PROGRESS IN PHOTOVOLTAICS WILEY

Photovoltaic defect classification through thermal infrared imaging using a machine learning approach

Christopher Dunderdale[1] | Warren Brettenny[1] | Chantelle Clohessy[1] | E. Ernest van Dyk[2]

[1]Department of Statistics, Nelson Mandela University, Port Elizabeth, South Africa

[2]Department of Physics, Nelson Mandela University, Port Elizabeth, South Africa

Correspondence
Christopher Dunderdale, Nelson Mandela University, Port Elizabeth, South Africa.
Email: s214046117@mandela.ac.za

Abstract

This study examines a deep learning and feature-based approach for the purpose of detecting and classifying defective photovoltaic modules using thermal infrared images in a South African setting. The VGG-16 and MobileNet models are shown to provide good performance for the classification of defects. The scale invariant feature transform (SIFT) descriptor, combined with a random forest classifier, is used to identify defective photovoltaic modules. The implementation of this approach has potential for cost reduction in defect classification over current methods.

KEYWORDS
deep learning, defect classification, defect detection, infrared thermography, machine learning, photovoltaic, random forest, SIFT, support vector machine

1 | INTRODUCTION

The utilisation of nontraditional energy resources is born from the reality that fossil fuels are not a long-term solution to the global energy demand. Depletion of oil and coal reserves are of great concern, with experts predicting that the demand for oil will exceed supply within the next two decades.[1] Solar energy has emerged as a leading renewable energy resource because of, in part, technological improvements and is consequently experiencing a rapid adoption globally.[2]

Energy received from the sun is abundant and practically inexhaustible. While the annual energy consumption worldwide is seemingly large, it is less than 1% of the yearly energy received from the sun.[3] Because of its geographic location, South Africa is fortunate to receive a large amount of solar irradiance per year with an average annual global solar irradiance of 220 W/m^2.[4] This places the country in a prime position to capitalise on this naturally occurring resource. In order to make use of the above average solar irradiance in South Africa, photovoltaic (PV) systems must function optimally. The power output of PV modules can be influenced by numerous factors including shading, manufacturing defects, and PV module defects due to wear and tear from the environment.[5] In instances such as these,

power output can be reduced by up to 50% of the total potential power production per module.[6] It is therefore clear that factors negatively affecting power production should be avoided and that the early detection and classification of defects is crucial to minimise power losses.

This study addresses the growing need to automate the process of classifying PV module defects. In large-scale plants housing millions of operational PV modules, manual inspection of PV modules is simply not viable. Current techniques for fault analysis can be broadly categorised into two groups: techniques that use electrical measurements to detect faults and alternative methods that do not, such as infrared thermography.

Using thermal infrared imaging, this study investigates the use of the scale invariant feature transform (SIFT) feature descriptor, spatial pyramid matching, and deep learning to detect and classify thermal infrared images of PV modules. This is achieved through the use of random forest and support vector machine (SVM) classifiers as well as the VGG-16 and MobileNet deep convolutional neural network (CNN) architectures. Results of the study are promising, as the feature-based classifiers are able to discriminate between defective and nondefective PV modules with up to 91.2% accuracy. Additionally, it has also been shown that the proposed deep learning models

are able to differentiate between various defects at an 89.5% accuracy level.

This study serves as a novel application of these methods for the purpose of defect analysis using infrared thermography in the PV domain. The successful implementation of the techniques discussed have the potential to increase the speed of maintenance as well as provide a significant cost reduction over currently available methods.

2 | RELATED WORK

Since this study incorporates topics from a variety of fields including computer vision, engineering, and statistics, it is necessary to preface the methodology section with a brief discussion of the various aspects related to the study.

2.1 | Infrared thermography

All objects with a temperature above absolute zero emit infrared radiation. Thermal infrared imaging cameras make use of this property to measure the near-infrared electromagnetic radiation in the wavelength range of 8 to 12 μm emitted from an object.[7]

Infrared imaging is generally regarded as a nondestructive methodology for use in testing. It is used in many industrial applications, one of which is examining heat dissipation of PV modules.[8] Once a thermal image is taken, it can be analysed for anomalies, also known as regions of interest or "hot spots." These regions are typically characterised by portions of an object that exhibit temperatures significantly higher than that of the rest of the object. Establishing these hot spots allows for the application of heuristic knowledge to determine the defect that is present. The hot spots observed in PV modules are of interest to this study.

Thermal images of PV systems may be captured from an elevated platform or by making use of unmanned aerial vehicles (UAVs), which fly over a PV system capturing images. When examining thermal images of PV modules, it is important to take the angle of observation into consideration. The PV module front glass reflects a portion of the sunlight it receives, which is recorded as thermal radiation by thermal cameras. This reflection can result in misinterpretations and false hot spots, as shown in Figure 1. FLIR Systems, the world's largest thermal infrared camera producer, recommends that thermal images be taken at an angle of 5° to 60° (where 0° is perpendicular to the PV module) to negate the effect of the glass reflection.[9] When thermal images are correctly captured, this can be used to ascertain the current state of the PV module.

2.2 | PV module defects

The approach of using thermal infrared imagery for defect detection and classification is of primary interest to this study. Several classification schemes for PV module defects, such as that of the International Energy Agency, have been proposed.[2] However, many of these classification schemes are too granular for practical use. This study proposes the use of a classification scheme based on five types of defects that are easily identifiable upon examination of a thermal image, namely, block, patchwork, single, soiling, and string.

Block defects can be seen visually as a vertical third of the PV module exhibiting temperatures significantly higher than that of the rest of the module. Single-cell defects are identifiable as a small rectangular structure exhibiting a thermal signature that is hotter than the rest of the module. Patchwork defects are visually identifiable as an extension of single-cell defects as patchwork defects are given by multiple single-cell defects scattered in a random fashion. Alternatively, string defects occur as multiple single-cell defects aligned along the vertical plane. The string class can be considered as a special case of the patchwork class.

A description of the soiling defect class is not as straightforward as that of other defect classes. Soiling defects can occur because of a number of factors including sediment, such as sand from the environment, and bird excrement. As a direct result of the factors causing the
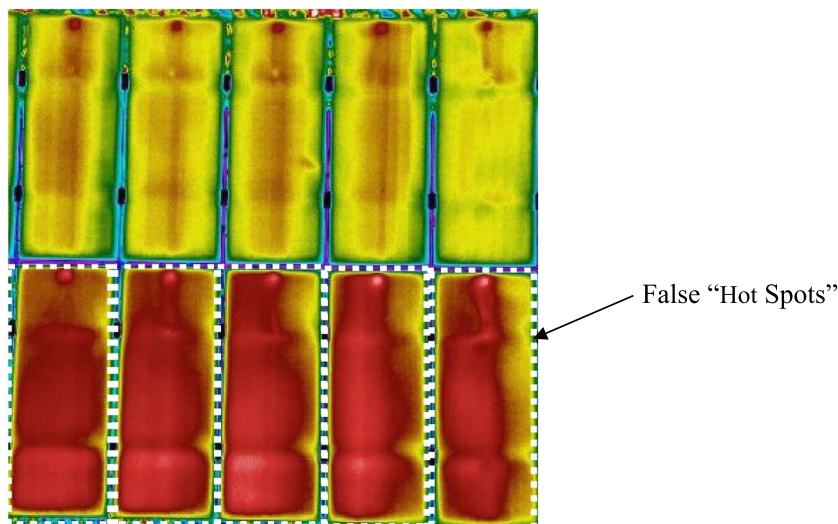
False "Hot Spots"

FIGURE 1 A thermal image of photovoltaic (PV) modules showing false "hot spots" as indicated [Colour figure can be viewed at wileyonlinelibrary.com]

defect, the exact morphology and location of the defect is difficult to ascertain as the shape and placement of these are highly irregular. An example of the proposed schematic with actual images of corresponding defects is given in Table 1.

As previously discussed, PV module defect diagnostics has been explored in various fields. Electrical characterisation has been used in many studies such Baba et al,[10] Chen et al,[11] and Dhimish et al,[12] amongst others, to perform defect diagnostics using electrical parameters such as current-voltage curves, performance ratios, and voltage ratios.

In addition to electrical characterisation, studies by Farress et al,[13] Bartler et al,[14] and Deitsch et al[15] have shown that electroluminescence (EL) imaging is suitable for the purposes of PV module defect detection. This is achieved using techniques such as deep learning.

Work by Jaffery et al[16] showed the viability of using infrared thermography for PV module defect classification. The authors use a fuzzy logic approach to classify the severity of PV module defects. While the approach may show promise, the methodology may not generalise well to variations in PV module technologies.

Pierdicca et al[17] shows that defect detection on PV modules could be achieved though using a deep learning approach. While achieving good classification accuracies, this was limited to dichotomising PV module images into normal and defective classes. Additionally, this study had the benefit of a large sample size.
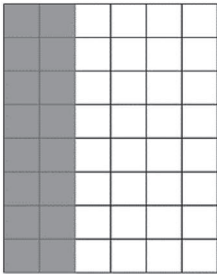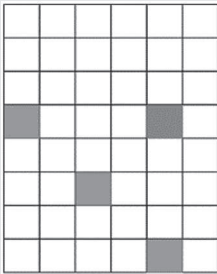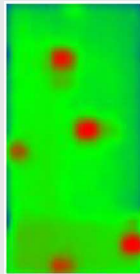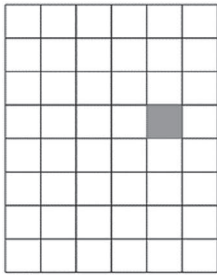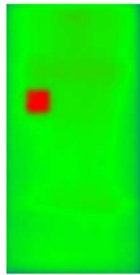
It is important to note that the different approaches for fault analysis do not necessarily need to exist in isolation. One example of how the proposed methods of this study can be incorporated into a processing pipeline is as follows. With the use of remote-controlled UAVs and thermal infrared imaging cameras, plant operators can quickly canvas a solar plant and identify areas that contain potentially defective modules, requiring further analysis. Beyond what is proposed in this study, techniques, such as EL imaging, can be used to establish a more detailed picture of the exact nature of the faults present.

The reason that techniques, such as EL imaging, are not used in every step in this pipeline is due to the time requirements of the technique. An apparent trade-off exists in EL imaging and infrared thermography in terms of measurement speed and detail. The benefit in using infrared thermo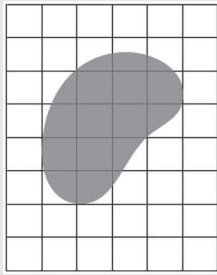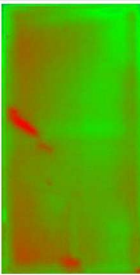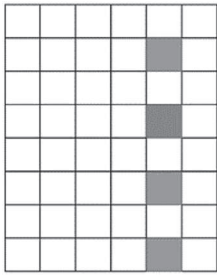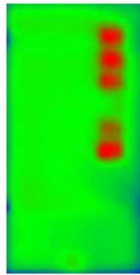graphy is that it can be used to quickly establish problematic areas in solar plants but cannot compete with EL imaging in terms of imaging detail. EL imaging is able to produce high-quality images for analysis but also suffers from practical drawbacks in that it is a time-consuming exercise and needs to be performed in a dark environment or at night.

It is envisioned that the techniques developed by this study can be used to create one complete analytical system in which defects are efficiently detected using infrared thermography and subsequent high-resolution EL imaging can be used to monitor modules that exhibit defects. In this way, the best aspects of each approach can be used to produce an excellent analytical system.

It is evident, when examining the relevant literature, that a large gap exists in approaches that use thermal infrared imaging for defect classification, specifically in classifying defects once it is known that a

defect is present. A methodology that classifies defects in an automated fashion while generalising to different PV module technologies is desirable.

The benefit of an automated analytical system that can differentiate between various defects is that labour and expertise can be

TABLE 1 A comparison of schematic defects with actual defects

Fault	Schematic Example	Practical Example
Block		
Patchwork		
Single		
Soiling		
String		

prioritised to the modules that are defective. Additionally, this prioritisation can also ensure that the defects that have the greatest impact on the power production are addressed with urgency.

2.3 | Image classification

Image classification has made great strides since its inception. Initial attempts at image classification used global features such as histograms of colour and histograms of orientated gradients (HOG).[18] HOG features, in particular, showed that the global features were viable approaches for tasks such as object detection.

Following global feature detection methods, local feature calculation garnered interest, and a significant amount of research went into developing this area. Two aspects in local feature detection became pressure points for research, namely, how to detect the local features (a feature detector) and how to describe these local feature points (a feature descriptor). Descriptors, such as the speeded up robust features (SURF), Oriented fast and Rotated Brief (ORB), and SIFT were subsequently developed to address the issue of feature description. The SIFT descriptor of Lowe,[19] in particular, showed good performances and could be used to both detect and describe local feature points.

In recent times, there has been a shift away from feature-based image classification to deep learning. The use of neural networks, CNNs, and generative adversarial networks have occupied the majority of space within the literature. The new approach is driven by the vast amounts of data being produced by companies such as Google and Facebook. However, these techniques are highly reliant on multitudes of training data to produce adequate results. Many studies, including this one, do not have the luxury of large datasets.

3 | METHODOLOGY

This section details the methodology followed in the detection and classification of defective PV modules. A description of the data, the image feature calculation methods, and the machine learning models used for classification are contained within. The process followed in this study can be broken down into six steps as shown in Figure 2.

The initial step is the data acquisition process, which is followed by the data cleaning and anomaly removals. If applicable, image feature calculations are performed in order to perform analysis. Once the input data has been processed into an appropriate format, machine learning algorithms are applied to the labelled data. The models are also cross-validated to ensure that the reported accuracies are indicative of the model's true performance on unseen data. Finally, the performances of the models are evaluated, and conclusions are drawn to establish which of the best performing models should be used.

3.1 | Data

Thermal image data of 398 singular defective and 400 singular non-defective PV modules from three different PV plants were obtained. Because of nondisclosure agreements, the location of the sites cannot be disclosed.

All three locations under study use crystalline silicon PV modules, which were analysed for defects. A FLIR Tau 2 640 thermal imaging camera was used to capture the images. Once captured, individual modules were cropped out of the thermal images for analysis. An example of this can be seen in Figure 3. The operating conditions in which the thermal infrared images were captured complied with the recommended operating conditions as stipulated in the International Energy Agency[20] report on infrared thermography field applications. In order to ensure the integrity of the training dataset, any potential false hot spots were removed from the dataset.

The defective dataset composition, by class, is given as follows: 32% block, 22% patchwork, 22% single, 21% string, and 3% soiling. Because of a lack of samples and an unacceptably high within-class variation, the soiling class was not used in analysis. This resulted in 383 images being used for analysis.

In order to obtain results from the proposed classifiers that are indicative of their true performance, a 5-fold cross-validation methodology is implemented in this study. This is achieved by randomly separating the data into five folds (of approximately 20% each), where, for each of the five iterations, four folds (80% of the data) are used as to fit the model, and the remaining fold (20% of the data) is used for testing to obtain a classification accuracy. This iterative procedure ensures that every group is used for testing, thereby giving an accurate indication of the model's performance on unseen data.
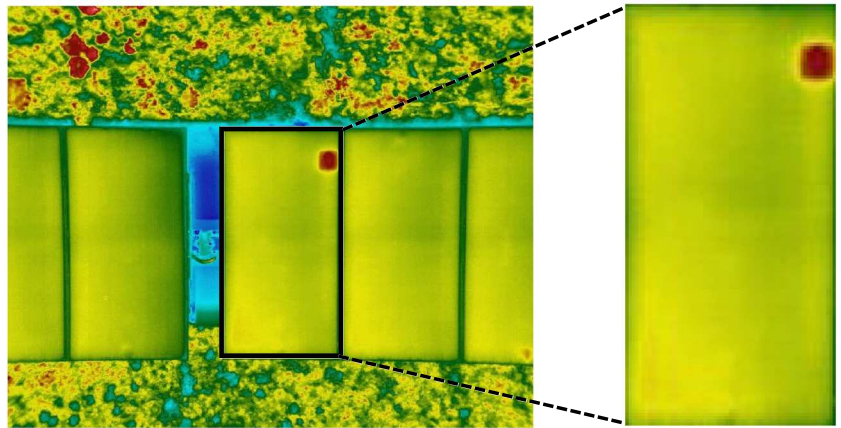
3.2 | SIFT descriptors and the bag of visual words model

The proposed feature descriptor used in this study is the SIFT descriptor. Introduced by Lowe,[19] this feature descriptor has seen mass adoption. In the same period, Fei-Fei and Perona[21] proposed the
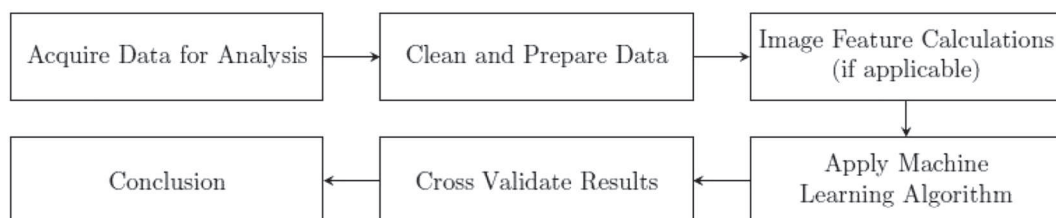
FIGURE 2 Proposed process for defect classification

FIGURE 3 An example of the imaging processing prior to analysis [Colour figure can be viewed at wileyonlinelibrary.com]

quantisation of image descriptors to create a bag of visual words. The following subsections discuss how each method is implemented.

3.2.1 | SIFT descriptors

The SIFT descriptor is an image descriptor with desirable properties such as invariance to scale, rotation, and other forms of image distortion such as affine transformations.[19] Beyond describing images, the methodology also provides a means to identify significant keypoints within an image. An abbreviated process of calculating SIFT descriptors is as follows[19]:

1. Scale space extrema detection: Using methods such as the Laplace of Gaussian, areas of rapid change at various resolutions are detected and labelled as keypoints.
2. Thresholding: For the purposes of increased accuracy, nonsignificant keypoints such as low contrast edges are removed.
3. Orientation assignment: A neighbourhood around each keypoint is calculated to determine the orientation and scale of the keypoint.
4. Keypoint description: A 16-block grid is placed around each keypoint wherein an n-bin orientation histogram is calculated for each sub-block.

By design, each SIFT descriptor returns a 128-length vector for every significant keypoint in an image. A more detailed explanation regarding the exact mathematical processes used to obtain SIFT descriptors is given in Lowe.[19]

An extension of the traditional SIFT descriptor methodology is Dense SIFT. Dense SIFT descriptors are obtained by calculating descriptors for each point on a dense grid instead of sparse keypoint identification as in the traditional framework of Lowe.[19] For low-quality images, such as this study, it is predicted that this approach should capture more information and thus yield better results. The difference between SIFT and Dense SIFT can be seen in Figure 4.

The properties of SIFT descriptors (namely, invariance to rotation and scale) make it a suitable candidate for this study as the positioning of the camera above PV modules is not necessarily consistent. The invariance of the descriptors to rotation and scale are key properties

in this regard. This can also alleviate the need to perfectly crop images, reducing the effect of human error when processing data.

Arguably, the most important parameters for the SIFT algorithm, which need to be optimised, are the following:

1. Sigma: A Gaussian filter is applied to images in the process of obtaining SIFT descriptors. Sigma (σ) is an input parameter used to determine the radius of the filter for image smoothing.
2. Contrast threshold: A value used to filter out low-contrast keypoints detected by the procedure.
3. Edge threshold: This is similar to the contrast threshold but is used to filter out edge-like features that are detected as significant keypoints by the algorithm.

While there may be an initial hurdle in terms of parameter choice, this approach has shown that it can be used in a variety of scenarios and still maintain good performances.

3.2.2 | The bag of visual words model

The bag of visual words model (also known as the bag of words model) takes its inspiration from text sentiment analysis. With image classification in mind, the bag of visual words model creates a succinct feature set that can be used in machine learning algorithms.

The process of implementing the bag of visual words model, as described by Fei-Fei and Perona,[21] can be broken down into three stages:

1. Local feature extraction: All images in the training dataset are scanned for keypoints using an appropriate image feature detection method. These keypoints are then described by an image feature descriptor algorithm. The SIFT methodology provides a means to perform both operations, thus making it a suitable candidate for this purpose.
2. Codebook generation: The generated descriptors from step 1 are quantised using a clustering algorithm such as K-means clustering. The number K is indicative of the size of the codebook or dictionary of visual code words, and the value chosen for K is application specific.
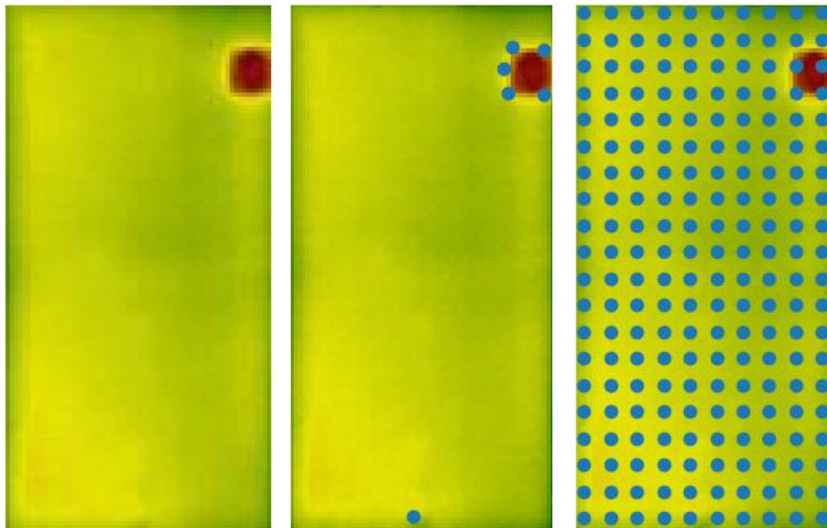
FIGURE 4 From left to right: The original image for reference, scale invariant feature transform (SIFT) applied to image obtaining sparse keypoints, and Dense SIFT applied obtaining a grid of keypoints [Colour figure can be viewed at wileyonlinelibrary.com]

3. Represent each image using code words: Post codebook generation, each image is represented in terms of code words. This is achieved by fitting the descriptors from each image to the K-means model. The resulting feature set for each image is the number of features fitting into each of the K bins of the K-means model.

For the purposes of image classification, once the above steps have been implemented on the training dataset and a model developed for classification, steps 1 and 3 are performed on the validation dataset where the same K-means model, developed by the training set, is used to construct the feature set.

3.3 | Spatial pyramid matching

Many methods, such as the SIFT combined with the bag of visual words model, identify keypoints for feature extraction and produce feature vectors wherein the spatial relationship between keypoints is not captured. This is referred to as an orderless model. Lazebnik et al[22] proposed a methodology wherein features of an image are calculated at varying resolutions, quantised using techniques such as that of the bag of visual words model, and concatenated into a large feature vector. Spatial pyramid matching is referred to as an ordered model since it captures spatial relationships. An example detailing the construction of features using this approach can be seen in Figure 5.[19]

In Figure 5, the image of interest is broken down into several levels. In each of the l levels, an image is subdivided into 2^l grids. At each level, the number of keypoints observed in every grid is recorded, which is used in the analysis.

In order to capture the spatial relationship of keypoints, Lazebnik et al[22] proposes the use of an SVM making use of a custom kernel known as a pyramid matching kernel given by
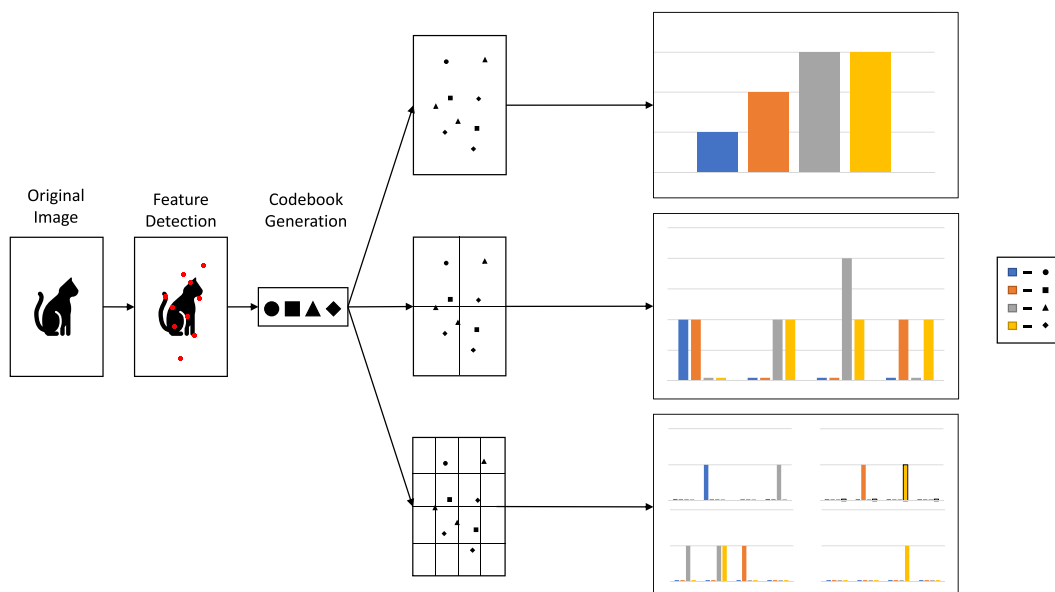
FIGURE 5 A mock example of constructing a three-level pyramid[22]

$$K^L(X, Y) = \frac{1}{2^L} I^0 + \sum_{l=1}^{L} \frac{1}{2^{L-l+1}} I^l,$$

where $I^l = I\left(H_X^l, H_Y^l\right) = \sum_{i=1}^{D} \min\left(H_X^l(i), H_Y^l(i)\right)$ and $H_X^l(i)$ is the number of keypoints falling into grid i at level l.

As suggested by Lazebnik et al,[22] SIFT descriptors are a suitable choice to be used in conjunction with spatial pyramid matching.

3.4 | Deep learning

Given that the general trend in image classification is moving towards the use of artificial neural networks, it is a natural progression that this study should evaluate the feasibility in using these models. This study proposes the using of two deep learning CNN architectures, MobileNet[23] and VGG-16,[24] to create a model that can adequately classify various defects.

VGG-16 is a frequently used model for the purpose of image classification. Its high usage can be attributed to its performance in the 2014 ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in which it achieved a second-place finish. Beyond the architecture's performance, it is easily implementable with many libraries, such as Tensorflow[25] and Keras,[26] supporting these models. The structure of the VGG-16 architecture can be seen in Figure 6.[24]

The MobileNet architecture provides an alternative to very deep CNN architectures, such as VGG-16. The model itself is much more compact than that of the VGG-16 model, requiring only a fraction of the parameters to fine tune. The MobileNet architecture was envisioned to be implemented in mobile applications. The compact design of the MobileNet architecture is due to the depthwise convolutional layer operations that are performed instead of traditional convolutional layer operations. The result of this choice is a more computationally efficient model. The structure of the MobileNet architecture can be seen in Figure 7.[23]

In order to fully harness the power of transfer learning, this study proposes that both the VGG-16 and MobileNet model architectures utilise the pretrained weight matrices of the ImageNet competition, which are readily available on Keras. Additionally, in order to utilise the models as easily as possible, the training images are required to be conformed into a 224×224 shape. The conformation of the training images to the model and not vice versa is so that the predesigned models can be easily implemented without needing to program convolution operations for a rectangular image. Furthermore, the original design of the VGG-16 and MobileNet architectures was for a 1000 class problem. In this study, the final fully connected layers in both models are replaced by a four-neuron output layer so as to construct models that are fit for purpose.

Table 1. MobileNet Body Architecture

| Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5× Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

FIGURE 7 The MobileNet architecture[23]

3.5 | Defect detection and classification

The primary aim of this study is to evaluate methods for the purpose of defect analysis.

It has been noted that no work has been done in establishing the effectiveness in using feature-based descriptors, in conjunction with machine learning and infrared thermography, for the purpose of defect detection. A short study will be conducted in this paper to establish the effectiveness of this approach. The defect detection will be performed by using the SIFT descriptor, in conjunction with the bag of words model, to classify the presence of defects. The random forest model and the SVM will be used for this purpose. The SVM classifier will make use of two kernels, the polynomial (SVM[Poly]) and radial basis function (SVM[Radial]).

For the purpose of defect classification, this study proposes the random forest classifier, when used in conjunction with SIFT descriptors and the bag of visual words model. In addition to random forests, the study will also examine the effectiveness of the SVM classifier as a reference. For spatial pyramid matching, the use of an SVM is needed to implement the custom kernel.

As discussed in Section 3.4, this study will also make use of the VGG-16 and MobileNet deep learning architectures to classify
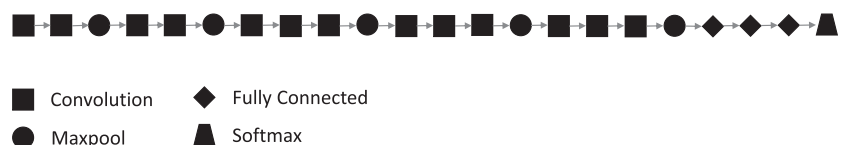
FIGURE 6 The structure of the VGG-16 architecture[24] [Colour figure can be viewed at wileyonlinelibrary.com]

■ Convolution ◆ Fully Connected
● Maxpool ▲ Softmax

defects. What stands as a practical hindrance to the use of the proposed deep learning models is the insufficient dataset size. In an attempt to boost the model performance on this dataset, data augmentation is proposed in order to increase the training dataset size. Two data augmentation techniques are proposed, horizontal and vertical flipping of images and 90°, 180°, and 270° image rotations. The choice of these data augmentation techniques is due to its widespread use in the literature as can be seen in Bartler et al,[14] Deitsch et al,[15] Pierdicca et al,[17] and Chollet et al.[26]

In order to optimise the MobileNet and VGG-16 models, an experimental approach is used to find an optimiser that yields the highest classification accuracy. This study proposes the use of the stochastic gradient descent (SGD) and Adam optimiser to maximise classification accuracy. Like the choice in data augmentation techniques, these optimisers are frequently implemented in the literature as can be seen in Bartler et al,[14] Deitsch et al,[15] and Pierdicca et al.[17]

3.6 | Implementation

The proposed methodology as described above is implemented on a number of platforms. The bag of visual words model is implemented on a Python 3.6.5 64-bit interpreter using the *opencv-contrib-python-headless* package, whereas the spatial pyramid matching methodology is implemented using Github source code provided by Li.[27] Finally, all classification and machine learning algorithms—except for the deep learning models—are implemented in R 3.5.0 using the *random forest* and *caret* packages.[28-30] The spatial pyramid matching model makes use of the dense SIFT descriptors for analysis.

This study uses Python 3.6.5 64-bit software to implement the various CNN models. In particular, a variant of the *tensorflow* package, namely, *tensorflow-gpu*, is used in order to make use of all the hardware available for analysis. The difference between *tensorflow* and *tensorflow-gpu* is that *tensorflow-gpu* has the ability to leverage graphics cards to accelerate processing time. The design of graphics cards is such that operations that can be run in parallel from one another can be computed on graphics cards instead of traditional CPUs. This parallel computation has the effect of drastically reducing model training time. The use of the *tensorflow-gpu* package as opposed to *tensorflow* has no bearing on model performance other than training time. If a graphics card is not available, the model will still produce the same results but take significantly longer when processed

on a CPU. In order to streamline the processes of the *tensorflow* package for model implementation, a number of application programming interfaces (APIs), such as Keras, have been developed. This study makes use of Keras in the analysis. The computational equipment used for experiments is two workstations, A and B. The specifications of the systems are given in Table 2.

4 | RESULTS

The models discussed previously were implemented, and the results are provided in the following subsections. The optimal parameter choices for the defect detection component of the study can be found in Tables 3 and 4. Similarly, the chosen parameters for the defect classification aspect of this study can be found in Tables 5–7. For the various implementations, varying combinations of parameter choices were used. The listed parameter choices were those yielding the highest accuracies for each implementation.

4.1 | Defect detection

The confusion matrix and table of accuracies for the bag of visual words model when using the Lowe[19] implementation of the SIFT descriptor is given in Table 8.

For a two-class classification problem, the model achieves an overall accuracy of 91.2% when using a random forest classifier. The confusion matrix can be interpreted as follows. Of the observed nondefective PV modules, 90% of the observations were correctly classified as nondefective, whereas 10% of the nondefective modules were

TABLE 2 Equipment used for the study

Specifications	Workstation A	Workstation B
Processor	2× Intel Xeon E5-2630	AMD Ryzen Threadripper 1950X
RAM	256GB	128GB
Operating system	64-bit Windows 10 Professional	64-bit Windows 10 Professional
Graphics card	Nvidia Quadro P5000	Nvidia GeForce GTX 1080Ti
Hard drive	1 TB SSD	500GB SSD

TABLE 3 Optimal parameter choices for the SIFT feature extraction algorithm when detecting the presence of defects

Parameter	Bag of Visual Words–SIFT	Reference Parameters from Lowe[19]
Sigma (σ)	0.8	1.6
Contrast threshold	0	0.04
Edge threshold	100	10
Dictionary size (K)	100	N/A

Abbreviation: SIFT, scale invariant feature transform.

TABLE 4 Optimal parameter choices for machine learning models when detecting the presence of defects

SIFT				
Model	Parameter 1	Value	Parameter 2	Value
Random Forest	Number of trees (n)	450	Predictors chosen (m)	10
SVM [Poly]	Degree (d)	2	Cost (C)	0.25
SVM [Radial]	Gamma (γ)	0.450	Cost (C)	1

Abbreviations: SIFT, scale invariant feature transform; SVM, support vector machine.

TABLE 5 Optimal parameter choices for feature extraction algorithms

Parameter	Bag of Visual Words–SIFT	Bag of Visual Words–Dense SIFT	Spatial Pyramid Matching	Reference Parameters from Lowe[19]
Sigma (σ)	0.8	0.8	0.8	1.6
Contrast threshold	0	0	0	0.04
Edge threshold	200	200	200	10
Dictionary size (K)	300	100	200	N/A

Abbreviation: SIFT, scale invariant feature transform.

TABLE 6 Optimal parameter choices for machine learning models

Model	Parameter 1	Value	Parameter 2	Value
SIFT				
Random Forest	Number of trees (n)	500	Predictors chosen (m)	18
SVM[Poly]	Degree (d)	3	Cost (C)	0.5
SVM[Radial]	Gamma (γ)	0.033	Cost (C)	1
Dense SIFT				
Random Forest	Number of trees (n)	400	Predictors chosen (m)	10
SVM[Poly]	Degree (d)	3	Cost (C)	1
SVM[Radial]	Gamma (γ)	0.025	Cost (C)	1
Spatial pyramid matching				
SVM[Custom Kernel]	Cost (C)	1.5	N/A	N/A

Abbreviations: SIFT, scale invariant feature transform; SVM, support vector machine.

TABLE 7 Optimal parameter choices for the VGG-16 and MobileNet implementations

VGG-16	
Optimiser	Stochastic gradient descent
Epochs	1000
Learning rate	10^{-5}
Decay	10^{-4}
Momentum	0.9
MobileNet	
Optimiser	Adam
Epochs	500
Learning rate	10^{-2}
Decay	10^{-4}
Alpha	1
Depth multiplier	1
Dropout	10^{-3}

classified as defective. Similarly, only 8% of the observed defective modules were misclassified as being nondefective when they were in fact defective.

The accuracy obtained using the SIFT descriptor is an improvement on the current best results obtained by Pierdicca et al (2018). A plausible explanation for this improvement can be garnered from the manner in which defects present themselves in thermal infrared images. PV modules that are nondefective typically do not display any thermal anomalies and appear as a blank uninterrupted image. Defective modules, however, are the opposite of this in that the presence of thermal anomalies creates the opportunity for more significant keypoints to be detected by the SIFT algorithm.

4.2 | Defect classification

4.2.1 | The bag of visual words model

The confusion matrix and table of accuracies for the bag of visual words model when using the Lowe[19] implementation of the SIFT descriptor is given in Table 9.

As can be seen in Table 9, although the model achieves an average overall accuracy of 73.9% when using an SVM with a polynomial kernel, it struggles to differentiate between the string and patchwork class. This is due to the fact that the bag of visual words model is an orderless model and, as stated previously, no spatial relationships are captured when local features are detected.

The corresponding confusion matrix and table of accuracies for the bag of visual words model, when using the Li (2016) implementation of the Dense SIFT descriptor, is given in Table 10.

An improvement in overall accuracy is seen when using the Dense SIFT descriptor approach. However, the model also struggles to differentiate between the string and patchwork class.

4.2.2 | Spatial pyramid matching

In the previous section, the SIFT descriptor approach had difficulties differentiating between patchwork and string. This is due to the fact that patchwork and string defects are only different by their placement of hotspots. In order to try capture the spatial relationship between hotspots, spatial pyramid matching was implemented. The confusion matrix when using spatial pyramid matching is given in Table 11.

An overall average accuracy of 77.1% shows that the spatial pyramid matching performs on par with the Dense SIFT approach. When examining the confusion matrix, there is no clear performance difference between the two methods. While the prediction accuracy of the string class has improved, the accuracy of the single and patchwork class has diminished.

4.2.3 | Deep learning

As an alternative to the feature-based approaches of SIFT and spatial pyramid matching, the VGG-16 and MobileNet architecture

TABLE 8 Confusion matrix and accuracy table for the bag of visual words model using the SIFT algorithm for the detection of defects

	Predicted			
Observed		**Nondefective**	**Defective**	
	Nondefective	90%	10%	
	Defective	8%	92%	
Algorithm	**Correctly classified images**	**Average accuracy (%)**	**Standard deviation (%)**	**Best accuracy (%)**
Random Forest	146/160	91.2	3.6	93.7
SVM[Poly]	145/160	90.7	4.3	94.9
SVM[Radial]	143/160	89.6	3.5	92.4

Abbreviations: SIFT, scale invariant feature transform; SVM, support vector machine.

TABLE 9 Confusion matrix and accuracy table for the bag of visual words model using the SIFT algorithm

	Predicted			
Observed		**Block**	**Patchwork**	**Single**	**String**
	Block	89%	11%	0%	0%
	Patchwork	0%	64%	6%	30%
	Single	5%	6%	89%	0%
	String	0%	31%	14%	54%
Algorithm	**Correctly classified images**	**Average accuracy (%)**	**Standard deviation (%)**	**Best accuracy (%)**
Random Forest	54/76	71.0	2.7	75.0
SVM[Poly]	56/76	73.9	6.3	78.9
SVM[Radial]	54/76	71.3	5.0	76.3

Abbreviations: SIFT, scale invariant feature transform; SVM, support vector machine.

was implemented. The results of the VGG-16 model with various

TABLE 10 Confusion matrix and accuracy table for the bag of visual words model using the Dense SIFT algorithm

	Predicted			
Observed		**Block**	**Patchwork**	**Single**	**String**
	Block	95%	0%	0%	5%
	Patchwork	0%	60%	7%	33%
	Single	7%	0%	90%	3%
	String	5%	25%	5%	65%
Algorithm	**Correctly classified images**	**Average accuracy (%)**	**Standard deviation (%)**	**Best accuracy (%)**
Random Forest	59/76	77.2	6.8	82.9
SVM[Poly]	59/76	77.0	5.2	81.6
SVM[Radial]	56/76	73.1	5.5	80.4

Abbreviations: SIFT, scale invariant feature transform; SVM, support vector machine.

optimiser and data augmentation approaches are given in Table 12.

As can be seen in Table 12, the highest accuracy achieved through the use of the VGG-16 model was 85.8%. This was achieved through the use of the SGD optimiser using a combination of both data augmentation techniques to boost the training dataset. Table 12 also highlights the importance of varied experimentation in optimisers.

Regardless of the values used, the Adam optimiser was never able to come close to the performance of the SGD optimiser for the VGG-16 model.

The MobileNet results can be seen in Table 13. In this table, the MobileNet CNN architecture outperforms the VGG-16 model as it was able to correctly classify 89.5% of the module defects using the Adam optimiser. This was the best classification performance that was achieved in this study.

TABLE 11 Confusion matrix and accuracy table for the spatial pyramid matching model

| | Predicted | | | |
Observed	Block	Patchwork	Single	String
Block	93%	0%	0%	7%
Patchwork	21%	52%	21%	6%
Single	`0%	23%	77%	0%
String	0%	16%	10%	74%
Algorithm	Correctly classified images	Average accuracy	Standard deviation	Best accuracy
SVM[Custom Kernel]	59/76	77.1%	7.9%	82.0%.

Abbreviation: SVM, support vector machine.

TABLE 12 Accuracy table for the VGG-16 model

	Adam, %	SGD, %
No augmentation	26.6	84.0
Horizontal and vertical flipping	28.7	85.0
Rotation	26.6	83.2
Horizontal and vertical flipping and rotation	27.4	85.8

Abbreviation: SGD, stochastic gradient descent.

TABLE 13 Accuracy table for the MobileNet model

	Adam, %	SGD, %
No augmentation	76.8	80.1
Horizontal and vertical flipping	89.5	85.5
Rotation	83.4	81.8
Horizontal and vertical flipping and rotation	88.2	86.1

Abbreviation: SGD, stochastic gradient descent.

It is clear by examining the results that the approach of using deep learning models yields better results than that of the feature-based approaches when performing defect classification. While the performances are better on average, what is not evident in the Tables 12 and 13 is the standard deviation of the accuracies. Much like the feature-based approaches, the analysis found that the classification accuracies of these models were equally erratic compared with the feature-based approaches with an average standard deviation in the model accuracies of 5.2%. It can be concluded that these models are suitable for their intended application.

4.3 | Computational requirements

Once the analysis has been completed, it is necessary to evaluate if the additional performance gains are worth the computational resources required to train the model. The training and testing time requirements for each model are given in Table 14.

As can be seen in Table 14, the bag of visual words model is not a computationally complex model to run. Most of the analysis could be run on consumer grade hardware without any major processing

TABLE 14 Training and testing time requirements for the implemented models

	Training Time (minutes)	Testing Time (minutes)
Model (based on the bag of visual words model)		
Random Forest	30	2
SVM[Radial]	36	4
SVM [Custom Kernel]	36	4
SVM[Poly]	45	4
Model (based on deep learning)		
MobileNet	110	<1
VGG-16	140	<1

Abbreviation: SVM, support vector machine.

bottlenecks. However, the same cannot be said regarding the training of the CNN models. Without the use of high specification graphical processing units, the required training time would become extremely long. The *tensorflow-gpu* library supports parallel processing on GPUs and the workstations parallelised all potential computations onto the graphics card.

Even with parallelised computations, the training time of the CNN models is extremely long. For a vast number of consumers, such high specification hardware is far too expensive. The feature-based approaches provide a faster method for model training and can be used by many consumers who might only have access to a typically consumer grade computer. With the cost of hardware and processing time in mind, users will need to compare and prioritise which is more important: training time or hardware costs.

5 | CONCLUSION

This study has demonstrated the feasibility in using thermal infrared imagery for defect detection and classification of PV modules. Despite the small dataset, the feature-based methodology was able to correctly discriminate between defective and nondefective modules with up to 91.2% accuracy. Additionally, the proposed deep learning models were able to differentiate between defects with up to 89.5% accuracy. These results are expected to improve when a larger dataset is used.

The proposed approach for defect analysis can be used to assist plant operators to quickly identify as well as rectify defects with minimal effort and cost. In addition, the need for manual inspection will be reduced, further reducing the overall costs of PV module maintenance, making solar energy an even more viable and attractive form of energy production. This study adds to the body of knowledge in PV defect analysis as this approach has not been applied in this field before.

ACKNOWLEDGEMENTS

ORCID

Christopher Dunderdale https://orcid.org/0000-0003-4276-1898

Warren Brettenny https://orcid.org/0000-0003-3294-8827

Chantelle Clohessy https://orcid.org/0000-0002-4612-2228

E. Ernest van Dyk https://orcid.org/0000-0001-7596-0223

REFERENCES

1. Morris C. *Energy Switch: Proven Solutions for a Renewable Future*. New Society Publishers; 2006.
2. International Energy Agency, "Annual Reports," 2017. [Online]. Available: http://www.iea-pvps.org/index.php?id=6.
3. Julian Chen C. *Physics of Solar Energy*. John Wiley & Sons, Inc; 2011.
4. Department of Energy, "Renewable energy: Solar-energy.," 2010. [Online]. Available: http://www.energy.gov.za/files/esources/renewables/r_solar.html.
5. Madeti S, Singh S. A comprehensive study on different types of faults and detection techniques for solar photovoltaic system. *Solar Energy*. 2017;158:161-185.
6. Tsanakas JA, Ha L, Buerhop C. Faults and infrared thermographic diagnosis in operating c-Si photovoltaic modules: a review of research and future challenges. *Renew Sustain Energy Rev*. 2016;62:695-709.
7. Minkina W, Dudzik S. *Infrared Thermography: Errors and Uncertainties*. Wiley; 2009.
8. Buerhop C. and Scheuerpflug H., "Field inspection of PV-modules using aerial, drone-mounted thermography," The 29th European Photovoltaic Solar Energy Conference and Exhibition (EUPVSEC), pp. 2975-2979, 2014.
9. FLIR Systems, "Thermal imaging cameras: a fast and reliable tool for testing solar panels," 2017. [Online]. Available: https://www.flir.eu/browse/professional-tools/thermography-cameras/.
10. Baba M., Shimakage T. and Takeuchi N., "Examination of fault detection technique in PV systems," 35th International Telecommunications Energy Conference, 2013.
11. Chen Z, Wu L, Cheng S, Lin P, Wu T, Lin W. Intelligent fault diagnosis of photovoltaic arrays based on optimized kernel extreme learning machine and I-V characteristics. *Appl Energy*. 2017;204:912-931.
12. Dhimish M, Holmes V, Mehrdadi B, Dales M, Mather P. Photovoltaic fault detection algorithm based on theoretical curves modelling and fuzzy classification system. *Energy*. 2017;140:276-290.
13. Farress F., El Hassani El Alaoui A., Saidi M. N., Tamtaoui A., Naimi Z. and Bennouna A., Defect detection in solar cells using electroluminescence imaging and image processing algorithms, EUPVSEC Conference, 2017.
14. Bartler A., Mauch L., Yang B., Reuter M. and Stoicescu L., "Automated detection of solar cell defects with deep learning," in 26th European Signal Processing Conference (EUSIPCO), 2018.
15. Deitsch S, Christlein V, Berger S, et al. Automatic classification of defective photovoltaic module cells in electroluminescence images. *Solar Energy*. 2019;185:455-468.
16. Jaffery ZA, Dubey AK, Haque IA. Scheme for predictive fault diagnosis in photovoltaic modules using thermal imaging. *Infrared Phys Technol*. 2017;83:182-187.
17. Pierdicca R, Malinverni ES, Piccinini F, Paolanti M, Felicetti A, Zingaretti P. Deep convolutional neural network for automatic detection of damaged photovoltaic cells. *Int Arch Photogramm Remote Sens Spat Inf Sci*. 2018;42:893-900.
18. Dalal N. and Triggs B., "Histograms of oriented gradients for human detection," in Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, 2005.
19. Lowe DG. Distinctive image features from scale-invariant keypoints. *Int J Comput Vis*. 2004;60(2):91-110.
20. International Energy Agency, "Review on infrared and electroluminescence imaging for PV field applications," 2018. [Online]. Available: http://www.iea-pvps.org/index.php?id=480.
21. Fei-Fei L. and Perona P., "A Bayesian hierarchical model for learning natural scene categories," in Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, 2005.
22. Lazebnik S., Schmid C. and Ponce J., "Beyond bags of features: spatial pyramid matching for recognizing natural scene categories," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006.
23. Howard A, Zhu M, Kalenichenko D, et al. MobileNets: efficient convolutional neural networks for mobile vision. *Commun Res Rep*. 2017.
24. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. *Commun Res Rep*. 2014.
25. Abadi M. et at., "TensorFlow: large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: https://www.tensorflow.org.
26. Chollet F. et al., "Keras," 2015. [Online]. Available: https://keras.io.
27. Li P., "Image classification using bag of words and spatial pyramid BoW," 16 June 2016. [Online]. Available: https://github.com/lipiji/PG_BOW_DEMO.
28. R Core Team. *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing; 2018.
29. Liaw A, Wiener M. Classification and regression by randomForest. *R News*. 2002;2(3):18-22.
30. Kuhn M. Building predictive models in R using the caret package. *J Stat Softw*. 2008;28(5).