RESEARCH

Photovoltaic Panels Fault Detection with Convolutional Neural Network and Bitterling Fish Optimization (BFO) Algorithm

Asghar Sabati[1] · Ramazan Bayindir[2] · Javad Rahebi[3]

Received: 1 May 2025 / Revised: 13 August 2025 / Accepted: 7 September 2025
© The Author(s) 2025

Abstract
This paper presents a robust framework for detecting faults in PV panels using Convolutional Neural Networks (CNNs) for feature extraction and Bitterling Fish Optimization (BFO) algorithm for feature selection. The system integrates five pre-trained CNN architectures—GoogleNet, SqueezeNet, ResNet-50, VGGNet-16, and AlexNet—with BFO to optimize feature selection and improve classification performance. The VGGNet-16 + BFO combination achieved the highest accuracy of 98.75%, with excellent sensitivity of 98.72%, specificity of 98.78%, precision of 98.76%, and F1 score of 98.74%, demonstrating highly balanced and reliable performance. The present research shows that the BFO algorithm plays a crucial role in selecting the most significant features and improving classification accuracy compared to conventional techniques. The measures like accuracy, sensitivity, specificity, precision, and F1 score validate the success of the model in correctly predicting faulty PV panels. From the overall performance perspective, BFO performs better than alternative optimization methods such as Gray Wolf Optimization (GWO) and Ant Colony Optimization (ACO). The success of this approach with various CNN models highlights the effectiveness of BFO for accurate PV panel defect identification. The combination of BFO and CNN provides high accuracy and balanced performance in PV panel defect detection, making it a viable solution for improving the efficiency and reliability of solar systems.

Keywords Conventional Neural Network · Fault detection · Bitterling Fish Optimization algorithm

1 Introduction

The growing dependence of more and more countries on renewable energy has caused a rapid increase in the use of solar panels [1]. While, keeping these panels in good working condition is crucial to achieve the maximum energy that they generate [2]. Gradually, problems such as small cracks, discoloration, or shading can happen and affect the performance of the panels [3], so early detection of such problems avoids significant repairs and facilitates ongoing energy generation [4].

Currently, panel inspections are usually done by hand. It takes a lot of effort and time, and occasionally mistakes are made [5]. For ease of use, computer systems with image and data learning features can be used to help detect problems more quickly [6]. A common approach is to use neural networks, which are able to recognize shapes and patterns in images [7]. However, these networks often extract a lot of data, including unnecessary ones. To



Published online: 01 October 2025

 https://doi.org/10.1007/s44196-025-00984-4

solve this problem, smarter selection tools, such as the Bitterling Fish Optimization (BFO) method, can select the important ones and ignore the rest.

Traditional fault detection methods, such as visual inspection or the use of sensors, are no longer efficient—especially when the number of panels is large. Therefore, it makes sense to use intelligent tools that can handle large configurations, detect problems early, and help maintain a smooth flow of energy.

Image-based fault detection offers a visual alternative to traditional methods. Using visual data captured from PV panels, it can identify faults much faster and more accurately than manual fault inspection. Recent breakthroughs in ML, especially through the Convolutional Neural Networks (CNNs), have made them highly effective at extracting complex features from images. CNNs automatically learned and recognized intricate patterns like micro-cracks or discoloration without extensive human engineering of features.

Among the many issues PV systems face is their outdoor location, which can drastically affect their energy output, potential, and, most importantly, their ability to handle varying load needs. Figure 1 summarizes the three fundamental categories faults: electrical, environmental, and physical.

Short circuits, circuit breakdowns, bypass diode problems, and shunt resistance insertion problems are examples of electrical defects. These defects lower output power by reducing voltage and current. Shading

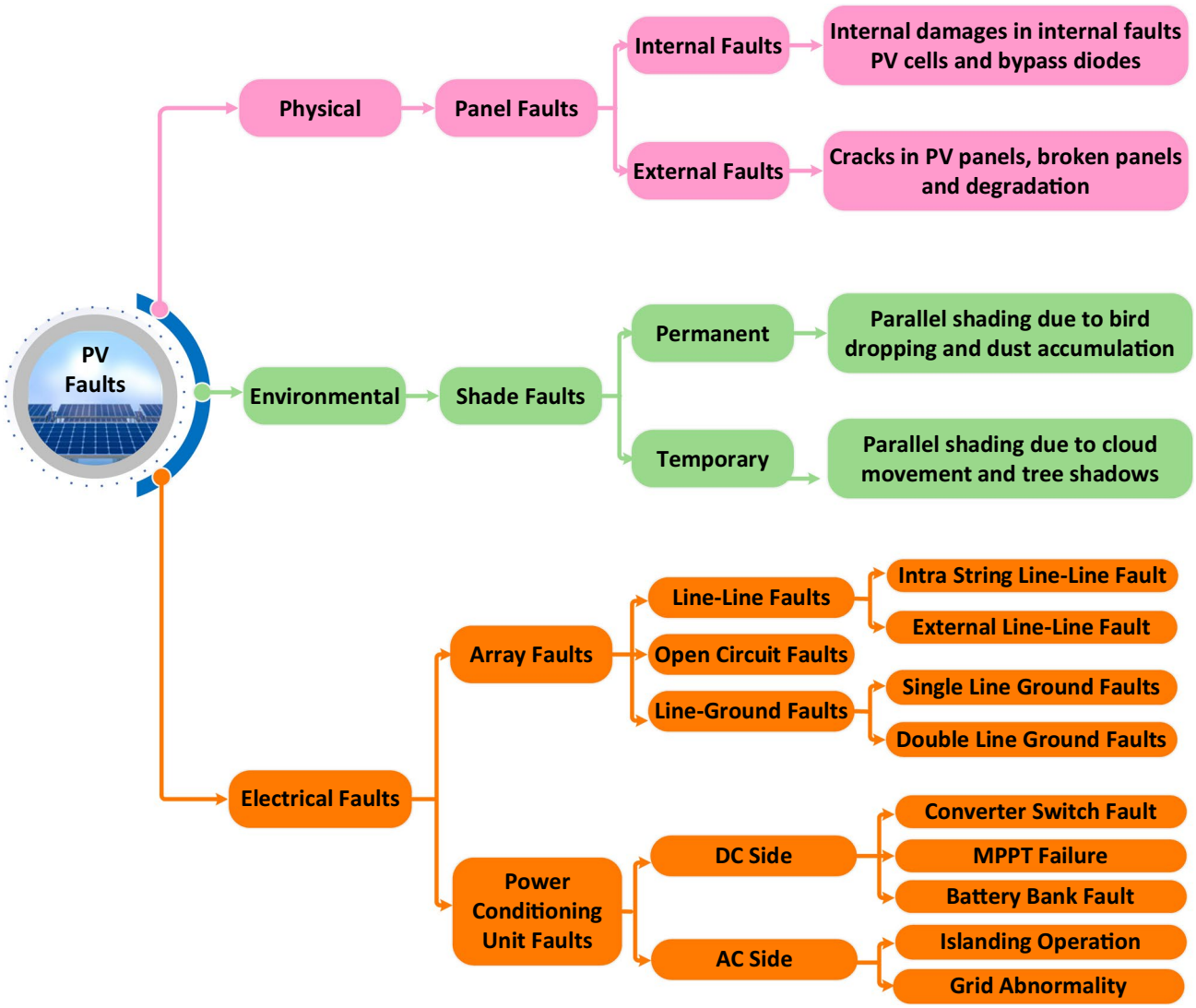


Fig. 1 PV system fault classification

is a common environmental failure that typically results in the bypass effect, where currents are diverted and rated lower than they should be. Internal defects and micro-cracks are examples of physical flaws.

Despite the strengths of CNNs, a large number of features extracted from PV fault images often present challenges. A feature set with many dimensions is usually characterized by redundancy or irrelevant information, which can reduce processing speeds, increase computational costs, and cause overfitting in fault classification. These challenges require effective dimensionality reduction techniques.

This issue is somewhat driven by ongoing research in feature selection optimization algorithms. On the other hand, the need to identify the best features from large datasets and reduce computational complexity has been considered in this research. The BOF algorithm, inspired by nature-inspired bitterling fish behavior, offers a distinct but efficient technique. By mimicking their reproduction approaches, BFO provides an efficient search for the optimal subset of features, which keeps the most relevant ones for classification. This approach reduces the burden of calculation and improves the accuracy and reliability of the fault detection system. Therefore, the primary issue is how to create an efficient and accurate system for fault detection in photovoltaic panels from images. To ensure the correctness of fault classification is necessary to extract the most wanted features from fault images while minimizing the feature sizes without losing much information.

The proposed approach includes two main parts:

1. Feature Extraction: Fault images of PV panels are input into a CNN that extracts the features. CNNs can easily do this since they are able to learn complex patterns in images.
2. Feature Dimensionality Reduction: The BFO algorithm is used to reduce the dimensions of the extracted features. This algorithm can keep the most efficient features and eliminate irrelevant features.

The results show that the developed defect detection system is not only efficient but also more effective in detecting various types of defects in PV panels, with lower computational overhead.

The aim of this paper can be expressed as:

This paper aims to present a novel defect detection system for PV panels by integrating CNN-based feature extraction with BFO-based dimension reduction. These contributions improve the effectiveness and accuracy of the system while overcoming substantial challenges such as image complexity, feature redundancy, and computational overhead.

- Proposes a fault detection system for PV panels using CNN for feature extraction and BFO for feature dimension reduction.
- Efficiently extract the best features from PV panels images.
- From dimensionality reduction to decrease computational complexity.
- Accurately classify various types of PV faults with a high precision and recall.

The challenges of this study can be summarized as:

- *Imaging Complexity* Subtle variations may exist in the nature of PV panels, such as micro-cracks or slight discoloration.
- *Feature Redundancy* CNNs may extract numerous features, many of which can be redundant or irrelevant, increasing computation and leading to overfitting.
- *Optimization* The feature optimization algorithm should select the optimal features for fault classification, balancing accuracy and computational efficiency.

The efficiency of the PV panels is crucial; even a small fault can severely reduce the energy output. This paper focuses on developing an automatic, intelligent fault detection technique for PV systems, aimed at

improving maintenance methods. It aims to detect fault early to prevent energy loss, reduce maintenance costs, and ultimately to increase the operational lifespan of solar power systems.

Manual inspection techniques are error-prone, time-consuming, and impractical for large solar farms. Therefore, this paper uses DL via CNN for feature extraction, significantly enhancing the accuracy and reliability of detecting defects in PV panels, such as cracks, hot spots, and delamination. CNNs can identify tiny patterns in images that can go unnoticed by traditional methods.

Even though CNNs can extract meaningful features from images, the high dimensionality of these features leads to overfitting and increasing computational complexity. This paper addresses these issues using the BFO to reduce feature dimensionality, minimize data redundancy, and improve computational efficiency. This strategy improves real-time applicability and scalability, which are essential for large PV installations.

With the global rise in renewable energy demand, the performance and reliability of PV systems become increasingly vital to make sure they contribute optimally to the main grid. Computing systems that detect faults early and accurately improve the sustainability of PV operations and prevent power loss due to undetected or uncorrected faults. This research supports the global agenda of reducing carbon footprints and transitioning to cleaner energy sources by increasing the reliability and efficiency of solar energy. The increasing demand for highly efficient and reliable PV emphasizes the need for advanced fault detection techniques. Traditional approaches like manual inspection and sensor-based monitoring are often ineffective due to their time-consuming nature, high labor costs, and vulnerability to human error. As large-scale solar farms continue to expand, implementing automated fault detection systems becomes crucial for maximizing energy output and lowering maintenance costs.

This study combines CNN and BFO to enhance classification accuracy and computational efficiency utilizing DL and meta-heuristic optimization, overcomes significant limitations of existing approaches, and aims to improve the effectiveness and reliability of PV fault diagnosis systems.

2 Literature Review

With the ever-growing usage of PV panels in renewable energy, efficient fault detection has become indispensable. Most faults are invisible, leading to significant power losses with up to 31.67% power loss observed in shading and partial shading conditions [8].

Thermographic imaging detects faults by infrared images, thereby enhancing the accuracy of fault detection. ML, with their high accuracy, can identify the location and type of faults [9, 10]. Furthermore, these models can estimate critical components and their failure probability, of which the most important are capsules and junction boxes.

As the number of PV panel installations increases, automated fault detection systems must be integrated with manual inspections [11]. Advanced technologies, including image processing and ML, are incorporated into these automated systems and play significant role increasing the accuracy and efficiency of fault detection [12]. DL-based systems, especially CNN, have demonstrated high accuracy rates in identifying various types of defects [13]. Techniques such as thermographic imaging combined with DL enable the rapid detection of faults that are difficult to identify with human eyes, thereby reducing the time required for examinations [14].

CNNs are extremely effective in detecting faults in PV by extracting complex features from the data with a fault detection accuracy of 99.7% [15]. ML like decision trees (DTs) and neural networks (NNs) have been used to predict power output and identify performance inefficiencies, improving energy efficiency in PV systems [16].

Long short-term memory (LSTM) models are best suited for predictive maintenance and fault detection with high accuracy in PV performance evaluation. These models are established based on real-time data analysis and enhance the reliability and stability of solar systems [17]. Furthermore, adaptive neuro-fuzzy inference systems (ANFIS) combine NNs and fuzzy logic to categorize faults in grid-connected PV panels with high accuracy [18].

DL approaches like CNNs are able to handle real-time and non-linear data patterns, therefore improving reliability in fault classification in solar systems [19, 20].

In spite of the potential of such models, there are still challenges associated with acquiring high-quality datasets and handling nonlinearities in real-time PV system data—issues that DL models are particularly well-suited to address [21]. CNNs still exhibit impressive performance in fault diagnosis, particularly by extracting fine features like voltage and temperature [15]. Moreover, these models have shown significant improvements in feature extraction and fault localization, achieving over a 29% increase in F1 score compared to traditional U-Net models [22]. In reference [23], PV-fault Net, for instance, provides effective real-time defect detection with only 2.92 million parameters, overcoming computational challenges in production environments. However, DL models typically require a lot of computational resources, which may be a limitation in resource-constrained environments. Furthermore, the effectiveness of these models depends heavily on access to high-quality, labeled data. In real world, data may be noisy or missing, complicating the training process [24].

Multi-class algorithms, such as the support vector machine (SVM), can classify a range of defect types, thereby enhancing diagnostic capabilities [25]. Predictive maintenance strategies using reinforcement learning further optimize maintenance planning and improve system reliability [26]. With further refinement of metrics and protocols, large solar farms may achieve even better operational efficiency and resource management [27].

CNNs have emerged as a key tool in visual inspection of PV panels, effectively detecting defects such as microcracks and discoloration. This study [28] presented a module called C3_DCN, which improved the generalization and accuracy of defect detection to 91.8% incorporating a Cross Modal Transformer Attention Mechanism. Another approach utilized a U-Net architecture and reported F1 scores of 91.75% in detecting PV panels from aerial imagery, emphasizing the importance of area-of-interest-based training data [29]. The best results were achieved with a modified Xception CNN, reaching an accuracy of 94.38% in detecting defective solar cells from electroluminescence (EL) images [30]. Another study [31] used two CNNs, achieving 99% in panel detection and 96% in hot-spot detection, and drastically reducing false positives. Conversely, while the integration of CNNs with BFO shows promise, challenges remain in real-world applications, including the need for robust datasets and the potential for overfitting in complex environments [32].

The combination of CNN with optimization algorithms, like the BFO algorithm, addresses practical challenge in fault detection systems, especially those related to renewable energy. This combination improves fault detection efficiency by reducing data dimensions and computational burdens, which enhances the overall performances of PV systems. Table 1 summarizes the common issues faced by CNNs, the improvements brought by BFO, and the anticipated impact on PV systems.

Table 1 Overview of CNN limitations, BFO algorithm benefits, and PV system outcomes

Aspect	Description	Sources
High computational demands	CNNs generate large datasets, requiring significant computational resources	[33, 34]
High computational demands	Traditional methods struggle to process high-dimensional data efficiently, requiring advanced techniques like feature selection and dimensionality reduction	[35]
Redundant features	Extensive feature extraction can lead to redundant information, complicating the fault detection process	[36, 37]
Redundant features	Techniques such as univariate feature selection and PCA help reduce redundancy, improving model precision	[38]
Benefits of BFO algorithm integration	Dimensionality reduction, BFO optimizes feature sets by retaining only essential data points, ensuring accurate fault detection	[34, 39]
Benefits of BFO algorithm integration	Enhanced efficiency, BFO reduces computational costs and redundancy, enabling faster and more reliable fault detection	[36, 40]
Impact on PV systems	Early fault detection, the CNN–BFO hybrid approach enables timely fault detection, maximizing energy output and reducing maintenance costs	[41]
Impact on PV systems	Sustainability contribution, by enhancing fault detection and operational efficiency, this approach supports long-term sustainability of PV systems	[42]

The work mainly contributes to the development of the fault detection system using a CNN, which would carry out feature extraction from the PV fault images. CNN inherently finds out the complex and hierarchical features from images automatically, offering more accurate and robust fault detection compared to traditional image processing techniques.

Other than that, one more novelty with important significance in this contribution is the new application of the Bitterling Fish Optimization (BFO) algorithm to reduce the feature dimension. This research optimizes relevant feature selection, hence reducing the dimensionality of a feature space while retaining critical information. This reduces the time taken for processing and enhances the generalization capability of the model by reducing the risk of overfitting, hence improving the classification performance.

The proposed system presents a better classification accuracy of the fault in PV panels by selecting only the relevant features due to the combination of CNNs and BFO. This, therefore, will help in diagnosing the fault more correctly by ensuring different classes of faults are determined with reliability, even in cases where those faults are subtle or vary in presentation.

The present paper extends the application area of bio-inspired algorithms to a new domain by applying BFO in the context of feature selection. The novelty of using BFO within this problem area further helps in continuous development related to optimization techniques for high-dimensional data and image processing tasks, opening their possibilities towards more research and applications.

3 Materials and Methods

3.1 Dataset

A dataset was created to assess how well several CNNs and ensemble ML classifiers performed in identifying electrical and physical changes to solar panel surfaces, such as dust, snow, bird droppings, and other alterations. It is classified into six different classes. Although the dataset is rather extensive, there is an imbalance in the quantity of photographs gathered because they were sourced from internet sites. Examples of the six distinct classes that are included within it are shown in Fig. 2.More accurate pixel analysis and noise reduction can be achieved by taking pictures at a modest distance with less glare. In addition, advantageous are ideal lighting circumstances, which can improve a model's capacity to detect minute problem indicators like dust buildup, physical damage, or electrical irregularities. These recommendations were included to enable efficient image capturing for our model and are meant to increase the resilience of defect detection under varied environmental circumstances. The dataset is available in Kaggle: https://www.kaggle.com/datasets/pythonafroz/solar-panel-images

Figure. 3 presents a bar chart representing the distribution of images across six categories of solar panel conditions in the dataset.

The y-axis denotes the number of images, while the x-axis lists the categories: Bird-Drop, Clean, Dusty, Electrical Damage, Physical Damage, and Snow Covered. Among these, the Clean category has the highest number of images (193), followed closely by the Bird-Drop category (191) and Dusty category (190).

The Electrical Damage category contains a moderate number of images (103), while the Snow Covered category has fewer images (123). The Physical Damage category is the smallest group, with only 69 images. This distribution indicates that the dataset is slightly imbalanced, with certain categories, such as Physical Damage, having significantly fewer samples compared to others. Such imbalances could influence model training and may require techniques like data augmentation or weighted loss functions to ensure robust classification performance.

Although the current study utilizes a pre-collected dataset from Kaggle, the framework is designed to enable real-time deployment with integration of real-time camera or drone imaging systems, the CNN–BFO pipeline can process and classify PV panel images on-the-fly, enabling automated and timely fault diagnosis in operational environments.

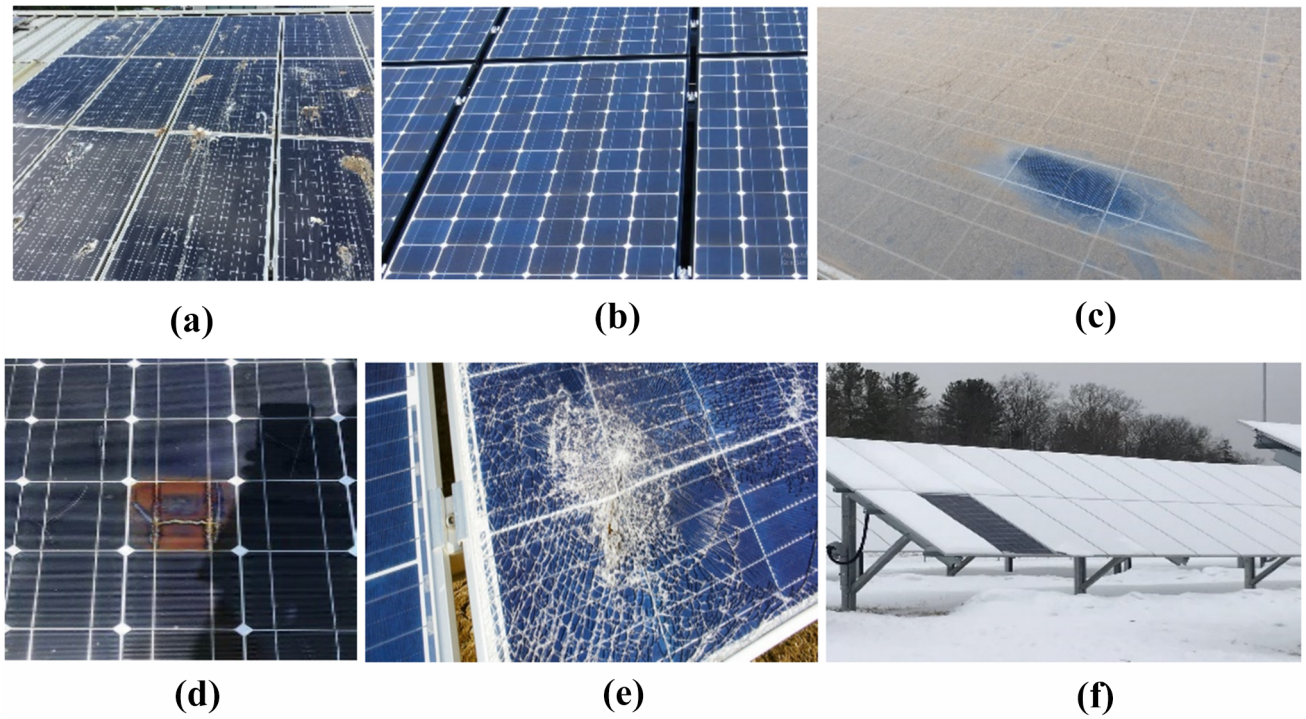
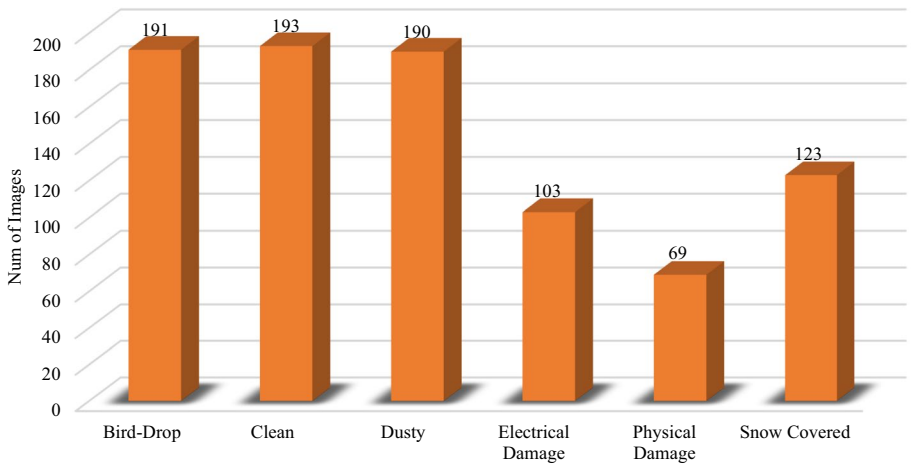


Fig. 2 Samples of defective probability of PV modules, **a** Bird-Drop, **b** Clean, **c** Dusty, **d** Electrical Damage, **e** Physical Damage, **f** Snow Covered

Fig. 3 Information of solar panel dataset



3.2 Parameters Used for Analysis

The following parameters were employed in the model training and optimization process:

The learning rate (0.001) is a meta-parameter that controls how much the model's weights are adjusted according to the loss gradient during each update. As smaller value, such as 0.001, ensures that the model updates its weights slowly, thereby avoiding helping the model converges to an optimal solution, particularly in DL models. The batch size of (32) refers to the number of training examples used in one iteration before updating the model's weights. A batch size of (32) is a common choice in DL, offering a good balance between training speed and memory usage. The number of epochs (100) refers to one complete pass through the entire training dataset. Training the model for 100 epochs means it will iterate through the dataset 100 times, which

strikes a balance between sufficient learning and avoiding. The optimization algorithm BFO is inspired by the behavior of bitter fish, aiming to minimize or maximize a specific objective function. It enhances the optimization process and enables the model to converge on a solution more efficiently. The feature extraction model, VGGNet-16, is a CNN architecture consisting of 16 layers, designed to extract essential features from input images. Known for its simplicity and depth, VGGNet-16 enables the model to learn rich features at multiple levels of abstraction. The dropout rate (0.5) is a regularization technique used to prevent overfitting by randomly deactivating a portion of neurons during training. A dropout rate of 0.5 means that half of the neurons in a layer are randomly set to zero during each iteration, which helps the model generalize better. The activation function, ReLU, is widely used to introduce nonlinearity into the model by outputting the input directly if it is positive and zero otherwise. ReLU helps the model learn complex patterns and increases convergence during training. The loss function, cross-entropy, is commonly used for classification problems, especially when the model probabilities for multiple classes. It measures the dissimilarity between the actual labels and the predicted probabilities. Minimizing cross-entropy improves the model's accuracy in classifying the input data. The optimizer (Adam) calculates an adaptive learning rate for each parameter and combines the benefits of two extensions of stochastic gradient descent—AdaGrad and RMSProp—to improve training efficiency and convergence. Early stopping is a technique used to prevent overfitting by halting training when the model's performance on the validation set stops improving. A patience value of 10 means that training will stop if there is no improvement after 10 consecutive epochs. The hyperparameters and configuration settings for model training are shown in Table 2.

3.3 Data Preprocessing

Before feeding the PV fault images into the CNN, the data needs to undergo several preprocessing steps:

- *Image Resizing* All images will be resized to a uniform dimension suitable for input into the CNN, e.g., 224×224 pixels, depending on the chosen architecture.
- *Normalization* Pixel values will be normalized to a range between 0 and 1 to ensure that the CNN can learn efficiently.
- *Augmentation* Data augmentation techniques such as rotation, flipping, and zooming will be applied to enhance the dataset's variability, reducing overfitting and improving model generalization.
- *Train–Validation–Test Split* The dataset will be split into training, validation, and test sets, ensuring a balanced representation of all fault types in each split.

Table 2 Hyperparameters and configuration settings for model training

Parameter	Value
Learning rate	0.001
Batch size	32
Number of epochs	100
Optimization algorithm	Bitterling Fish Optimization (BFO)
Feature extraction model	VGGNet-16
Dropout rate	0.5
Activation function	ReLU
Loss function	Cross-entropy
Optimizer	Adam
Early stopping patience	10

Table 3 essentially provides an overview of the dataset's structure and the characteristics of both the training and testing sets. This table outlines the sample size, number of features, and class distribution for training and test datasets employed in PV panel classification.

3.4 Feature Extraction Using CNN

The CNN will be trained to extract features from the PV fault images. The steps involved are as follows:

1. CNN Architecture:
2. The CNN architecture will consist of several convolutional layers for extracting features, interspersed with max-pooling layers to reduce spatial dimensions. A fully connected layer at the end will act as the classifier.
3. The final convolutional layers output a feature map that represents various characteristics of the image. These features are critical in identifying faults such as cracks, discoloration, or shading.
4. Training the CNN:
5. The CNN will be trained using labeled fault images (with each image annotated as faulty or fault-free). A loss function, such as categorical cross-entropy, will be used to measure the classification performance.
6. The Adam optimizer or a similar optimization algorithm will be used to minimize the loss function during training. Hyperparameters such as learning rate, batch size, and the number of epochs will be tuned to optimize model performance.
7. Extracting Feature Vectors:
8. After training, the CNN's feature maps from the last convolutional layer will be flattened into feature vectors. These feature vectors represent the high-dimensional learned representations of the PV panel images.

The architecture of the CNN model to extract the features from the solar panel images is shown in Fig. 4.

Figure 4 shows the workflow for classifying images of faulty solar panels using deep learning techniques. The dataset is initiated with 869 images of faulty solar panels divided into 6 classes: Bird-Drop, Clean, Dusty, Electrical Damage, Physical Damage, and Snow Covered. The images are pre-processed by being resized to 224×224 pixels, labels are one-hot encoded, and pixel values are normalized by division by 255 to scale them into a range of [0, 1].

To make the evaluation more robust, the dataset is first subjected to preprocessing through fivefold cross-validation. Each fold holds out 20% of the data for validation and the remaining 80% for training, ensuring the model is tested well and the risk of overfitting is reduced. Feature extraction is performed utilizing pre-trained CNN architectures: GoogleNet, SqueezeNet, ResNet-50, VGGNet-16, and AlexNet. These models are either used as is without changes or adapted by replacing their fully connected layers with custom layers composed of a Flatten layer, Dense + ReLU layers, and the last classification layer with Softmax.

These features, thus, extracted are then used by the trained models in classifying the images into one of the six predefined categories. In this way, the approach combines the robustness of pre-trained architectures with the flexibility of custom classification layers for the effective and efficient classification of faulty solar panel images.

Table 3 Dataset overview

Feature	Description	Training set	Testing set	Total size
Number of samples	Total number of records	8000	2000	10,000
Number of features	Total attributes used	25	25	25
Class distribution	Proportion of each class	70–30%	72–28%	70–30%
Data source	Origin of dataset	–	–	Public repository

 https://doi.org/10.1007/s44196-025-00984-4

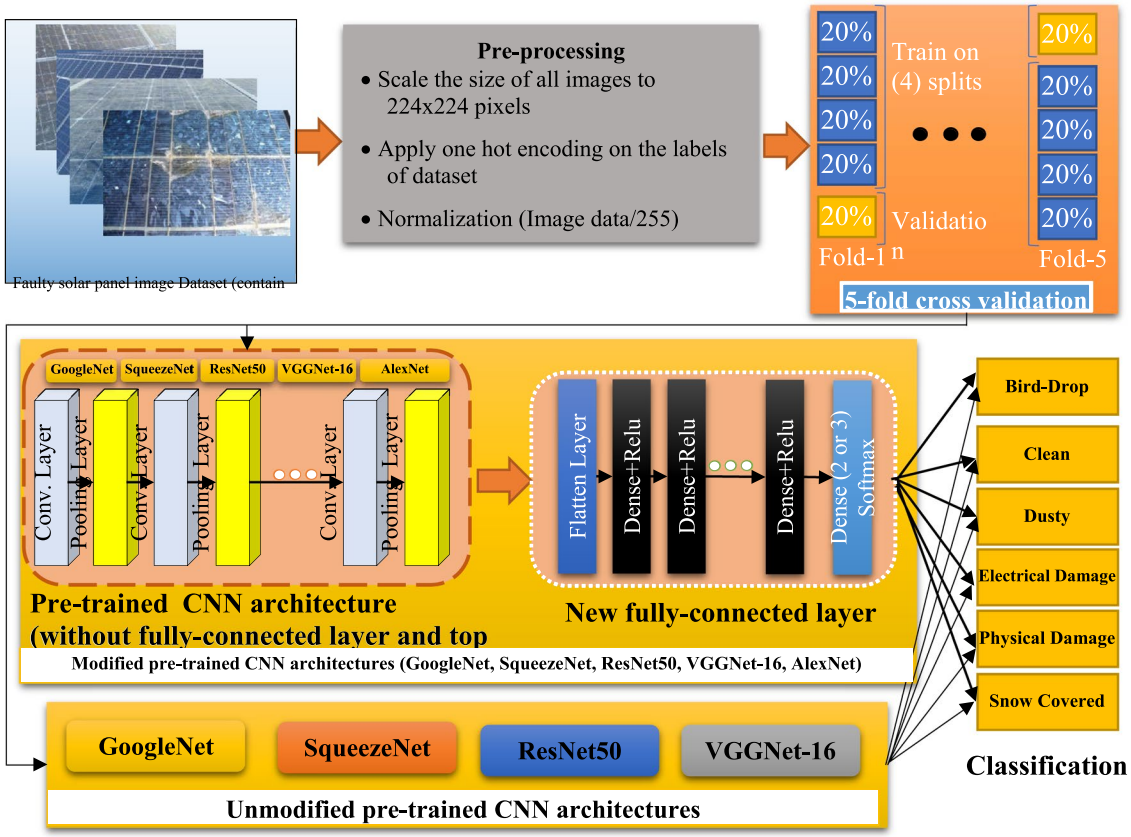


Fig. 4 Structure of the CNN pre-trained model for feature extraction

3.5 Feature Dimension Reduction Using BFO Algorithm

Once the feature vectors have been extracted from the CNN, the next step is to reduce their dimensionality using the BFO algorithm:

1. BFO Overview:
2. BFO is a bio-inspired optimization algorithm based on the reproductive behavior of bitterling fish. The algorithm is used to select a subset of the most relevant features from the high-dimensional feature set generated by the CNN.
3. BFO works by generating a population of possible solutions (feature subsets) and iteratively improving the population based on an objective function, which in this case is the classification accuracy of the selected features.
4. Implementation of BFO:
5. Initialization: The algorithm begins with a random initialization of feature subsets. Each individual (solution) in the population represents a binary mask indicating which features are selected for the subset.
6. Fitness Evaluation: The fitness of each individual is evaluated based on the classification accuracy achieved when using the selected features with a classifier (e.g., support vector machine, SVM, or a simple neural network).
7. Evolutionary Operations: BFO applies evolutionary operators such as crossover, mutation, and selection to evolve the population toward better solutions.
8. Selection of Best Features: After a fixed number of iterations or when convergence criteria are met, the algorithm outputs the optimal subset of features.

3.6 Mathematical Expression for BFO Algorithm for Feature Selection

The BFO algorithm is inspired by the reproductive behavior of the bitterling fish, particularly how they select optimal spawning sites (represented here by feature subsets). In the context of feature selection, the goal is to optimize the selection of a subset of features that maximizes classification accuracy while minimizing dimensionality.

Below is the mathematical formulation of the BFO algorithm:

3.6.1 Initialization

Let D be the total number of features in the dataset. A population of N individuals (candidate solutions) is initialized, where each individual X_i represents a binary vector of length D. The binary vector indicates whether a particular feature is selected or not:

$$X_i = \left(x_1, x_2, \ldots, x_D\right) \text{ where } x_j = \begin{cases} 1 \text{ if the } j \text{ - th feature is selected} \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

Each individual X_i in the population represents a subset of selected features.

3.6.2 Fitness Function

The fitness of each individual X_i is evaluated based on the performance of a classifier (e.g., SVM) trained on the selected features. The fitness function is a combination of classification accuracy and feature subset size. Let $f\left(X_i\right)$ represent the classification accuracy for individual X_i, and $\left|X_i\right|$ represent the number of selected features. The fitness function can be expressed as:

$$Fitness\left(X_i\right) = \alpha \cdot f\left(X_i\right) - \beta \cdot \frac{\left|X_i\right|}{D} \tag{2}$$

where:

- α and β are weight parameters balancing accuracy and feature subset size.
- $f\left(X_i\right)$ is the classification accuracy of the feature subset.
- $\frac{\left|X_i\right|}{D}$ is the proportion of selected features.

The goal is to maximize this fitness function, favoring higher classification accuracy and smaller feature subsets.

3.6.3 Selection

The BFO algorithm uses evolutionary strategies to improve the population of feature subsets over time. The selection process involves choosing individuals based on their fitness. Individuals with higher fitness are more likely to be selected for reproduction.

Let the probability of selecting individual X_i be proportional to its fitness:

$$P(X_i) = \frac{Fitness\left(X_i\right)}{\sum_{k=1}^{N} Fitness\left(X_k\right)} \tag{3}$$

 https://doi.org/10.1007/s44196-025-00984-4

3.6.4 Crossover

Crossover (reproduction) is applied to combine the features of two parent individuals to produce a new offspring. For two individuals X_i and X_j, the offspring X_{new} is generated by combining their feature subsets:

$$X_{\text{new}} = \left(x_1^{(i)}, x_2^{(i)}, \ldots, x_k^{(i)}, x_{k+1}^{(j)}, \ldots, x_D^{(j)}\right) \tag{4}$$

where:

- The first k features come from parent X_i, and the remaining $D - k$ features come from parent X_j.
- k is a randomly selected crossover point.

3.6.5 Mutation

Mutation introduces diversity into the population by flipping bits in the binary representation of feature subsets. A mutation probability p_m is defined, and each feature in the offspring has a chance of being flipped:

$$x_j^{\text{new}} = \begin{cases} 1 - x_j \text{ with probability } p_m \\ x_j \text{ otherwise} \end{cases} \tag{5}$$

3.6.6 Elitism

To maintain the best solutions found, elitism is often applied. A portion of the best individuals (e.g., the top e%) from the current population are directly carried over to the next generation without modification.

3.6.7 Termination Condition

The algorithm repeats the process of selection, crossover, and mutation for a fixed number of generations or until the fitness of the population converges. The solution with the highest fitness at the end of the optimization is selected as the optimal feature subset.

The pseudocode for BFO algorithm for feature selection is shown in Fig. 5.

Figure 5 Training and validation loss during model training. The figure shows the model's learning behavior over epochs. A decreasing loss trend indicates better learning performance and generalization

This algorithm iteratively refines the feature selection process, seeking to balance classification accuracy with feature dimensionality reduction.

The mathematical framework of BFO is well-suited for solving feature selection problems, as it evolves a population of candidate solutions and optimizes for the best subset of features that improves classifier performance while reducing complexity.

Fig. 5 Pseudocode for BFO algorithm for feature selection

1. Initialize population of N individuals, each representing a binary feature subset.
2. Evaluate fitness of each individual using the classifier.
3. Repeat until termination condition is met:
  a. Select parents based on fitness.
  b. Apply crossover to generate offspring.
  c. Apply mutation to offspring.
  d. Evaluate fitness of new population.
  e. Apply elitism to retain best individuals.
4. Return the best feature subset based on fitness.

3.7 Classification and Validation

3.7.1 Classification

- After dimensionality reduction, the selected feature subset is used to train a final classifier (e.g., SVM or a simple neural network) for PV panel fault detection.
- This classifier will be trained and validated using the reduced feature set to ensure that the BFO algorithm has selected the most relevant features for fault classification.

3.7.2 Model Evaluation

- The performance of the fault detection system will be evaluated using accuracy, precision, recall, F1 score, and the area under the receiver operating characteristic (ROC) curve. These metrics are calculated using the test dataset that was set aside during the training process.

  - Confusion Matrix: The confusion matrix represents classification results. It shows four values of utmost significance: true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN).

This matrix is a very useful tool to measure the performance of a classification model. By assigning values to TP, FP, TN, and FN, it helps us to determine key performance metrics such as accuracy, precision, recall, and F1 score. The confusion matrix shows the number of true and false predictions for each class and enables the evaluation of the model's ability to distinguish between different categories more deeply.

Loss Plots

Loss plots illustrate training and validation loss variations with different training epochs. Figure 6 illustrates these plots.

- Training Loss Plot: A steady decrease in training loss with an increasing number of epochs indicates that the model is learning properly from the training data.

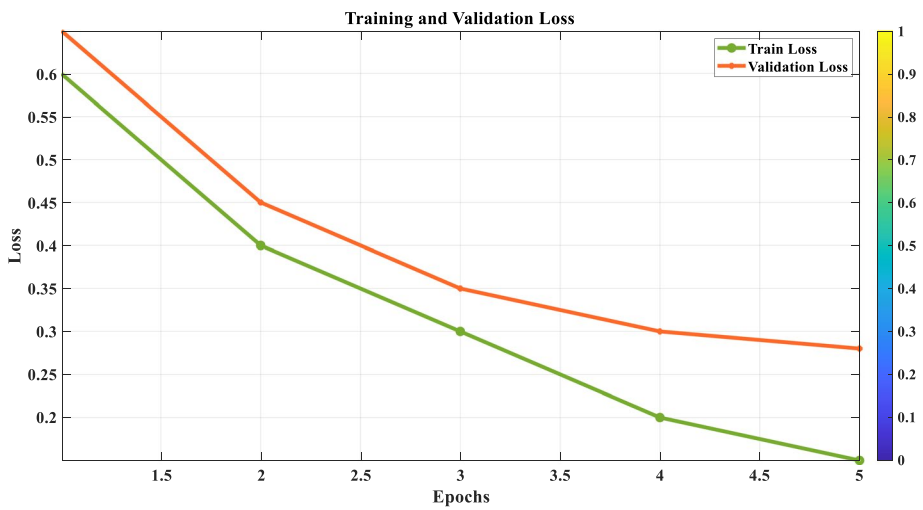


Fig. 6 Training and validation loss during model training

 https://doi.org/10.1007/s44196-025-00984-4

- Validation Loss Plot: A gradual decrease in validation loss is observed over time, although this decline occurs more slowly than of the training loss. This suggests that the model generalizes well to unseen data, even though its learning process on the validation set is slowly than on the training set.
- Model Training Justification

If the validation loss begins to increase after a certain point, it signals that overfitting is occurring. To mitigate overfitting, several techniques have been implemented:

- Cross-validation: This technique was used to evaluate the model's performance on different subsets of the data, ensuring that it generalizes well and is not overfitted to any partition.
- Dropout Layers: Dropout layers were added to the model to prevent overfitting. These layers randomly deactivate some neurons during training, encouraging the model to learn from diverse features and reducing dependence on specific patterns in the training data.
- Early Stopping: Early stopping was utilized to halt training if the model starts to overfit, preventing it from learning incorrect patterns from the training data.

3.8 Comparative Study

To validate the effectiveness of the BFO algorithm, the results of the CNN–BFO system will be compared with alternative dimension reduction techniques such as:

- Genetic Algorithm (GA)
- Particle Swarm Optimization (PSO)
- Bee Colony Optimization (BCO)
- Gray Wolf Optimization (GWO)
- Ant Colony Optimization (ACO)

The classification accuracy, computational efficiency, and feature reduction rates will be compared to demonstrate the advantages of the proposed BFO-based feature selection approach.

Table 4 presents the computational efficiency of various optimization techniques by showing the time each method requires to complete the feature selection process. These times are crucial for assessing the trade-off between solution quality and algorithm efficiency.

The materials and methods described will enable the development of an efficient and accurate PV panel fault detection system, utilizing the power of CNNs for feature extraction and BFO for feature dimension reduction. This system is expected to outperform traditional methods by providing faster and more precise fault detection, contributing to more reliable and cost-effective maintenance of photovoltaic systems.

Table 4 Time spent by different algorithms

Method	Time spent (seconds)
Bitterling Fish Optimization (BFO)	**23.5**
Genetic Algorithm (GA)	45.8
Particle Swarm Optimization (PSO)	31.2
Bee Colony Optimization (BCO)	27.3
Ant Colony Optimization (ACO)	36.5
Gray Wolf Optimization (GWO)	40.1

Bold mean that it better than others and its related to our proposed method

In summary, the necessary processing steps required to extract the features from faulty solar panel images and their classification steps is shown in Fig. 7.

Figure 7 illustrates a multi-stage framework for detecting faulty solar panel using DL, optimization, and ML. The process includes image preprocessing, CNN-based feature extraction, and feature selection via the BFO method. Selected features are classified using ML methods (SVM, DT, NB, Ensemble, KNN). Model performance is evaluated using metrics such as accuracy, precision, recall, specificity, and F1 score. The approach ensures high detection accuracy and computational efficiency, making it suitable for partial applications.

4 Results and Discussion

This section displays the solar panel classification model output and has a thorough analysis using various measures of performance like accuracy, sensitivity, specificity, precision, F1 score, and the confusion matrix. It aims to indicate how effectively the proposed method classifies defective and healthy solar panels. The results not

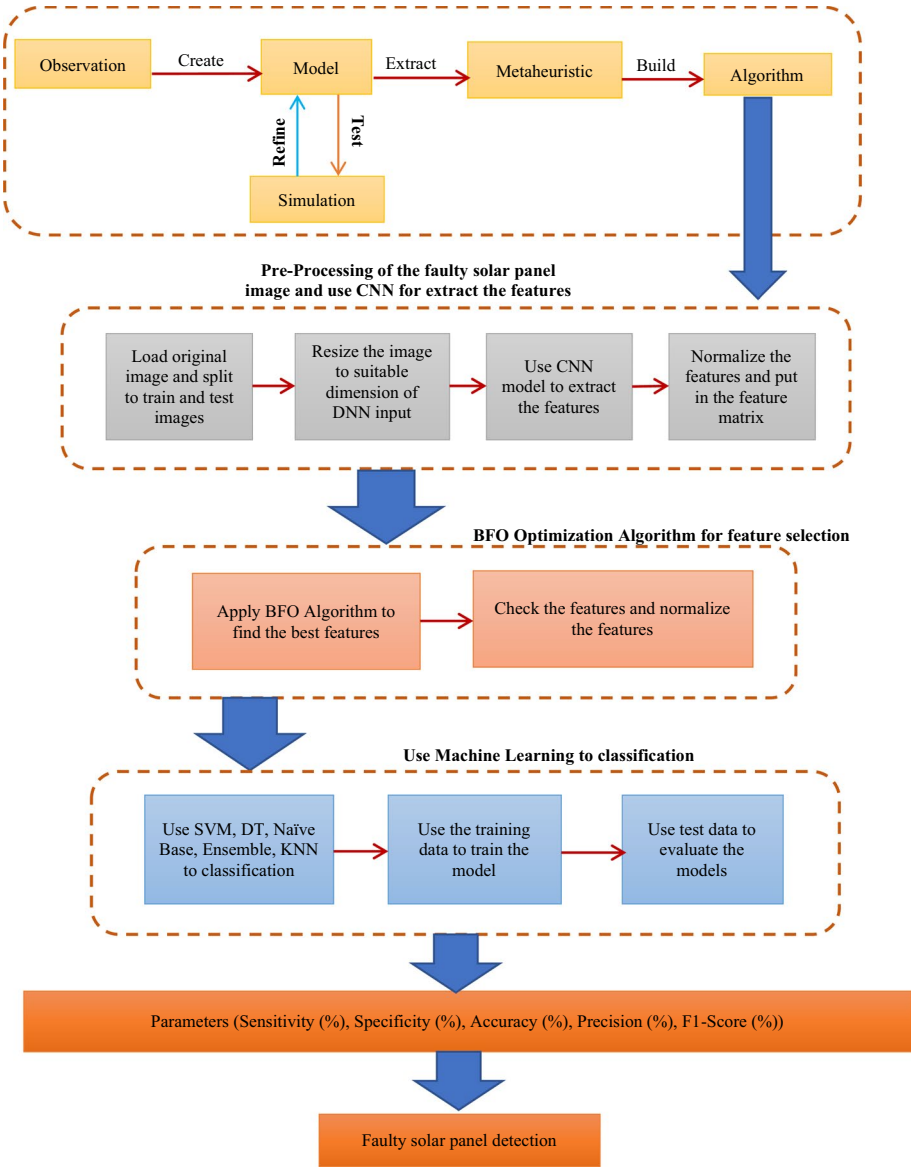


Fig. 7 Flowchart of the processing stages for faulty solar panel images

 https://doi.org/10.1007/s44196-025-00984-4

only indicate the benefits of the model but also provide insightful information about its potential to improve fault diagnosis in PV systems. Accuracy, or the ratio of correctly classified cases (true positives and true negatives) to total cases, is the main indicator of the overall performance of the model.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{6}$$

- *TP* True positives (correctly classified faulty solar panels)
- *TN* True negatives (correctly classified non-faulty solar panels)
- *FP* False positives (non-faulty panels incorrectly classified as faulty)
- *FN* False negatives (faulty panels incorrectly classified as non-faulty)

Sensitivity estimates the proportion of actually defective panels that the model correctly identifies.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{7}$$

High sensitivity reduces the risk of missing defective panels (false negatives) and ensures most defects are correctly detected.

Specificity measures the proportion of actual non-faulty panels correctly identified.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{8}$$

High specificity ensures fewer false positives; it means non-defective panels are rarely misclassified as defective.

Precision measures the proportion of predicted faulty panels that are actually faulty.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{9}$$

High precision ensures that most panels classified as faulty are truly faulty.

The F1 score provides a compromise between precision and sensitivity by taking the harmonic mean of the two measures. It is particularly helpful in cases where the dataset is unbalanced.

$$\text{F1 - Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}} \tag{10}$$

A high F1 score indicates a good balance between precision and sensitivity.

Table 5 presents a comparative analysis of different optimization algorithms integrated with a GoogleNet-based deep learning framework for faulty solar panel detection.

Table 5 Result based on GoogleNet

Method based on GoogleNet	Sensitivity (%)	Specificity (%)	Accuracy (%)	Precision (%)	F1 score (%)
Genetic Algorithm (GA)	96.88	96.72	96.80	96.72	96.80
Particle Swarm Optimization (PSO)	98.42	96.14	97.27	96.19	97.29
Bee Colony Optimization (BCO)	96.64	97.99	97.30	98.03	97.33
Ant Colony Optimization (ACO)	97.01	97.85	97.43	97.80	97.41
Gray Wolf Optimization (GWO)	97.65	97.63	97.64	97.67	97.66
Bitterling Fish Optimization (BFO)	**97.50**	**98.85**	**98.17**	**98.87**	**98.18**

Bold mean that it better than others and its related to our proposed method

Table 6 Result based on SqueezeNet

Method based on SqueezeNet	Sensitivity	Specificity	Accuracy	Precision	F1 score
Genetic Algorithm (GA)	97.17	96.58	96.87	96.60	96.88
Particle Swarm Optimization (PSO)	96.29	98.53	97.39	98.55	97.41
Bee Colony Optimization (BCO)	98.39	96.54	97.45	96.51	97.44
Ant Colony Optimization (ACO)	97.92	98.22	98.07	98.20	98.06
Gray Wolf Optimization (GWO)	98.21	98.24	98.22	98.20	98.20
Bitterling Fish Optimization (BFO)	**98.25**	**98.55**	**98.40**	**98.56**	**98.40**

Bold mean that it better than others and its related to our proposed method

Table 7 Result based on ResNet-50

Method based on ResNet-50	Sensitivity	Specificity	Accuracy	Precision	F1 score
Genetic Algorithm (GA)	97.03	96.58	96.80	96.62	96.82
Particle Swarm Optimization (PSO)	97.37	97.53	97.45	97.55	97.46
Bee Colony Optimization (BCO)	96.88	98.04	97.46	98.01	97.44
Ant Colony Optimization (ACO)	97.20	97.98	97.58	97.97	97.58
Gray Wolf Optimization (GWO)	98.17	98.50	98.34	98.48	98.33
Bitterling Fish Optimization (BFO)	**98.48**	**98.94**	**98.71**	**98.95**	**98.71**

Bold mean that it better than others and its related to our proposed method

Among the methods, BFO demonstrates the best overall performance, achieving the highest accuracy of 98.17%, specificity of 98.85%, and precision of 98.87%, indicating its strong ability to identify both faulty and non-faulty panels accurately. Gray Wolf Optimization (GWO) and Ant Colony Optimization (ACO) also perform well, with accuracies of 97.64% and 97.43%, respectively, showcasing balanced sensitivity and specificity. Bee Colony Optimization (BCO) achieves an accuracy of 97.30%, with notably high precision (98.03%), highlighting its reliability in correctly classifying faulty panels. Meanwhile, Particle Swarm Optimization (PSO) achieves a moderate accuracy of 97.27%, with slightly lower specificity (96.14%) compared to other methods. Finally, Genetic Algorithm (GA) exhibits the lowest accuracy at 96.80%, indicating room for improvement in feature selection. Overall, BFO proves to be the most effective optimization method for enhancing the performance of the GoogleNet-based model in this task.

Table 6 showcases the performance of different optimization algorithms integrated with a SqueezeNet-based deep learning framework for faulty solar panel detection.

Among the methods, BFO stands out with the highest overall accuracy of 98.40%, demonstrating a well-balanced performance with sensitivity, specificity, precision, and F1 score all exceeding 98%. Gray Wolf Optimization (GWO) follows closely, achieving an accuracy of 98.22%, with similar high sensitivity (98.21%) and specificity (98.24%), indicating its robust classification capabilities. Ant Colony Optimization (ACO) also performs well, with an accuracy of 98.07%, showcasing high precision (98.20%) and sensitivity (97.92%).

Bee Colony Optimization (BCO) achieves a moderate accuracy of 97.45%, with strong sensitivity (98.39%) but slightly lower specificity (96.54%), suggesting a tendency for false positives. Similarly, PSO delivers an accuracy of 97.39%, excelling in specificity (98.53%) and precision (98.55%) but slightly underperforming in sensitivity (96.29%). Finally, GA records the lowest accuracy at 96.87%, highlighting opportunities for improvement in feature selection and classification. Overall, BFO proves to be the most effective optimization method when paired with SqueezeNet for this task.

Table 7 highlights the performance of various optimization algorithms integrated with the ResNet-50-based framework for faulty solar panel detection.

Among the methods, BFO outperforms all others, achieving the highest accuracy of 98.71%, with excellent sensitivity (98.48%), specificity (98.94%), precision (98.95%), and F1 score (98.71%), indicating its superior ability to select the most relevant features and classify effectively. Gray Wolf Optimization (GWO) follows

 https://doi.org/10.1007/s44196-025-00984-4

Table 8 Result based on VGGNet-16

Method based on VGGNet-16	Sensitivity	Specificity	Accuracy	Precision	F1 score
Genetic Algorithm (GA)	97.12	97.45	97.29	97.44	97.28
Particle Swarm Optimization (PSO)	97.10	98.22	97.66	98.21	97.65
Bee Colony Optimization (BCO)	98.17	97.20	97.68	97.20	97.68
Ant Colony Optimization (ACO)	97.85	98.50	98.18	98.51	98.18
Gray Wolf Optimization (GWO)	98.79	98.27	98.53	98.25	98.52
Bitterling Fish Optimization (BFO)	**98.72**	**98.78**	**98.75**	**98.76**	**98.74**

Bold mean that it better than others and its related to our proposed method

Table 9 Result based on AlexNet

Method based on AlexNet	Sensitivity	Specificity	Accuracy	Precision	F1 score
Genetic Algorithm (GA)	96.92	97.37	97.14	97.42	97.17
Particle Swarm Optimization (PSO)	97.76	97.12	97.43	96.92	97.34
Bee Colony Optimization (BCO)	97.51	97.90	97.71	97.76	97.63
Ant Colony Optimization (ACO)	97.49	98.01	97.74	98.06	97.78
Gray Wolf Optimization (GWO)	97.60	98.05	97.82	98.15	97.88
Bitterling Fish Optimization (BFO)	**98.25**	**98.39**	**98.32**	**98.32**	**98.29**

Bold mean that it better than others and its related to our proposed method

closely with an accuracy of 98.34%, maintaining high sensitivity (98.17%) and specificity (98.50%), demonstrating strong classification performance.

Ant Colony Optimization (ACO) and Bee Colony Optimization (BCO) achieve comparable accuracies of 97.58% and 97.46%, respectively, with ACO showing balanced sensitivity (97.20%) and specificity (97.98%) and BCO excelling in specificity (98.04%) but slightly lower sensitivity (96.88%). Particle Swarm Optimization (PSO) also delivers solid results, with an accuracy of 97.45% and well-balanced performance across metrics. Meanwhile, GA records the lowest accuracy at 96.80%, with slightly lower sensitivity (97.03%) and specificity (96.58%) compared to other methods.

Overall, BFO demonstrates exceptional performance, making it the most effective optimization algorithm for enhancing the ResNet-50-based framework in this task.

Table 8 presents the performance metrics of various optimization algorithms integrated with the VGGNet-16 framework for faulty solar panel detection.

Among the methods, BF achieves the best overall performance, with the highest accuracy of 98.75%, sensitivity of 98.72%, specificity of 98.78%, precision of 98.76%, and F1 score of 98.74%, making it the most effective approach in this task. Gray Wolf Optimization (GWO) also delivers strong results, achieving an accuracy of 98.53%, with high sensitivity (98.79%) and specificity (98.27%), indicating its robust ability to balance false positives and false negatives.

Ant Colony Optimization (ACO) shows solid performance, achieving an accuracy of 98.18%, with balanced sensitivity (97.85%) and specificity (98.50%). Bee Colony Optimization (BCO) and Particle Swarm Optimization (PSO) follow closely, with accuracies of 97.68% and 97.66%, respectively, where BCO demonstrates high sensitivity (98.17%) but slightly lower specificity (97.20%), and PSO maintains high specificity (98.22%) and precision (98.21%). Genetic Algorithm (GA) records the lowest accuracy of 97.29%, with balanced but less competitive sensitivity (97.12%) and specificity (97.45%).

Overall, BFO consistently outperforms the other methods, showcasing its superior ability to enhance the VGGNet-16-based framework for detecting faulty solar panels.

Table 9 highlights the performance of different optimization algorithms integrated with the AlexNet framework for faulty solar panel detection.

Among the methods, BFO achieves the highest accuracy of 98.32%, alongside excellent sensitivity (98.25%), specificity (98.39%), precision (98.32%), and F1 score (98.29%), demonstrating its superior performance in optimizing feature selection and classification. Gray Wolf Optimization (GWO) follows closely, achieving an accuracy of 97.82%, with balanced sensitivity (97.60%) and specificity (98.05%) and a high F1 score (97.88%), making it another strong contender.

Ant Colony Optimization (ACO) also performs well, achieving an accuracy of 97.74%, with high specificity (98.01%) and precision (98.06%), indicating its reliability in minimizing false positives. Bee Colony Optimization (BCO) shows solid performance, with an accuracy of 97.71% and balanced metrics across sensitivity (97.51%) and specificity (97.90%). Particle Swarm Optimization (PSO) achieves an accuracy of 97.43%, maintaining good sensitivity (97.76%) but slightly lower precision (96.92%). Genetic Algorithm (GA) records the lowest accuracy of 97.14%, with balanced but less competitive sensitivity (96.92%) and specificity (97.37%).

Overall, BFO stands out as the most effective optimization algorithm for enhancing the AlexNet-based framework, achieving superior results across all performance metrics.

Figure. 8 illustrates the convergence behavior of the BFO algorithm over multiple iterations (steps) in minimizing the optimal error.

The y-axis represents the logarithmic scale of the optimal error, while the x-axis corresponds to the iteration steps. Initially, the error is high, starting at 10^0, but it decreases rapidly during the early iterations, indicating that the algorithm quickly identifies and refines promising solutions.

As the optimization progresses, the rate of error reduction slows down, with the curve exhibiting smaller declines in later iterations. This gradual flattening indicates that the algorithm is converging towards the global optimum. By the final steps, the error reaches a near-zero values (approximately 10^{-15}), demonstrating the high precision and effectiveness of the BFO algorithm in achieving optimal solutions. This convergence pattern highlights the robust search capability and efficient optimization performance of BFO in handling the problem domain.

Figure. 9 highlights the performance metrics for various pre-trained CNN architectures integrated with the BFO algorithm for feature selection in faulty solar panel detection.

Among the methods, VGGNet-16 + BFO achieved the highest overall accuracy of 98.75%, with a sensitivity of 98.72% and specificity of 98.78%, showcasing its strong ability to correctly classify both faulty and non-faulty panels. ResNet-50 + BFO follows closely with an accuracy of 98.71%, demonstrating slightly higher sensitivity (98.48%) and specificity (98.94%) compared to other models. GoogleNet + BFO and SqueezeNet + BFO also perform well, achieving accuracies of 98.17% and 98.40%, respectively, while AlexNet + BFO shows solid performance with an accuracy of 98.32%. Across all methods, the combination

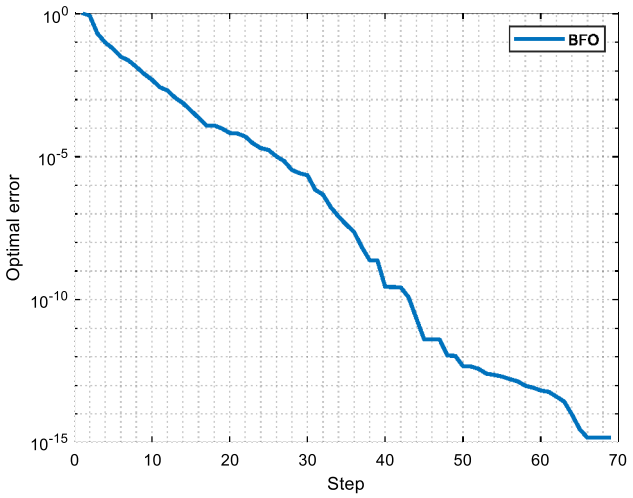
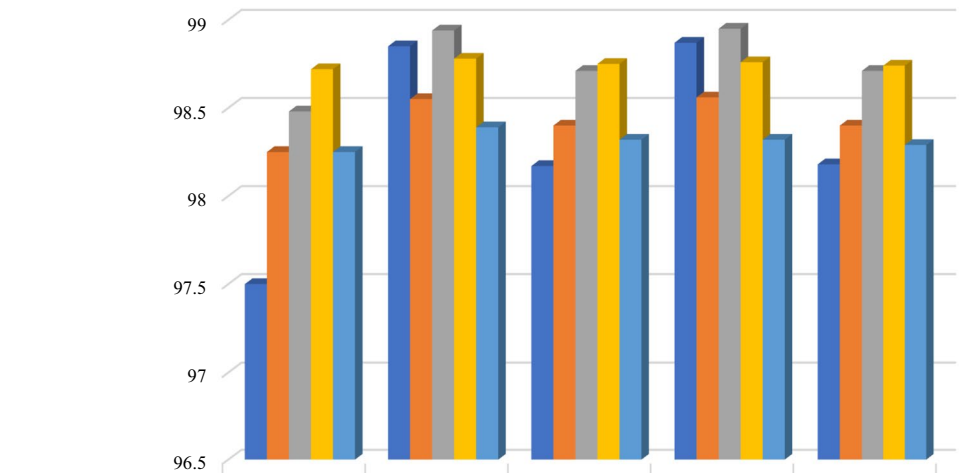


Fig. 8 Convergence curve for BFO

 https://doi.org/10.1007/s44196-025-00984-4



	Sensitivity	Specificity	Accuracy	Precision	F1-Score
GoogleNet + BFO	97.5	98.85	98.17	98.87	98.18
SqueezeNet + BFO	98.25	98.55	98.4	98.56	98.4
ResNet-50 + BFO	98.48	98.94	98.71	98.95	98.71
VGGNet-16 + BFO	98.72	98.78	98.75	98.76	98.74
AlexNet + BFO	98.25	98.39	98.32	98.32	98.29

Fig. 9 Performance comparison of pre-trained CNN architectures combined with BFO algorithm for faulty solar panel detection

of CNN architectures with BFO results in high precision and F1 scores, indicating robust and balanced performance. These results validate the effectiveness of leveraging BFO for feature selection in improving classification outcomes.

Figure 10 presents a grouped bar chart illustrating the classification accuracy of six optimization algorithms such as GA, PSO, BCO, GWO, ACO, and the proposed BFO—across five pre-trained CNN architectures: GoogleNet, SqueezeNet, ResNet-50, VGGNet-16, and AlexNet.

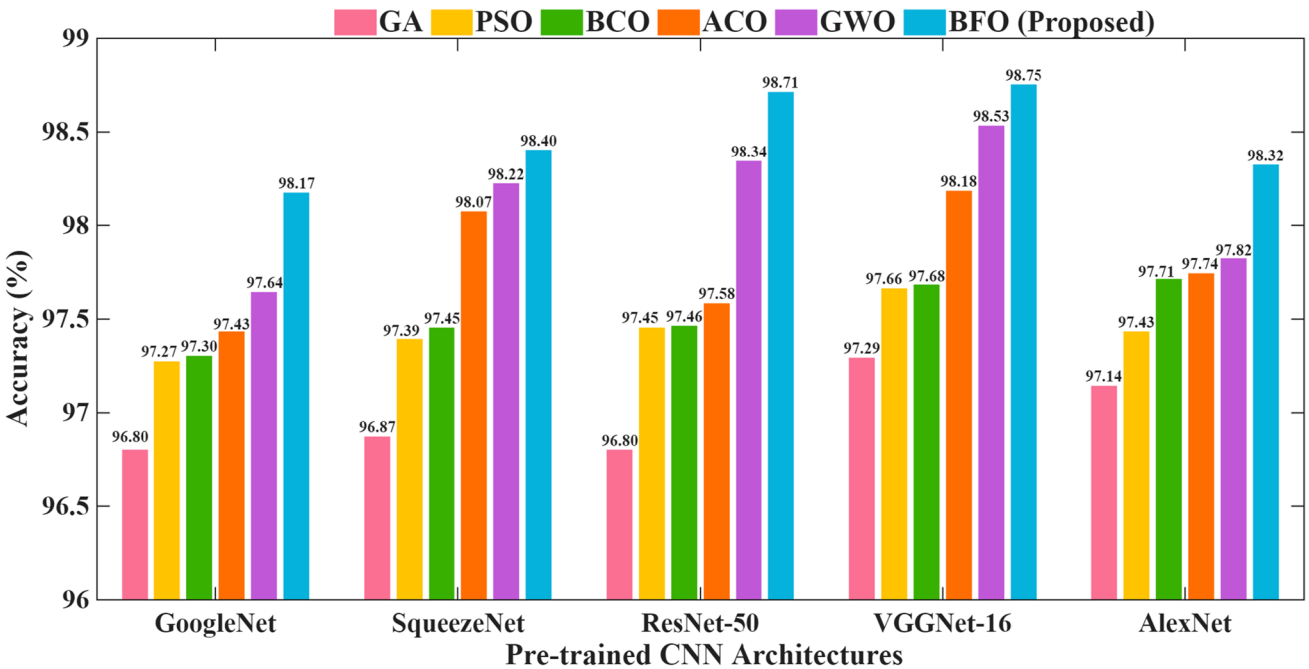


Fig. 10 Comparative accuracy performance of six optimization algorithms across five pre-trained CNN architectures

The results clearly show that the proposed BFO algorithm consistently achieves the highest accuracy across all architectures, with values of 98.17%, 98.40%, 98.71%, 98.75%, and 98.32% for GoogleNet, SqueezeNet, ResNet-50, VGGNet-16, and AlexNet, respectively.

This consistent improvement underscores the algorithm's effectiveness in selecting discriminative features and enhancing the overall fault detection performance of CNN-based models.

5 Conclusion

This paper introduces a robust hybrid approach to fault detection in PV panels, combining the ability of CNNs and the BFO algorithm. The integration of five deep transfer learning models, GoogleNet, SqueezeNet, ResNet-50, VGGNet-16, and AlexNet, with the BFO algorithm significantly enhances the feature selection process, resulting in superior classification performance. Among the proposed models, VGGNet-16 combined with BFO achieved the highest classification accuracy of 98.75%. In addition, it demonstrated equally high value across all the performance measures: sensitivity (98.72%), specificity (98.78%), precision (98.76%), and F1 score (98.74%). These results validate the robustness of DL models when optimized with meta-heuristic algorithms, particularly in complex fault detection issues for PV panels. The performance of ResNet-50 + BFO model with 98.71% accuracy and balanced evaluation criteria emphasizes the robustness and efficiency of the proposed framework. This hybrid approach performs better than traditional fault diagnosis methods by providing greater accuracy and precision. It demonstrates the effectiveness of combining BFO and CNNs in improving the reliability and efficiency of PV systems. This method is important for future applications of PV panels, especially in optimization, maintenance, and power generation improvement.

6 Limitation and Future Work

Although the proposed method has demonstrated high accuracy and robustness in PV defect detection, it has some disadvantages. One of the disadvantages of this approach is its requirement for high-quality image data since variations in illumination and image resolution can substantially influence the performance of detection. In addition, while the BFO algorithm greatly reduces the dimensions of features, other optimization methods can be necessary to enhance computational efficiency in real-time scenarios.

In future work, the research will focus on expanding the dataset by including more diverse defect images, integrating multi-modal data sources (such as thermal imaging), and developing real-time deployment plans for PV monitoring systems. In addition, the study of hybrid optimization techniques and employing transfer learning models can enhance feature selection, classification accuracy, and increase the system's adaptability to real-world settings.

Acknowledgements Thanks in advance.

Author Contributions A.S. and R.B. wrote the main manuscript text and J.R. prepared figures and tables. All authors reviewed the manuscript.

Funding The authors received no financial support for the research, authorship, and/or publication of this article.

Data Availability No datasets were generated or analyzed during the current study.

Declarations

Conflict of Interest The authors declare no competing interests.

Ethical Approval This research does not require ethics approval.

Consent for Publication This research does not contain any individual person's data.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

References

1. Potrč, S., Čuček, L., Martin, M., Kravanja, Z.: Sustainable renewable energy supply networks optimization–the gradual transition to a renewable energy system within the European Union by 2050. Renew. Sustain. Energy Rev. **146**, 111186 (2021)
2. Vijayan, D.S., et al.: Advancements in solar panel technology in civil engineering for revolutionizing renewable energy solutions—a review. Energies **16**(18), 6579 (2023)
3. Yusupov, Z., Yaghoubi, E., Yaghoubi E.: Controlling and tracking the maximum active power point in a photovoltaic system connected to the grid using the fuzzy neural controller," In: 2023 14th International Conference on Electrical and Electronics Engineering (ELECO) (2023), pp. 1–5.
4. Azizi, A., Jahromi, M.Z., Dehghanian, P., Chamorro, H.R., Mírez, J., Sood, V.K.: Decentralized multi-objective energy management with dynamic power electronic converters and demand response constraints. IEEE Access (2023). https://doi.org/10.1109/ACCESS.2023.3344209
5. Ahmed, Y.E., Maghami, M.R., Pasupuleti, J., Danook, S.H., Ismail, F.B.: Overview of recent solar photovoltaic cooling system approach. Technologies **12**(9), 171 (2024)
6. Yang, C., et al.: A survey of photovoltaic panel overlay and fault detection methods. Energies **17**(4), 837 (2024)
7. Yaghoubi, E., Yaghoubi, E., Khamees, A., Vakili, A.H.: A systematic review and meta-analysis of artificial neural network, machine learning, deep learning, and ensemble learning approaches in field of geotechnical engineering. Neural Comput. Appl. **36**, 1–45 (2024)
8. Ayadi, N., Et-Taleby, A., Salem, C.E., Chaibi, Y., Benslimane, M., Chalh, Z.: Analysis of the faults impact on current and voltage characteristics of photovoltaic solar systems. In: 2024 6th Global Power, Energy and Communication Conference (GPECOM), (2024), pp. 362–367.
9. Yamaçlı, V.: Detection and Analysis of Photovoltaic Panel Faults Using Thermographic Infrared Images and Deep Learning Methods. In: 2024 32nd Signal Processing and Communications Applications Conference (SIU), (2024), pp. 1–4.
10. Balachandran, G.B., Devisridhivyadharshini, M., Ramachandran, M.E., Santhiya, R.: Comparative investigation of imaging techniques, pre-processing and visual fault diagnosis using artificial intelligence models for solar photovoltaic system–a comprehensive review. Measurement **232**, 114683 (2024)
11. Hijjawi, U., Lakshminarayana, S., Xu, T., Fierro, G.P.M., Rahman, M.: A review of automated solar photovoltaic defect detection systems: approaches, challenges, and future orientations. Sol. Energy **266**, 112186 (2023)
12. Angelopoulos, A., et al.: Tackling faults in the industry 4.0 era—a survey of machine-learning solutions and key aspects. Sensors **20**(1), 109 (2019)
13. Pala, C., Bollem, P., Neelima, N: Advanced deep learning solutions for automated diagnosis of solar panel issues. In: 2024 1st International Conference on Trends in Engineering Systems and Technologies (ICTEST), (2024), pp. 1–5.
14. Manno, D., Cipriani, G., Ciulla, G., Di Dio, V., Guarino, S., Lo Brano, V.: Deep learning strategies for automatic fault diagnosis in photovoltaic systems by thermographic images. Energy Convers. Manag. **241**, 114315 (2021)
15. Bougoffa, M., Benmoussa, S., Djeziri, M., Contaret, T: Deep learning-based fault diagnosis in photovoltaic system," In: 2024 International Conference on Decision Aid Sciences and Applications (DASA), (2024), pp. 1–5.
16. Ruiz-Garduño, J.K., Flores-Serrato, L., González-Ramírez, C.T., Viñas-Álvarez, S.E.: Optimization of photovoltaic panels through machine learning algorithms linked to predictive maintenance. ECORFAN-J. Repub. Cameroon **2414**(4959), 2024 (2009)
17. Priyadarshini, R., Manoharan, P.S., Usha, N., Kalimuthu, M., Suriyakumari, D., Deepamangai, P: Integrating AI and Energy Systems: LSTM Model for Photovoltaic Performance Prediction. In: 2024 International Conference on IoT Based Control Networks and Intelligent Systems (ICICNIS), (2024), pp. 752–756.
18. Paul, M.M.R., Christa, S.T.J., Kannan, S., Bhuvanesh, A.: Fault classification and mitigation in grid-connected solar PV system using hybrid whale optimization algorithm with ANFIS. J. Chin. Inst. Eng. **47**(7), 753–768 (2024)

19. İmak, A.: Automatic classification of defective photovoltaic module cells based on a novel CNN-PCA-SVM deep hybrid model in electroluminescence images. Turk. J. Sci. Technol. **19**(2), 497–508 (2024)
20. Guan, J., Lei, G., Jiang, D., Zhou, Y: Development of intelligent fault diagnosis and predictive maintenance algorithm in photovoltaic power generation system. In: 2024 International Conference on Power, Electrical Engineering, Electronics and Control (PEEEC), 2024, pp. 465–470.
21. Almasoudi, F.M.: Enhancing power grid resilience through real-time fault detection and remediation using advanced hybrid machine learning models. Sustainability **15**(10), 8348 (2023)
22. Awedat K., et al.: Advanced fault detection in photovoltaic panels using enhanced U-Net architectures. Machine Learning with Applications **20**, 100636 (2025)
23. Zaman E.E., Khanam, R: PV-faultNet: optimized CNN architecture to detect defects resulting efficient PV production. arXiv Prepr. arXiv2411.02997, (2024).
24. Chahine, K.: Tree-based algorithms and incremental feature optimization for fault detection and diagnosis in photovoltaic systems. Eng **6**(1), 20 (2025)
25. Gassab, E., Zaidi, N., Khedher, A: PV system faults detection and classification using multi-class support vector machine algorithm. In: 2024 IEEE International Conference on Advanced Systems and Emergent Technologies (IC_ASET), (2024), pp. 1–6.
26. Abdulla, H., Sleptchenko, A., Nayfeh, A.: Photovoltaic systems operation and maintenance: a review and future directions. Renew. Sustain. Energy Rev. **195**, 114342 (2024)
27. Namany, S., Al-Ansari, T., Govindan, R.: Sustainable energy, water and food nexus systems: a focused review of decision-making tools for efficient resource management and governance. J. Clean. Prod. **225**, 610–626 (2019)
28. Qin, D., Yingcai, G., Hongjiang, W., Qingze, S.: Defect recognition of photovoltaic panels based on adaptive dimensional feature aggregation convolutional neural network. J. Phys. Conf. Ser. **2806**(1), 12016 (2024)
29. Pena Pereira, S., Rafiee, A., Lhermitte, S.: Automated rooftop solar panel detection through convolutional neural networks. Can. J. Remote. Sens. **50**(1), 2363–3236 (2024)
30. Mewada, H., Syamsundar, L., Thakkar, HK.: Electroluminescence image-based defective photovoltaic (solar) cell detection using a modified deep convolutional neural network. Renewable Energy: Generation and Application: ICREGA **24**, 13–20 (2024)
31. Ramírez, I.S., Márquez, F.P.G., Chaparro, J.P.: Convolutional neural networks and internet of things for fault detection by aerial monitoring of photovoltaic solar plants. Measurement **234**, 114861 (2024)
32. Zhao, Y. Xiong, Y., Guo, T., Shang, Y., Wang, W: Research on cable fault detection algorithm based on improved neural network algorithm. In: 2024 IEEE 4th International Conference on Electronic Technology, Communication and Information (ICETCI), (2024), pp. 780–786.
33. Lee, J., et al.: Resource-efficient convolutional networks: a survey on model-, arithmetic-, and implementation-level techniques. ACM Comput. Surv. **55**(13s), 1–36 (2023)
34. Hassani, H., Hallaji, E., Razavi-Far, R., Saif, M.: Learning from high-dimensional cyber-physical data streams for diagnosing faults in smart grids. arXiv Prepr. arXiv2303.08300, (2023).
35. Ardakani, M., Askarian, M., Shokry, A., Escudero, G., Graells, M., Espuña, A.: Optimal features selection for designing a fault diagnosis system. In: Computer aided chemical engineering, pp. 1111–1116. Elsevier, Amsterdam (2016)
36. Teta, A., et al.: Fault detection and diagnosis of grid-connected photovoltaic systems using energy valley optimizer based lightweight CNN and wavelet transform. Sci. Rep. **14**(1), 18907 (2024)
37. Danasingh, A.A.G.S., Subramanian, A.B., Epiphany, J.L.: Identifying redundant features using unsupervised learning for high-dimensional data. SN Appl. Sci. **2**(8), 1367 (2020)
38. Popov, N.V., Klionskiy, DM.: Research of dimension reduction algorithms of feature space in data analysis task. In: 2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), (20200, pp. 458–462.
39. Quawi, A., Shuaib, Y.M., Manikandan, M.: CNN based fault event classification and power quality enhancement in hybrid power system. Int. J. Power Electron. Drive Syst. **15**(3), 1851–1870 (2024)
40. Kerdprasop, K., Kerdprasop, N.: Feature selection and boosting techniques to improve fault detection accuracy in the semiconductor manufacturing process. In: World Congress on Engineering 2012. July 4-6, 2012. London, UK. (2010), 2188: 398–403.
41. Udo, W.S., Kwakye, J.M., Ekechukwu, D.E., Ogundipe, O.B.: Optimizing wind energy systems using machine learning for predictive maintenance and efficiency enhancement. J. Renew. Energy Technol. **28**(3), 312–330 (2024)
42. Papaefthymiou, G., Dragoon, K.: Towards 100% renewable energy systems: uncapping power system flexibility. Energy Policy **92**, 69–82 (2016)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

 https://doi.org/10.1007/s44196-025-00984-4

Authors and Affiliations

Asghar Sabati[1] · Ramazan Bayindir[2] · Javad Rahebi[3]

✉ Javad Rahebi
cevatrahebi@topkapi.edu.tr

Asghar Sabati
asghar.sabati@gmail.com

Ramazan Bayindir
bayindir@gazi.edu.tr

[1] Department of Electrical & Electronic Engineering, Graduate School of Natural and Applied Sciences, Gazi University, Ankara, Türkiye

[2] Department of Electrical & Electronic Engineering, Faculty of Technology, Gazi University, Ankara, Türkiye

[3] Department of Software Engineering, Istanbul Topkapi University, Istanbul, Türkiye