



Machine vision based fault diagnosis of photovoltaic modules using lazy learning approach

Naveen Venkatesh S, Sugumaran V*

School of Mechanical Engineering (SMEC), Vellore Institute of Technology, Chennai 600127, India

ARTICLE INFO

Keywords:
 Fault diagnosis
 Photovoltaic modules
 Locally weighted learning
 Convolutional neural networks
 K-star classifier
 Nearest neighbor
 K-nearest neighbor

ABSTRACT

Machine Vision is an advanced and powerful imaging based technique that has been applied in various fields like robotics, inspection and process control. Machine vision, in industrial terms, is termed as a subcategory of computer vision. The primary aim of the proposed study is to distinguish various visual fault conditions that hinder the performance and life span of photovoltaic (PV) modules using computer vision and a machine learning approach. Literatures state that thermographic and electroluminescence images were used in deep learning for identifying faults in photovoltaic modules. However, the effectiveness of normal RGB images with the fusion of deep learning and machine learning techniques is less explored. This paper deals with the classification of normal RGB images of PV modules acquired using a fusion of deep learning and machine learning techniques. Visual faults like delamination, snail trail, burn marks, glass breakage and discoloration that occur in a photovoltaic module (PVM) were considered in the study. A machine learning approach was used to handle this problem which contains three stages: (i) feature extraction, (ii) feature selection and (iii) feature classification. Initially, the features from the aerial images of PVM (acquired from unmanned aerial vehicles (UAVs) equipped with digital cameras) were extracted using convolutional neural networks (CNN). Secondly, J48 decision tree algorithm was employed to select the features of utmost significance and dominance from the extracted image features. Finally, a set of lazy classifiers like locally weighted learning (LWL), K-star algorithm (KS), nearest neighbor (NN) and k-nearest neighbor (kNN) were adopted to execute the classification task on the selected image features. The classification accuracies of all the aforementioned classifiers were compared and it was found that the k-nearest neighbor classifier achieved a maximum accuracy of 98.95% with a lesser computational time of 0.04 s.

1. Introduction

Solar energy is claimed to be a promising alternative for fossil fuel energy production to provide clean and low-carbon electricity worldwide. Adopting renewable energy sources can be an excellent solution for the rising concerns of climatic changes and environmental pollution due to fossil fuel energy production [1]. Wide accessibility and abundant solar energy availability have made the photovoltaic (PV) industry more prominent among other available renewable energy sources [2]. The photovoltaic module (PVM) is an essential component used to convert solar energy to electricity. The increase in global PV installations (expected to be 438 GW by 2022) prove that solar energy is the extensively used and most preferred form of renewable energy. Also, PVM has seen a drastic drop in the price (by 99% over the last three decades), attracting many industrialists and capitalists to opt for solar energy power

production [3]. However, several challenges are still faced by the PV industry in recent times: (i) reliability of modules, (ii) power degradation, (iii) cost of initial installation, (iv) occurrence of faults, (v) fluctuating outdoor environmental conditions and (vi) various other necessary characteristics. PVM is operated outdoors and exposed to varying harsh climatic conditions, making them susceptible to several fault occurrences such as delamination, snail trail, burn marks, glass breakage and discoloration [4]. Such fault occurrences can lead to various safety concerns, thereby affecting the reliability, power output and life span of PVM. Recent reports reveal that the overall loss in power output of PVM due to fault occurrences accounts for 18.9% annually [5]. On-time and accurate diagnosis of faults can considerably improve power capacity of the module, life span, reliability and ensure safe operation of PV systems. The prime objective of fault diagnosis is to identify and categorize the faults such that the appropriate preventive

* Corresponding author.

E-mail addresses: naveenashrae1@gmail.com (S. Naveen Venkatesh), v_sugu@yahoo.com (V. Sugumaran).

measures can be arranged on time. Hence, fault diagnosis of PVM is an important activity. During early times, fault diagnosis in PVM was carried out by skilled personnel through visual inspections. In recent years, owing to the advancements in technology, more advanced and non-destructive techniques such as thermographic assessments, photoluminescence imaging, electroluminescence imaging, electrical measurements etc., were applied for fault diagnosis in PVM [6]. Some drawbacks like higher time consumption, non – feasibility in large installed capacities and excessive manpower requirement have created the need for an advanced fault diagnosis technique.

Usage of unmanned aerial vehicles (UAVs) to monitor the occurrence of faults is gaining interest among industrialists and capitalists. Due to the rapid development and versatility of UAV technology, it has been used in several applications like surveillance, search and rescue, disaster relief, photography, large-scale inspections and photography. The major reasons to opt for automated remote-controlled systems are (i) minimal human interference, (ii) non-destructive nature and (iii) reduced time consumption. UAVs usage and application in fault diagnosis have been discussed in several literatures. Thermal cameras installed in UAVs are widely used in fault diagnosis of PVM to perform a non – destructive and contactless inspection [7]. However, the higher airspeed of UAVs and poor resolution of thermal imaging cameras restrict the inspection system from detecting hotspots at elevated temperatures. Hot spots can indicate the presence of several other faults such as partial shading, solder bond failure, micro-cracks, corrosion, and short circuit. Such fault occurrences can be challenging to be detected on the application of thermograph assessments since the acquired thermal images contain thermal radiation information that is pseudocolored [8]. Owing to the drawbacks mentioned above, thermal cameras in UAVs are replaced with digital cameras of high resolution such that defects of smaller size can be detected. The digital cameras installed in UAVs collect true-color images of PVM which helps detect visual faults like delamination, snail trail, burn marks, glass breakage and discoloration. Several image processing techniques like edge detection [9], aerial triangulation [10], correlated texture feature extraction [11] and image mosaicing [12] have been used in the fault diagnosis of PVM. However, the resolution of the acquired UAV images plays a vital role in the performance of the above techniques. Various external factors like reflection, haze, wind speed and vehicle vibration can hinder the resolution of images acquired by UAVs.

Convolutional neural networks (CNN) are capable of working on low-resolution images and have produced accurate results in certain fault diagnosis applications. CNN forms the basic blocks for deep learning and is considered a powerful tool to extract and classify image features in computer vision. Deep learning has been applied in several fields like automation, health care and robotics due to their superior feature extraction and image classification ability [13]. Adopting deep learning techniques in fault diagnosis has produced more accurate results than conventional methods. The use of CNN in fault diagnosis of PVM has been exhibited in various literature. Deep learning based intelligent fault pattern recognition was performed by [14] to detect and classify five test conditions. Support vector machine (SVM) algorithm was used as a classification tool in the former approach. Metallic surface defects were identified using a novel cascading autoencoder (CASAE) architecture. The defective regions in the images were segmented and localized using the architecture while the defects were categorized into their respective classes with the help of a compact CNN [15]. Akram et al., [16] performed automatic fault detection on electroluminescence images utilizing a light CNN architecture. The model reported an overall classification accuracy of 93.02% while working with the first public repository electroluminescence dataset. The authors in [17] attempted a deep learning based approach to detect and classify faults in a PVM installed in large PV farms. Deep features were extracted from PVM images and further processed for classification of faults.

CNN model architectures can be built from scratch or adapted from pre-trained network models. In general, pre-trained network models

hold an upper hand in classification tasks over models built from scratch. The reason for adopting pre-trained networks is that (i) they are trained upon millions of images, (ii) can be customized according to the requirement of the user, (iii) publicly available and ease of access, and (iv) training weights can be used to load the model. Such advantages have made researchers opt for pre-trained network models. Among all the available pre-trained network models, VGG16 [18] and AlexNet [19] have been widely used for image classification problems. Some works of literature based on the application of pre-trained network models are discussed as follows. Automatic fault detection of damaged PV cells was carried out by authors in [20] using thermal images. A deep convolutional neural network (DCNN) portraying VGG16 architecture was applied to identify and detect damaged PV cells. The authors in [21] investigated the application of AlexNet in the detection of PV array faults. CNN is an efficient feature extraction tool due to its high compatibility and self-learning ability even while working with low resolution images. However, the amount of dataset and training time determines the classification accuracy and performance of deep learning models. Thus, CNN models require larger datasets for training such that the image features are appropriately learned. Acquiring large datasets for a particular application is a challenging task. Data augmentation is one technique that solves the challenge posted for dataset expansion. Generative adversarial network (GAN) based augmentation is applied in a number of research works to artificially expand datasets and eliminate data scarcity [22]. However, GAN-based techniques contain numerous convolutional layers, resulting in higher hardware requirements and training time. Simple and quick data augmentation techniques can be a practical option.

Additionally, a number of machine learning (ML) algorithms have been applied to classify faults in several literatures which are discussed as follows. Thermal images of three fault classes of PVM were classified using the Naïve Bayes algorithm with a mean classification accuracy of 94.1% [23]. SVM algorithm was applied by the authors in [24,25] for classifying fault occurrences in a PVM. k – nearest neighbor (kNN) algorithm was utilized to classify string level faults in a PVM. The model based approach resulted in a classification accuracy of 98.7% [5]. A multichannel CNN was adopted by Ying et al., to detect the defects in a PV cell. Significant features from electroluminescence images were extracted using CNN and classified with the help of a random forest (RF) algorithm [26]. In another study, an intelligent fault diagnosis of PV arrays was carried out using RF based ensemble learning algorithm [27]. Based on the literatures, it is clear that ML algorithms produce accurate classification results and can be applied in a wide range of applications.

Numerous works were reported on the usage of thermal or electroluminescence images for fault classification. However, usage of RGB or true-color images were limited and considered only in a few studies. Thermographic assessments of PVM are limited only for detecting the presence of hot spots only. Detection and classification of faults in a PVM using deep learning and machine learning based techniques were carried out in the literature. However, such techniques usage was limited in the reported literature. Also, the effectiveness and capability of a combined approach using deep learning and machine learning were not explored. The non-availability of public repositories and scarcity of dataset has made data acquisition challenging. GAN-based data augmentation contains numerous convolutional layers leading to higher hardware requirements and training time. Hence, data augmentation can eliminate the challenges posted. From the literatures, it is evident that a confined set of faults were only considered during the detection and classification of faults in PVM. Thus, there is a strong necessity to design an advanced fault diagnosis technique that can detect multiple faults using a combination of deep learning and machine learning techniques. The methodology of the proposed work is presented in Fig. 1. The present study attempts to improvise upon the state of artwork carried out and the essential technical contributions are listed below:

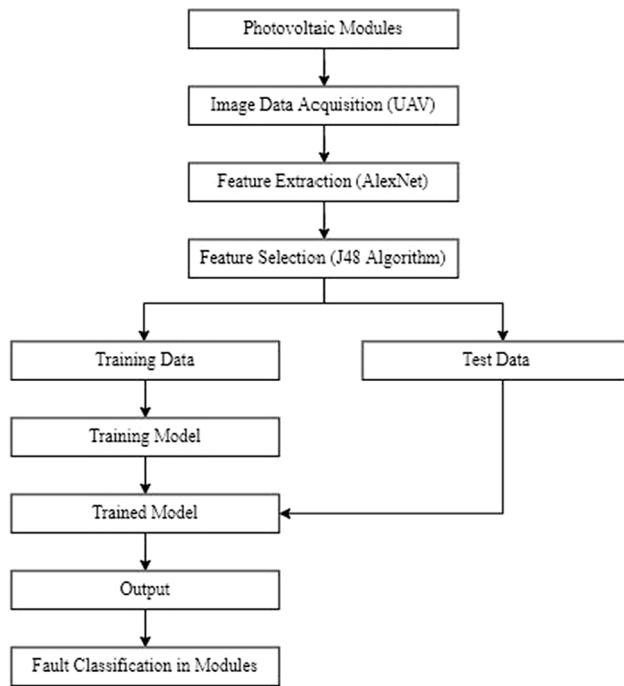


Fig. 1. Proposed Methodology.

1. In the present study, RGB images (acquired from UAVs) were utilized to perform fault detection and diagnosis of PVM.
2. Multiple visual faults like delamination, snail trail, burn marks, glass breakage and discoloration, along with good condition panel, were considered in the study.
3. Data augmentation technique that applies various transformation functions was used to expand the acquired aerial image dataset.
4. A combined approach of deep learning (for feature extraction) and machine learning (for feature selection and classification) was not attempted.
5. Use of lazy classifiers like K-star algorithm (KS), nearest neighbor (NN), locally weighted learning (LWL) and k-nearest neighbor were not reported in literatures for PVM multiple fault classification.

The rest of the paper is outlined as follows. **Section 2** represents the experimental setup, experimental procedure and a brief description of various visual faults in PVM. In **Section 3**, feature extraction using the AlexNet model is explained, followed by feature selection using the J48 algorithm in **Section 4**. Short notes of lazy family of classifiers are described in **Section 5**. The obtained experimental results are reported in **Section 6** followed by conclusion in **Section 7**.

2. Experimental studies

The primary objective of the present work is to discriminate the condition of PVM as good or defective. If the condition of PVM is found defective, then the objective of the proposed approach is to find the type of fault. Based on the proposed methodology presented in **Fig. 1**, the initial two blocks are explained in the following subsection. The study was performed on modules placed over a stationary stand in laboratorial conditions.

2.1. Experimental setup

The experimental setup contains a UAV-based monitoring platform equipped with a high-resolution digital camera, a set of sensors, onboard processors and a ground control station [28]. The general process involved in UAV based monitoring platform depicted in **Fig. 2** is explained as follows. A DJI Mavic 2 Zoom drone (remotely controlled with the help of a hand-held remote controller) equipped with RGB digital camera was used to capture aerial images of PVM under laboratory conditions. The captured images are sent to the ground control station through wireless communication and are stored in a storage device. The collected images are further processed offline with the help of deep learning and machine learning approaches accompanied by the interpretation of classification results. A detailed specification of the UAV used is displayed in **Table 1**. Five faulty and one good condition test module were considered for acquiring images with the help of UAVs. The modules were placed across the laboratory at six different points for the purpose of image acquisition.

The UAV was operated at heights between 1 and 5 m above the modules to acquire images. **Fig. 3** depicts the PVM images acquired using UAV. The acquisition of aerial image data was performed in two sessions, each of approximately 14 min. In every session, the images of the six test conditions (delamination, snail trail, burn marks, glass

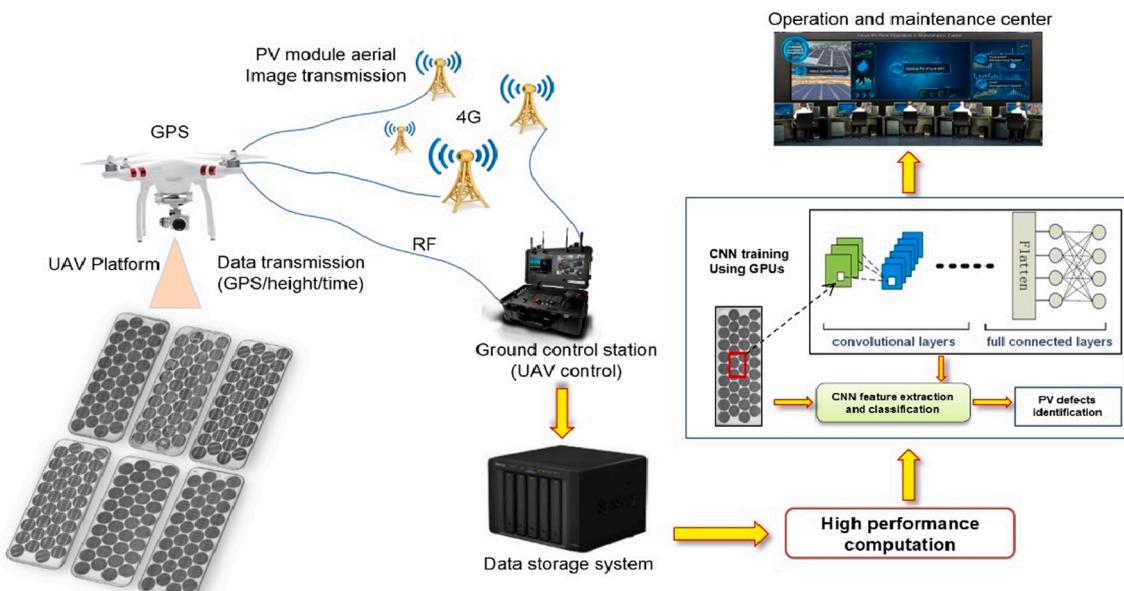


Fig. 2. Overall procedure of UAV monitoring platform.

Table 1

Detailed specification of DJI Mavic 2 Zoom UAV.

Adopted factors	Values
Cruise speed	10–20 m/s
Mission altitude	20–30 m
Operational range	5 km
Weight	1.6 kg
Flight endurance	0.28–0.70 h
Wing span	0.354 m
Length	322 mm
Maximum resolution	4000 × 3000
Propulsion	Electric Power
Image sensor size	6.18 mm × 4.50 mm

breakage, discoloration and good panel) of PVM were acquired using UAV.

For every test condition, 100 images were captured and stored in approximately 2.3 min per module condition. Udhaya Semiconductors Limited manufactured the PVM used in the experimental study. **Table 2** describes the complete specification of the PVM used. The values displayed were measured at standard test conditions *i.e.*, at 25°C temperature, the irradiance of 1000 W/m² with AM1.5 spectrum.

2.2. Experimental procedure

The overall experiment is performed in four stages: (i) dataset acquisition/creation, (ii) CNN-based feature extraction, (iii) feature selection using the J48 algorithm and (iv) feature classification using lazy classifiers. In the present study, UAV was employed to acquire image data of various test conditions of PVM. Six test conditions were considered: delamination, snail trail, burn marks, glass breakage, discoloration, and good panel. **Fig. 4** depicts the overall image data acquisition system. As depicted in **Fig. 4**, the hand-held remote controller is used to control the flight of the UAV and is provided with provisions to temporarily store the acquired PVM images. These images can later be transferred to data storage systems with the help of data cables or memory card. For initial experimentation, a total of 600 images *i.e.*, 100 images per test condition were collected and categorized into their respective classes. However, deep learning-based models may not work efficiently with smaller datasets. Therefore, to upgrade the performance of deep learning models, larger datasets are required to train the model such that the features of high significance are extracted properly.

Dataset acquisition/creation is a challenging task when working with CNN. Data augmentation technique addresses the challenges faced during the creation of datasets. The acquired 600 images are artificially

expanded to 3150 images (uniform dataset containing 525 images for each condition) with the help of data augmentation technique. A set of transformation parameters like shift, rotation, flip, noise, warp, blur, zoom etc., were applied to the images to expand the dataset. The parameters used to transform images are presented in **Table 3**. Post creation of the dataset, the features of the augmented images are extracted using a pre-trained AlexNet and stored as comma separated value (CSV) files. Features of high significance and contribution towards classification are selected from the extracted features with the help of the J48 decision tree (DT) algorithm. Finally, the selected features are used to classify the PVM test conditions into their respective classes using lazy classifiers.

2.3. Visual faults in PVM

Thermal stresses, extreme environmental uncertainties and changing climatic conditions have created a vulnerable situation for the occurrence of PVM faults. Such fault occurrences will affect the performance, reliability and life span of PVM. The most commonly occurring visual faults that have been considered in the present study are briefly discussed in this section and are portrayed in **Fig. 5**.

2.3.1. Burn marks

Localized heating (reverse biased current), ribbon breakage and solder bond failure are a few common symptoms that indicate the occurrence of burn marks in PVM. Ribbon breakage between interconnects and solder bond failure in a PVM is induced by the thermal stresses created by the changing extreme climatic conditions, resulting in component failure. Localized heating prevails due to cell cracks or partial shading conditions (due to bird droppings and cloud movement). Burn marks affect the power output of PVM, resulting in power loss.

Table 2

PVM specifications in detail.

Parameter	Value
Model Name	USP-36
Maximum Power	36 W
Maximum Power Point Voltage (V _{mpp})	17 V
Type	Monocrystalline
Short Circuit Current (I _{sc})	2.25 A
Efficiency	9 – 10%
Weight	3.5 kg
Maximum Power Point Current (I _{mpp})	2.1
Dimensions	1011 × 435 × 36 mm
Open Circuit Voltage (V _{oc})	20.6 V
Number of Cells	36

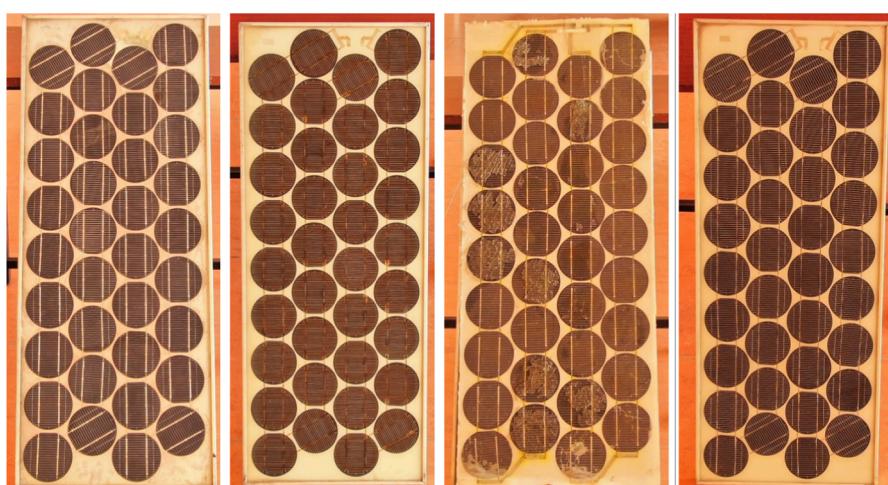
**Fig. 3.** Sample PVM images acquired using UAV.



Fig. 4. Aerial image data acquisition using UAV.

Table 3
Transformation functions applied for augmenting image dataset.

Transform operation	Value
Warp	40
Blur	Gaussian
Noise	Random
Flip (Horizontal, Vertical)	90°
Rotation (Clockwise, Anticlockwise)	0° – 180°

Complete breakage of solder bonds in a PV cell leads to cell blockage that resists the current flow and causes electrical short circuits that create a safety concern [29].

2.3.2. Delamination

PV cells are resisted from the invasion of moisture with the help of an encapsulant layer. Ethylene vinyl acetate (EVA) is the most commonly used encapsulant layer that compactly holds the PV cells. However, failure in adhesive properties between the encapsulant layer and PV cell

will give rise to a phenomenon known as delamination. Heat and moisture accelerate the degradation of the encapsulant material causing a series of problems like water penetration, increased reflection, air bubbles, electrical failure and power loss. In general, delamination occurs predominantly along with the corners and edges of a PVM [30].

2.3.3. Discoloration

The appearance of browning or yellowing in a PVM is considered as an indication of discoloration. The change in physical properties of the type of encapsulant material used (EVA usually) is considered the primary reason for discoloration. The other reasons for the degradation of encapsulant material are exposure to elevated temperatures and ultraviolet (UV) radiation. In the early days discoloration was considered as a failure in PV cells. However, recent studies have revealed that discoloration is a degradation phenomenon. Some studies infer that the rate of discoloration is proportional to the UV radiation intensity. Also, a higher rate of discoloration affects the light transmittance ability of PV cells resulting in depreciated power production [31].

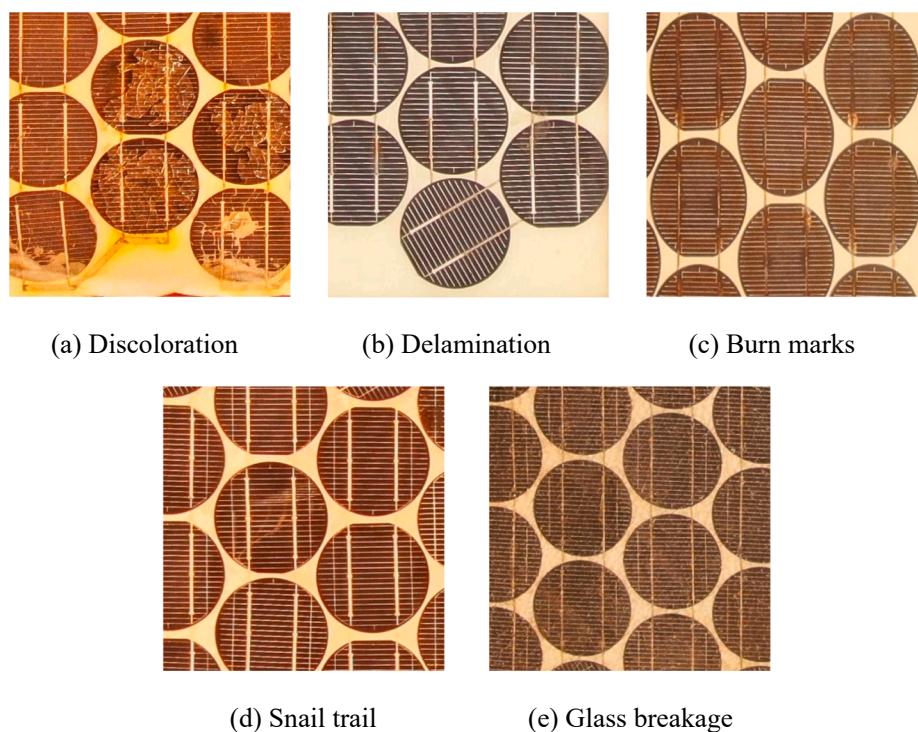


Fig. 5. Visual faults occurring in a PVM.

2.3.4. Glass breakage

Glass breakage in PVM can occur due to the following reasons: (i) sudden impact caused by falling components, (ii) poor packaging of materials, (iii) temperature variations/thermal stresses, (iv) vibration caused during transportation, (v) harsh climatic conditions like lightning and hail storm and (vi) mishandling panels during installation. A tempered glass unit is provided to protect the front glass of the panel. Damage to tempered glass does not affect the performance of the PVM. However, greater impacts can damage the front glass, resulting in moisture penetration, which affects the power output, induces corrosion in interconnects, and raises a safety concern due to short circuiting [32].

2.3.5. Snail trails

Snail trails appearance in a PVM indicates the presence of micro-cracks in cells of PVM. They are commonly observed at the edges of PVM exposed and operated over an extended time. Higher temperature exposure and application of thermal stresses tend to expand the micro cracks present inside a PV cell. Such expansion of micro cracks creates an irregular line-like structure that resembles the trails of a snail. The occurrence of snail trails will result in degraded power loss and accelerated aging of PVM [33].

3. CNN based feature extraction

The present section describes an overview of the convolutional neural networks (CNN) along with the extraction and selection of features carried out in the present study. The feature extraction was performed using AlexNet while the J48 decision tree algorithm was adopted to select the most significant features. In addition, brief descriptions of various lazy classifiers are also provided below.

3.1. An overview of convolutional neural networks

Deep learning is considered a powerful tool that has spread its usage over a wide range of computer vision applications. The fundamental blocks in a deep learning algorithm are constructed using convolutional neural networks. The basic structure of CNN within a deep learning algorithm is non-linear in nature. Higher levels of feature extraction and capability to work with images of poor resolution have made CNN a popular feature extraction tool. The convolutional layer, pooling layer and fully connected layer are the three important layers that constitute the CNN architecture besides numerous hyperparameters and special layers. Convolution layers present in a CNN architecture contain a number of filters and kernels used to learn the optimal feature maps from the input data. Pooling layers stacked along the convolution layer acts as a downsampling layer to reduce the feature dimensions. Fully connected layers perform the classification task on the learned features with the help of the softmax activation function [34]. The convolution and pooling layers continually learn the image patterns and features during the training process. The weights/filters in a CNN model are automatically adjusted to minimize errors with the help of the error backpropagation optimization technique. Generally, the network is trained to absorb specific features from the input data that contribute to the accurate classification. Modeling and training a CNN network from scratch requires a huge number of properly labeled datasets. Such a process is time-consuming and requires more extensive data analysis. However, several studies have demonstrated and recommended the successful use of a pre-trained network model. In general, pre-trained models exhibit superior feature extraction properties since they are trained over many image data. Several CNN architectures like AlexNet, GoogLeNet [35], VGGNet, etc., along with their pre-trained versions, have been made available for public access.

3.2. Feature extraction

Feature extraction process involves a reduction in variables to un-

derstand and describe a relatively larger set of data. In the present study, a pre-trained AlexNet CNN model was employed to extract features from the acquired aerial images of PVM. AlexNet was modeled and introduced by Alex Krizhevsky et al., to classify images in the annual ImageNet Large Scale Visual Recognition Challenge (ILSVRC) competition. The network model outperformed every classical state of art technique and created a trend in computer vision in the form of deep learning. The pre-trained AlexNet was trained upon 1.2 million high resolution images that were composed of 1000 different object classes. Fig. 6 represents the overall feature extraction process and the architecture of pre-trained AlexNet composed of eight deep layers (five convolution layers, two fully connected layers, and one softmax output layer). Image features are extracted by AlexNet and stored as comma separated value files further processed into feature selection and classification tasks. AlexNet accepts images of size 227×227 pixels. Three major operations namely, convolution, max pooling and local response normalization (LRN) are performed in the first convolution layer. A total of 96 different filters each of size 11×11 are present in the layer that resizes the image to 55×55 pixels. The image output is fed into the following layer with 256 receptive filters and max pooling filter size of 3×3 in which the image is again resized to 27×27 pixels. On further passing through the third, fourth and fifth convolution layers the image size is further reduced to 13×13 pixels with rectified linear units (ReLU) as the activation function [36]. The resulting image is passed through two fully connected layers with 4096 output parameters are connected to a softmax output layer for further classification. Solving non-linear problems has made ReLU the most commonly used activation function. Adding to this, the ReLU activation function network required less training time than other activation functions. Overfitting has been a prominent drawback in DL applications. Hence, fully connected layers are connected via a dropout layer such that the problem of overfitting is eradicated. The present study uses a dropout layer in the ratio of 0.5.

An important trait of CNN is that features to discriminate every class are learned automatically from the labels provided in the dataset to facilitate classification. Transfer learning has become an effective way to extract and classify custom image datasets with simple changes in the final few layers. In the present study, the features are extracted from the activated neurons present in the final fully connected layer of AlexNet i.e., fully connected 7. The image features were stored in a CSV file containing 1000 features for every image passed. Machine learning classifiers can be used on the extracted image features to determine the most suitable classifier to detect PVM faults. The architecture details of AlexNet are provided in Table 4.

4. Feature selection using J48 decision tree algorithm

Feature selection is the phenomenon of identifying and selecting the most significant features that could effectively contribute to the desired class prediction. The performance of the classifier can be affected due to the presence of irrelevant data which can significantly increase the computational complexity. Therefore, the feature selection process excludes features of less significance to improve the classification accuracy of the classifier. Decision tree algorithms are widely used in the feature selection process due to their ability to reflect information effectively and standardly. A decision tree resembles a model of a graph that looks like a tree to form the rules of classification. A typical tree consists of leaves, branches, roots and nodes. The classification attributes are represented as nodes connected from root to leaf through branches. The leaves in a decision tree depict the labels of various classes while the classes to be classified are associated with the nodes. The collective decisions that lead to the leaves are denoted and reflected by the branches [37]. The classification of features in a decision tree starts from the root and drops deep across the nodes until a pure leaf is discovered. The most significant and useful features for classification are determined based on the suitable estimation criteria at the decision node. J48 is prominently used for feature selection among the available decision tree

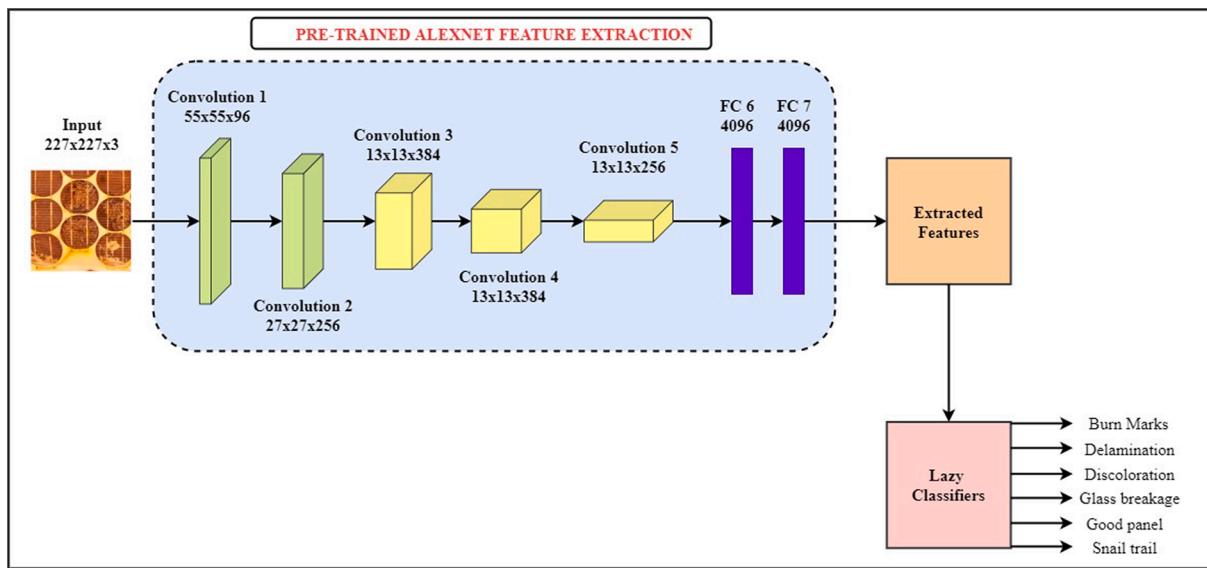


Fig. 6. Overall architecture of pre-trained AlexNet network.

Table 4
The pre-trained AlexNet architecture.

Layer	Convolution 1	Convolution 2	Convolution3	Convolution 4	Convolution 5
Details	96 filters, 11X11, ReLU, Max pooling, LRN	256 filters, 3X3, ReLU, Max pooling, LRN	384 filters, ReLU	384 filters, ReLU	256 filters, ReLU

algorithms. The present study utilizes the J48 decision tree algorithm to select the most significant features obtained from the extracted features of aerial images [38–41].

An experimental test on the extracted features was performed by varying the number of objects (2 to 525) in the J48 decision tree algorithm to identify the corresponding time required to build the model and classification accuracy. The values were recorded and plotted on a 2D graph with the classification accuracy and time taken on the Y-axis against number of objects on the X-axis (Fig. 7). From Fig. 7, it can be observed that the classification accuracy decreases with an increase in the number of objects. However, the time to build a model was high for the minimum number of objects. It can be observed that there is a sudden drop in the time to build the model when the number of objects

was 70. This drop represents the identification of optimum value that corresponds to the classification accuracy of 89.24% which was built in a time of 4.78 s for minimum number of objects 70. Also, the corresponding number of features that contribute to classification was 12. Fig. 8 represents the selected features produced by the J48 decision tree algorithm. The decision tree is represented in the form of a depth-first technique in which the significance of the features reduces from top to bottom. The most significant feature is represented as the top node while the least significant features are represented at the bottom. The list of features that are adequate to classify PVM test conditions are features 582, 972, 359, 963, 150, 51, 927, 770, 961, 394, 918, 685.

5. Feature classification using lazy classifiers

Feature classification is the process performed post selection of features in which the instances are classified into their respective classes. The present study adopts lazy classifiers to perform the classification task. As the name suggests, lazy classifiers accumulate the instances during training and do not carry out any real work until classification. Lazy classifiers identify the training sample with respect to the given test sample based on shorter Euclidean distance and predict the corresponding class as that of the training sample. The classifiers adopted in this study are the K-star algorithm [42], locally weighted learning (LWL), nearest neighbor and k-nearest neighbor (kNN) [43]. The following section briefly describes the lazy classifiers considered in the study.

5.1. Locally weighted learning (LWL)

Locally weighted learning (LWL) uses a parametric free strategy in which neighboring functions accomplish the current prospect with the help of a simple subset of information. The primary idea of LWL is to build models closer to the neighboring functions instead of constructing a comprehensive model for every interest point over an entire functional

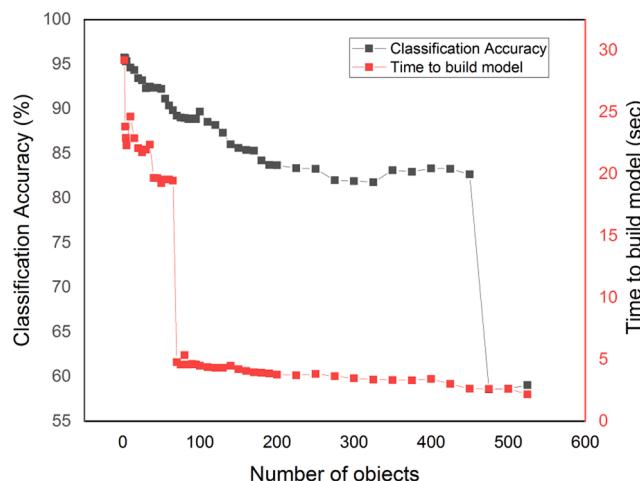


Fig. 7. Feature selection plot with classification accuracy and time to build model (Y-axis) against number of objects (X-axis).

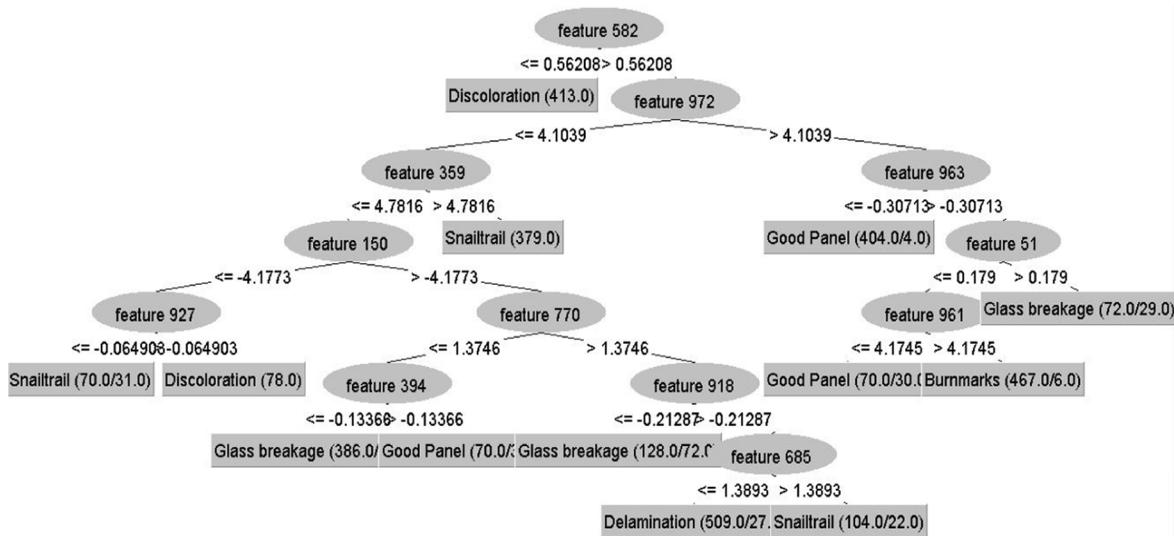


Fig. 8. Features selected by J48 decision tree algorithm (for minimum number of objects – 70).

space. In general, the information present in the closest region of the present query point is assigned with a weight of higher value than the information present in a farther query point. Based on the working principle of LWL, it is termed as a lazy learning classifier as the classifier does not respond until the training data reach a query point. Hence, it is difficult to add new training points in LWL, transforming them into an exact function estimation strategy.

5.2. K-star Algorithm (KS)

Lazy classifiers have been found effective for datasets with limited features. These classifiers adopt a ‘lazy learning strategy’ in which the classifier will not summarize the information in a dataset until a question is raised. The K-star algorithm classifies instances in the test class based on the instances in the training class that are similar in nature. Hence, the KS algorithm is termed an instance based classifier. These classifiers inherit an entropy-based separation function, making them distinct and unique from other instance-based learning algorithms. The calculation of the classifier depends on two variables, missing mode and global bend. Missing mode decides how a missing property value is dealt with within a classifier that employs four modes to treat the missing attributes. The modes applied to treat the missing attributes are:

- Missing attributes condition are ignored (M1)
- Attribute normalization (M2)
- Missing abilities are treated as extremely distinctive (M3)
- Column entropy plots average (M4)

5.3. Nearest neighbor (NN)

Nearest neighbor is an approximation algorithm used to determine an optimum result for a specific problem. The algorithm can store the entire training samples that are capable of classifying every new sample by creating a comparison between the two scenarios. The working of the nearest neighbor algorithm is represented in the following steps:

- Hub selection.
- Out of the selected hub, a round path with the smallest weight value is picked, which connects another hub.
- The hub linked to the selected hub identifies a circular segment with the lowest weight that does not complete a cycle but adds to it anyway.
- Continue the above steps until every part of the hub is linked.

- After completing the links between the hubs, connect the initial and final hub using a spherical section to complete a cycle around the system.

5.4. k-Nearest Neighbor (kNN)

A number of machine learning algorithms are adopted in solving regression and classification problems. Among all the available algorithms, k- Nearest Neighbor is widely accepted in the field of pattern recognition due to the parameter free nature, simple construction, and lazy working strategy. kNN works on the principle of majority voting in which the objects are classified based on the votes provided by the neighboring objects present in the inner parameter space. The object classification performed by the classifier is dependent upon the similarity in features. The parameter free nature of kNN makes the classifier not make assumptions on distributed data. Also, the lazy working strategy of kNN makes the classifier reluctant to learn or adapt any new model and generalize data. In general, kNN classifiers distinguish between various data by forming clusters and calculating the distance among neighboring objects. The ability to work with noisy data, efficient result production and quick implementation on large datasets are certain advantages of using a kNN classifier.

6. Results and discussions

The present study focuses on the detection and classification of faults in PVM with the help of machine learning algorithms. The dataset with various test conditions of PVM images is pre-processed before being fed into the pre-trained AlexNet. The CNN model extracts useful features for all the image classes and stores the extracted features in a CSV file. The most significant and contributive features for fault classification are nominated using the J48 decision tree algorithm from the extracted features. Lazy classifiers like locally weighted learning (LWL), K-star algorithm (KS), nearest neighbor (NN) and k-nearest neighbor (kNN) are adopted to accomplish the classification task on the selected image features. The resulting classification accuracy of all the classifiers is analyzed to determine the dominant classifier. The impact of features on classification accuracy and performance comparison of lazy classifiers are presented in this section.

6.1. Impact of features

A total of 1000 image features were extracted from the final fully

connected layer of the pre-trained AlexNet for every image passed into it. However, the entire set of extracted features might not contribute significantly to the effective PVM test condition classification. Additionally, irrelevant features in classification tasks can demand higher training time and increased computational complexity. Hence, removing irrelevant features during classification tasks will enhance the performance of classifiers. The J48 decision tree algorithm performs the process of irrelevant feature removal and significant feature selection. The J48 decision tree used to select image features for the present study is depicted in Fig. 8. The decision tree ranks the features in descending order of importance starting from the most significant to the least significant feature. The features of less significance and importance are neglected and will be found missing while visualizing the tree. The dimensionality reduction process carried out in the present study is discussed as follows. Initially, the classification accuracy is observed by considering only the root node in the tree. Further, the classification accuracy of the combination of the root node and the next prominent node is observed. The above process is continually performed for all the feature combinations present in the decision tree and the performance of the classifier for all the combinations is noted down. The impact of features combination on the classification accuracy is displayed in Table 5. From Table 5, one can observe that the classification accuracy gradually increases from 53.52% to 84.22% for features ranging from one to four and minor variations can be observed when the number of features is increased beyond four. The maximum classification accuracy of 89.36% was attained when the selected number of features was 19. However, a good accuracy of 87.81% was achieved for 12 features. The reason for selecting 12 features is evident from Fig. 7, as one can observe a steep fall in the time taken to build a model for 70 numbers of objects. Also, selecting 12 features would be an optimal suggestion; increasing the number of features demands high hardware requirements, raises the computational complexity of the model and consumes more time and cost.

6.2. Optimal Hyper-parameter identification for lazy classifiers

Hyper-parameter is a model argument that is provided to a model before the start of a training process. The process of hyper-parameter tuning is described as the selection of optimal values of a function that can change the performance of a specific model. Every model depends on a specific set of hyper-parameters that vary with each model and dataset. Therefore, more accurate results can be obtained using hyper-

parameter tuning. Hyper-parameters can impact the performance of machine learning algorithms to a great extend. Models hyper-parameters are selected based on trial and error experimentation. The model parameters of a particular algorithm are defined based on the input data type while the user tunes the hyper-parameters to achieve the best fit of the model. Parameter tuning can help control the over-fitting and under-fitting scenarios of a particular model. Also, optimal hyper-parameter values can differ based on the type of dataset adopted. The optimal hyper-parameters of a particular model can be identified with the help of the following steps: (i) model evaluation must be performed for every hyper-parameter setting, (ii) selection of hyper-parameters that deliver the best performing model. In the present study, hyper-parameters for every lazy classifier are performed based on the trial and error method for the given PVM dataset. The set of optimal hyper-parameter values for lazy classifiers is represented in Table 6.

For instance, the hyper-parameters of k-Nearest Neighbour were determined by changing every hyper-parameter defined. Initially, the value of kNN was changed continuously from 1 to 525 and the performance of the model was assessed. Based on the models performance it was observed that a value of 1 produced the maximum classification performance. Similarly, the values of distance weighting (No distance weighting, Weight by 1/distance and Weight by 1-distance), search algorithm (Ball tree, Cover tree, Filter Neighbour Search, KD tree and Linear NN search) and distance function (Chebyshev distance, Euclidean distance, Filtered distance, Manhattan distance and Minkowski distance) were varied. Experiments were carried out by changing every hyper-parameter namely, distance weighting, search algorithm and distance function one by one and the results were recorded. The derived results inferred no change in classification accuracy even while changing the hyper-parameters. Hence, the default values were suggested as the optimal values i.e., No distance weighting, Linear NN search and Euclidean distance. The same process was continued for all the lazy classifiers with different hyper-parameters and the optimal values were determined.

6.3. Lazy classifiers performance comparison

The performance comparison of lazy classifiers on the selected features obtained from the J48 algorithm is explained in this section. Delamination, snail trail, burn marks, glass breakage, discoloration and good panel are the six test conditions considered in this study. The aerial image dataset consisting of 3150 images is created using the data augmentation technique. Each class in the dataset contains an equal number of images, accounting for 525 images per class. A pre-trained AlexNet is utilized to extract features from the image and is stored in CSV files. The selected PVM image features acquired using the J48 algorithm are classified using a lazy set of classifiers namely, locally weighted learning (LWL), K-star algorithm (KS), nearest neighbor (NN) and k-nearest neighbor (kNN). The selected twelve features adequate for classifying visual faults in a PVM are features 582, 972, 359, 963, 150, 51, 927, 770, 961, 394, 918 and 685. The performance of the lazy classifiers was evaluated with the help of a tenfold cross validation technique. During a tenfold cross validation, the complete dataset is split

Table 5
Impact of features on classification accuracy.

Number of features	Classification Accuracy (%)
1	53.52
2	69.68
3	77.11
4	84.22
5	84.85
6	85.58
7	86.15
8	86.00
9	85.11
10	86.38
11	86.57
12	87.81
13	87.74
14	87.39
15	87.39
16	89.26
17	87.46
18	87.33
19	89.36
20	88.92
21	87.26
22	88.41

Table 6
Optimal Hyper-parameter values of lazy classifiers.

S. No	Classifiers	Optimal Hyper-parameters
1	Nearest Neighbor (NN)	Do not check capabilities - True
2	k-Nearest Neighbor (kNN)	kNN = 1, Distance Weighting – No, Search Algorithm- Linear NN search, Distance function – Euclidean distance
3	K-Star algorithm	Global bend parameter – 30
4	Locally weighted learning (LWL)	kNN = 0, Classifier – IbK, Search Algorithm – Linear NN search, Distance function - Euclidean distance

into ten equal parts in which nine parts are used for training while one part is kept for testing. Further, the procedure is made to run for ten cycles until every part is used as a test set. **Table 7** depicts the comparison of classification accuracy and time required to build a model for various lazy classifiers. A plot displaying the difference in classification accuracy of various lazy classifiers is presented in **Fig. 9**.

From **Table 7** and **Fig. 9**, one can observe only a slight variation in the classification accuracy between k-nearest neighbor, nearest neighbor and K-star algorithms. However, based on the classification accuracy and time required to build the model, it can be inferred that k-nearest neighbor algorithm produces higher classification accuracy of 98.95% in a shorter time of 0.04 s. The classifiers accuracy of all the lazy classifiers was unaffected even during the application of parameter changes. **Fig. 10** represents a pictorial representation of the correctly classified and misclassified instances in the kNN classifier. The horizontal and vertical axis in **Fig. 10** illustrates the fault classes. The correctly classified instances are in the form of clusters while the misclassified instances are spread across the space. One can observe that burn marks and good panel conditions are misclassified to a large extent.

The derived classification accuracy of the k-nearest algorithm is evident from the confusion matrix presented in **Fig. 11**. The diagonal elements in the confusion matrix represent the correctly classified instances while the regions around the diagonal matrix represent misclassified instances. The row and column labels in the confusion matrix depict the test conditions or the respective classes considered in a particular study. To elaborate more on the confusion matrix, the interpretation of the matrix is discussed as follows. Consider the burn marks condition represented in the first row of the confusion matrix. The first element represents the number of images that are correctly classified and originally belong to the burn mark condition. The other elements in the row with values 1, 2 and 8 portray that the burn mark images are misclassified as delamination, glass breakage and good panel conditions respectively. All the misclassified instances are interpreted similarly with the help of elements that fall around the diagonal matrix. From **Fig. 11**, one can infer that the misclassified instances are smaller in number indicating that most of the image features are learned properly and that only a few features are left out during the training process. Overall, from the total of 3150 images used for classification 3117 images instances are correctly classified and 33 image instances were misclassified using kNN classifier.

6.4. Evaluation of the feature extraction capability of various deep learning models

Image features play an important role in the application of image processing and computer vision. Features are entities that carry useful information about a particular image; typically representing properties of a specific region in an image. Every image contains specific structures namely, edges, points, pixels or objects that are considered as features of an image. Image features can also be a general representation of feature detection or neighborhood operation applied on an image. Feature extraction is the process of extracting useful information from an image that can be utilized to distinguish between various class of objects. In the present study, the feature extraction capability of various pre-trained network models like VGG-16, ResNet-50, VGG-19, GoogLeNet and

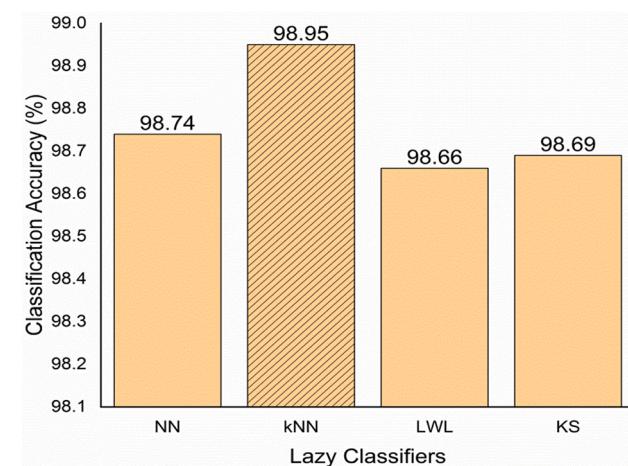


Fig. 9. Performance comparisons of Lazy Classifiers.

DenseNet-201 were compared with AlexNet. The extracted features from the aforementioned pre-trained networks are compared using kNN as a common classifier. The comparison of the feature extraction capability of various pre-trained models are depicted in **Table 8**.

From **Table 8**, one can observe that the performance of AlexNet network with kNN classifier is higher in comparison with other pre-trained networks. The overall classification accuracy of AlexNet was 98.95 % followed by ResNet-50, DenseNet-201, GoogLeNet, VGG-16 and VGG-19 with classification accuracy of 98.79 %, 98.73 %, 98.09 %, 98.00 % and 97.84 % respectively. Based on the aforementioned inference, one can suggest features of AlexNet pre-trained network for PVM fault diagnosis.

6.5. Evaluation of the proposed method with deep learning models

The performance of the proposed fusion methodology of deep learning and machine learning techniques is evaluated in the present section by comparing it with various deep learning pre-trained models. Deep learning algorithms in general are data consumers which extract the image features completely with the aid of convolution neural networks. Pre-trained models are considered as an instantaneous or tailor made solution that is trained already over large datasets comprising of numerous classes. Adopting a pre-trained network to fit into a particular application is known as transfer learning. Model availability is one of the primary requirements in transfer learning such that the model can exhibit good performance for a specific task. The weights achieved during the training process are generally stored and shared in public repositories to transfer learning. VGG-16 [44] and ResNet-50 [45] are two well known and established transfer learning models in the field of computer vision. The aforementioned models among all available pre-trained models have produced efficient results in image classification problems. In the present section, the established pre-trained model performances are compared with the proposed solution. The performance comparison of pre-trained networks and the proposed solution is tabulated in **Table 9**.

The performance of the pre-trained networks was delivered with a softmax classifier. From **Table 9**, one can observe that VGG 16 model exhibited moderate classification accuracy (except good condition) for PVM test conditions. The model accuracy degrades below 70% for PVM conditions like delamination, burn marks and snail trail. On the other hand, the ResNet 50 model delivered better classification accuracy for most PVM conditions. However, poor accuracy was displayed by the model for burn marks condition with a value below 50%. The comparative results in **Table 8** infer that the proposed solution consisting of a fusion of deep learning and machine learning technique produced a higher classification accuracy of 98.95% with a minimum time of 0.04 s.

Table 7
Performance comparison of Lazy Classifiers after hyper-parameter tuning.

S. No	Lazy Classifiers	Classification Accuracy (%)	Time take to build model (seconds)
1	Nearest Neighbor	98.74	0.11
2	k-nearest neighbor	98.95	0.04
3	Locally Weighted Learning	98.66	0.19
4	K-star Algorithm	98.69	0.09

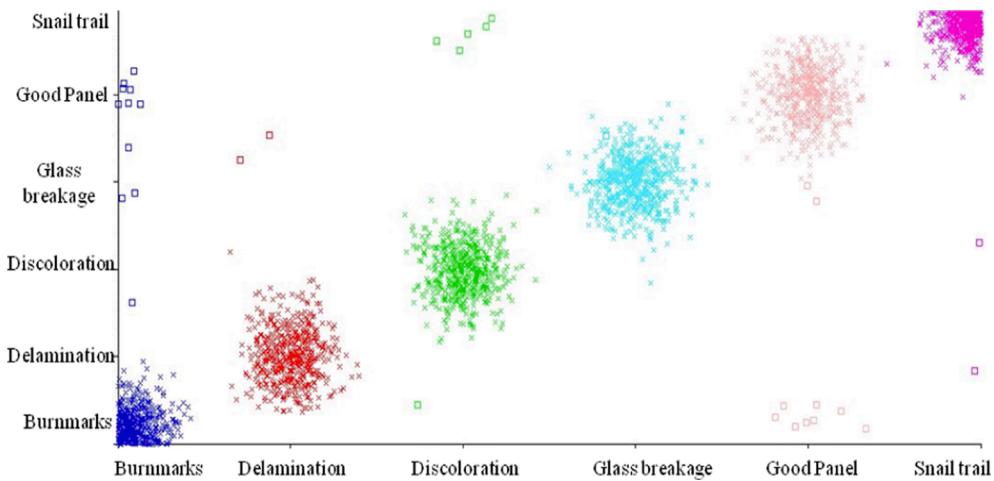


Fig. 10. Correctly classified and misclassified instances for kNN classifier.

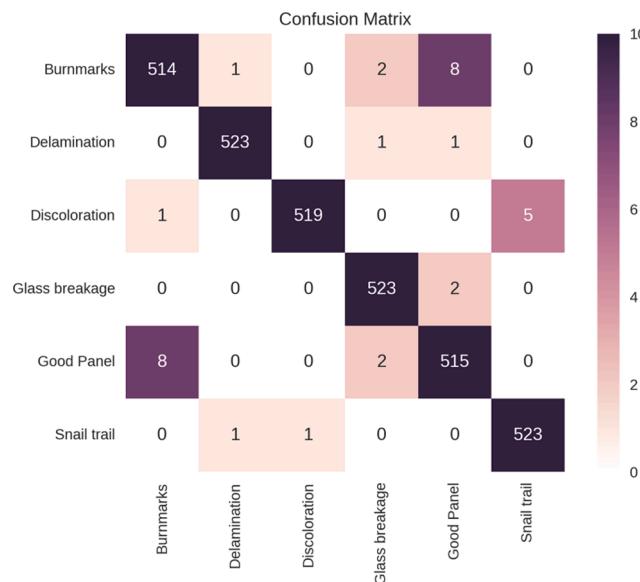


Fig. 11. Confusion matrix of k-nearest neighbor classifier.

The pre-trained models were overfitting and highly complex for the given dataset size. Robust systems with high-end specifications are necessary to run such complex models in limited computational time, thereby elevating the capital cost. The proposed method reduced the computational complexity and time, thereby producing better accuracy and eliminating the necessity for high-end systems.

6.6. Performance comparison with other machine learning classifiers

Numerous algorithms are available in machine learning that can be utilized to perform regression and classification for any given task. In the present study, the classification performance of kNN algorithm is compared with other state of the art machine learning classifiers like Naïve Bayes (NB), Support Vector Machines (SVM), Multilayer Perceptron, Logistic Regression, J48 and Random Forest (RF). Table 10

represents the performance comparison of various machine learning classifiers.

From Table 10, one can infer that kNN performs exceptionally well for PVM fault diagnosis in comparison with other machine learning algorithms. Random forest performs very close to kNN; however, the time to build model and perform classification was higher than that of kNN. Thus, kNN is suggested for real time application in PVM fault diagnosis considering the computational time and higher performance. To avoid the problem of randomness the experiment was trailed for five times. The image features were extracted five times using AlexNet and the classification was performed for all the five trials (AlexNet feature extraction and kNN feature classification) remained constant at 98.95%.

7. Conclusion

The present study demonstrated the use of lazy classifiers to discriminate various PVM test conditions with the help of images acquired from UAV. A pre-trained AlexNet is utilized for extracting features from the acquired aerial images and significant features were determined using the J48 decision tree algorithm. A set of four lazy classifiers namely, locally weighted learning (LWL), K-star algorithm (KS), nearest neighbor (NN) and k-nearest neighbor (kNN) were used to perform the classification task. A tenfold cross validation was opted for verifying the performance of the selected classifiers. Furthermore, the impact of features was studied and the performance comparison of

Table 9
Performance comparison for different PVM conditions.

PVM Condition	Classification Accuracy (in %)		
	Deep learning models		Proposed Solution
	VGG-16 [44]	ResNet-50 [45]	
Good Panel	89.40	89.40	98.09
Delamination	68.60	77.50	99.61
Burn marks	67.60	47.90	97.90
Discoloration	73.50	96.80	98.85
Snail trail	67.90	96.80	99.61
Glass Breakage	76.30	89.70	99.61

Table 8

Comparison of the feature extraction capability of various pre-trained models.

Pre-Trained Networks	VGG-16	ResNet-50	VGG-19	GoogleNet	DenseNet-201	AlexNet
Classification Accuracy (%)	98.00	98.79	97.84	98.09	98.73	98.95

- years of operation in Southern Europe, *Prog. Photovoltaics Res. Appl.* 19 (6) (2011) 658–666, <https://doi.org/10.1002/pip.1083>.
- [31] H. Han, X. Dong, B. Li, H. Yan, P.J. Verlinden, J. Liu, J. Huang, Z. Liang, H. Shen, Degradation analysis of crystalline silicon photovoltaic modules exposed over 30 years in hot-humid climate in China, *Sol. Energy.* 170 (2018) 510–519, <https://doi.org/10.1016/j.solener.2018.05.027>.
- [32] S.S. Chandel, M. Nagaraju Naik, V. Sharma, R. Chandel, Degradation analysis of 28 year field exposed mono-c-Si photovoltaic modules of a direct coupled solar water pumping system in western Himalayan region of India, *Renew. Energy.* 78 (2015) 193–202, <https://doi.org/10.1016/j.renene.2015.01.015>.
- [33] A. Dolara, G.C. Lazarou, S. Leva, G. Manzolini, L. Votta, Snail Trails and Cell Microcrack Impact on PV Module Maximum Power and Energy Production, *IEEE J. Photovoltaics.* 6 (5) (2016) 1269–1277, <https://doi.org/10.1109/JPHOTOV.2016.2576682>.
- [34] K. O’Shea, R. Nash, An Introduction to Convolutional Neural Networks, (2015). <http://arxiv.org/abs/1511.08458>.
- [35] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2015: pp. 1–9. <https://doi.org/10.1109/CVPR.2015.7298594>.
- [36] X. Jiang, Y. Pang, X. Li, J. Pan, Y. Xie, Deep neural networks with Elastic Rectified Linear Units for object recognition, *Neurocomputing.* 275 (2018) 1132–1139, <https://doi.org/10.1016/j.neucom.2017.09.056>.
- [37] A. Joshuva, V. Sugumaran, Comparative study on tree classifiers for application to condition monitoring of wind turbine blade through histogram features using vibration signals: A data-mining approach, *SDHM Struct. Durab. Heal. Monit.* 13 (2019) 399–416, <https://doi.org/10.32604/sdhm.2019.03014>.
- [38] A. Sharma, V. Sugumaran, S. Babu Devasenapati, Misfire detection in an IC engine using vibration signal and decision tree algorithms, *Meas. J. Int. Meas. Confed.* 50 (2014) 370–380, <https://doi.org/10.1016/j.measurement.2014.01.018>.
- [39] V. Muralidharan, S. Ravikumar, H. Kangasabapathy, Condition monitoring of Self aligning carrying idler (SAI) in belt-conveyor system using statistical features and decision tree algorithm, *Meas. J. Int. Meas. Confed.* 58 (2014) 274–279, <https://doi.org/10.1016/j.measurement.2014.08.047>.
- [40] A.D. Patange, R. Jegadeeshwaran, A machine learning approach for vibration-based multipoint tool insert health prediction on vertical machining centre (VMC), *Meas. J. Int. Meas. Confed.* 173 (2021), 108649, <https://doi.org/10.1016/j.measurement.2020.108649>.
- [41] P. Radhakrishnan, K. Ramaiyan, A. Vinayagam, V. Veerasamy, A stacking ensemble classification model for detection and classification of power quality disturbances in PV integrated power network, *Meas. J. Int. Meas. Confed.* 175 (2021) 109025, <https://doi.org/10.1016/j.measurement.2021.109025>.
- [42] S. Ravikumar, H. Kanagasabapathy, V. Muralidharan, Fault diagnosis of self-aligning toughening rollers in belt conveyor system using k-star algorithm, *Meas. J. Int. Meas. Confed.* 133 (2019) 341–349, <https://doi.org/10.1016/j.measurement.2018.10.001>.
- [43] K. Jafarian, M. Mobin, R. Jafari-Marandi, E. Rabiei, Misfire and valve clearance faults detection in the combustion engines based on a multi-sensor vibration signal monitoring, *Meas. J. Int. Meas. Confed.* 128 (2018) 527–536, <https://doi.org/10.1016/j.measurement.2018.04.062>.
- [44] K.i. Kim, H. Hong, G.i. Nam, K. Park, A study of deep CNN-based classification of open and closed eyes using a visible light camera sensor, *Sensors (Switzerland).* 17 (7) (2017) 1534, <https://doi.org/10.3390/s17071534>.
- [45] L. Wen, X. Li, L. Gao, A transfer convolutional neural network for fault diagnosis based on ResNet-50, *Neural Comput. Appl.* 32 (10) (2020) 6111–6124, <https://doi.org/10.1007/s00521-019-04097-w>.