6] Yi, Z., Etemadi, A.H. (2017). Line-to-line fault detection for photovoltaic arrays based on multiresolution signal decomposition and two-stage support vector machine. *IEEE Transactions on Industrial Electronics*, 64(11): 8546-8556. <https://doi.org/10.1109/TIE.2017.2703681>
- [7] Chen, Y., Altermatt, P.P., Chen, D., et al. (2018). From laboratory to production: learning models of efficiency and manufacturing cost of industrial crystalline silicon and thin-film photovoltaic technologies. *IEEE Journal of photovoltaics*, 8(6): 1531-1538. <https://doi.org/10.1109/JPHOTOV.2018.2871858>
- [8] Kumar, B.S., Sudhakar, K. (2015). Performance evaluation of 10 MW grid connected solar photovoltaic power plant in India. *Energy reports*, 1: 184-192. <https://doi.org/10.1016/j.egy.2015.10.0014>
- [9] Wang, F., Zhang, Z., Chai, H., Yu, Y., Lu, X., Wang, T., Lin, Y. (2019). Deep learning based irradiance mapping model for solar PV power forecasting using sky image. In 2019 IEEE industry applications society annual meeting Baltimore, MD, USA, pp. 1-9. <http://dx.doi.org/10.1109/IAS.2019.8912348>
- [10] Prince Winston, D., Ganesan, K., Samithas, D., Baladhanautham, C.B. (2020). Experimental investigation on output power enhancement of partial shaded solar photovoltaic system. *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, 1-17. <https://doi.org/10.1080/15567036.2020.1779872>
- [11] Mathi, D.K., Chinthamalla, R. (2020). A hybrid global maximum power point tracking method based on butterfly particle swarm optimization and perturb and observe algorithms for a photovoltaic system under partially shaded conditions. *International Transactions on Electrical Energy Systems*, 30(10): e12543. <https://doi.org/10.1002/2050-7038.12543>
- [12] Ibrahim, M.S., Dong, W., Yang, Q. (2020). Machine learning driven smart electric power systems: Current trends and new perspectives. *Applied Energy*, 272: 115237. <https://doi.org/10.1016/j.apenergy.2020.115237>
- [13] Kellil, N., Aissat, A., Mellit, A. (2023). Fault diagnosis of photovoltaic modules using deep neural networks and infrared images under Algerian climatic conditions. *Energy*, 263: 125902. <https://doi.org/10.1016/j.energy.2022.125902>
- [14] Khan, K., Rashid, S., Mansoor, M., Khan, A., Raza, H., Zafar, M.H., Akhtar, N. (2023). Data-driven green energy extraction: Machine learning-based MPPT control with efficient fault detection method for the hybrid PV-TEG system. *Energy Reports*, 9: 3604-3623. <https://doi.org/10.1016/j.egy.2023.02.047>
- [15] Wang, F., Xuan, Z., Zhen, Z., Li, K., Wang, T., Shi, M. (2020). A day-ahead PV power forecasting method based on LSTM-RNN model and time correlation modification under partial daily pattern prediction framework. *Energy Conversion and Management*, 212: 112766. <https://doi.org/10.1016/j.enconman.2020.112766>
- [16] Boukoffa, K., Metatla, A., Msabah, I.L., Benzahiou, S. (2023). Faults diagnosis in PV systems using structured residuals and indicator parameters techniques. *Journal Européen des Systèmes Automatisés*, 56(2): 317-327.

<https://doi.org/10.18280/jesa.560217>

- [17] Zeghoudi, A., Benmouiza, K. (2023). Solar power heliostat control using image processing technology and artificial neural networks. *Journal Européen des Systèmes Automatisés*, 56(1): 165-171. <https://doi.org/10.18280/jesa.560120>

## NOMENCLATURE

$h_t$	hidden layer vectors.
$x_t$	input vector.
$b_z, b_r, b_h$	bias vector.
$W_z, W_r, W_h$	parameter matrices
$\sigma, \tanh$	activation functions.