Photovoltaic Modules Diagnosis Using Computer Vision

Triveni Deepak Dhamale, Rohit Nakashe, Haritha Selvakumaran, Timish Dhage

Pimpri Chinchwad College of Engineering and Research, Pune

Corresponding author: Triveni Dhamale, Email: triveni.dhamale@pccoer.in

As environmental awareness has been culminating for the past few decades, people have been switching towards renewable sources of energy such as solar energy, wind energy, and hydropower. Among these, solar energy seems to be largely used due to the availability of solar energy in abundance, which in turn, has led to a mass increase in the number of solar power plants. Nevertheless, these solar panels are left unattended due to difficulty in the maintenance of solar power plants in large areas and the manual work required. It is important to find and replace these defective panels in time before any severe event occurs. Detecting hotspots, cracking and various other malfunctions in the photovoltaic cell can lead to an increase in the life of the solar panels by 5-10 years. In this paper, we propose a compact intelligent photovoltaic module diagnosis system based on a Deep Learning model that uses multilayer perceptron (MLP) to accurately identify and classify the faults in the photovoltaic module. We have achieved this by tweaking the VGG16 convolutional network architecture to detect 3 kinds of malfunctions – Hotspots, Cracking, Diode, and No-Anomaly for panels that are in good condition by using 2310 images for training, 665 for validation, and, 665 for testing. By using accuracy-score from sklearn.metrics, the model's accuracy was estimated to be above 90 percent for classifying photovoltaic modules.

Keywords: Solar Energy, Convolutional Neural Network, Fault Classification, Transfer Learning.

2023. In Saroj Hiranwal & Garima Mathur (eds.), *Artificial Intelligence and Communication Technologies*, 953–961. Computing & Intelligent Systems, SCRS, India. https://doi.org/10.52458/978-81-955020-5-9-91

1 Introduction

Solar photovoltaic energy is becoming increasingly popular worldwide. In addition to addressing climate change, the reason behind this is also to generate new economic opportunities and most importantly to provide energy access to countless people still deprived of modern energy facilities. The global clamor for harnessing the sun's energy has increased rapidly over the past decade, as highlighted in the IRENA 2021 India Status Report, which asserted that solar PV capacity increased from 65 Megawatts in 2010 to 38 Gigawatts in 2020. In 2020 alone, 4 gigawatts were added. There have been numerous solar PV installations across India, including in shopping malls, residences, and commercial establishments. The concern now is that, after these solar modules have been installed, less attention is paid to evaluating whether they are still operating at their best after five or ten years. The ideal solar PV module should last for 20 to 25 years.

Due to a variety of factors, this lifespan is reduced, and without proper monitoring, one would not know if a solar PV module has already ceased to operate [6]. Typically, these damages begin as hotspots. A hotspot is a defect found in a solar photovoltaic module that adversely affects its performance. Hotspots can occur for a variety of reasons. Shading is one of the most common causes.

Any object that physically blocks the solar PV module can produce shade, such as a branch of a tree, an antenna, a nearby building, or dirt. When one cell is shaded, the current flowing through the other cells is reduced. The good cells that produce higher voltages usually reverse bias the bad ones. Poor cells dissipate a lot of energy when this happens. Overheating or hotspots are caused by the large dissipation of power in small areas. This can result in ongoing yield reductions, and hotspots can potentially catch fire in the worst-case scenario. Another reason for a hotspot could be a flaw in the module, such as a fracture or a dent. This defect becomes a load, so the current additionally concentrates on the current space, inflicting a huge dissipation of power within the broken cell, hence, making a hotspot [7]. As a result, the output power decreases, thereby decreasing the potency of the PV module [5]. For the purposes of assessing the performance of solar PV modules and obtaining parameters such as maximum power output and conversion efficiency, current-voltage (I-V) characterization is used. It is, however, not feasible to do this in large-scale PV systems because each module needs to be disconnected from the array and attached to the I-V curve tracers [6]. This is time-consuming and inconvenient if large solar PV power plants are to be considered. Taking into account these points, an imaging system that can be used to determine the operating conditions of a solar PV module is presented in this study. Therefore, these are the objectives of this project: Develop a non-invasive technique to locate hotspots in the solar PV module using a hotspot detection algorithm, this measure can be used to detect different kinds of malfunctions, and it is non-destructive, contactless, and efficient. There are three major types of malfunctioning PV modules, i.e., hot spots, potential-induced degradation (PID), and open circuits [4]:

a) **Cracking**: When microcracks form in a solar panel, the affected solar cells will have trouble conducting electric currents, which lead to poor energy production and hot spots.
b) **PID**: PID (Potential Induced Degradation) is a condition that may occur a few years after installation. It can be caused by humidity, heat, or voltage. The temperature of the malfunctioned cell is also higher than others and results in a larger and extremely hot area.
c) **Hot spot**: A hot spot is the most common PV module defect. Hot spot results in a higher temperature and may be caused by many reasons, including short circuits, overhead objects, surface fouling, cell material defects, cell cracks, broken glass, and so on.

2 Analysis

Table 1. Comparing proposed model accuracy with state-of-the-art algorithms

Serial No.	Algorithm used	Accuracy
1	K-means Clustering	85
2	K-means Clustering using DBSCAN	87
3	YOLO-PV[20]	91
4	AlexNet	90
5	K-nearest neighbors	87
6	Region-Based Convolutional Neural Networks and supervised learning	73
7	YOLOv3	70
8	VGG16	92

3 Methodology

1) **Dataset:** The proposed model is trained on a labeled dataset, InfraredSolarModules, that contains real-world imagery of different anomalies found in solar power plants. This [4] dataset contains 11 different classes of anomaly and the remaining class is non-anomaly, which is a nominal solar panel, out of which we have chosen 4 of those categories – Cracking, Diodes, Hotspots, and No-Anomaly to train the model (see Fig. 2). Out of these 2310 images have been chosen for constituting each class for training, 665 images have been used for validation, and 665 images for testing the model which has been represented in fig. 1. Every infrared image has a pixel value of 24 by 40 each.

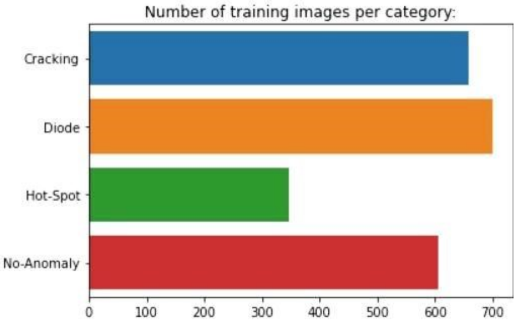


Fig 1. The number of training images used per category using the flow from directory function

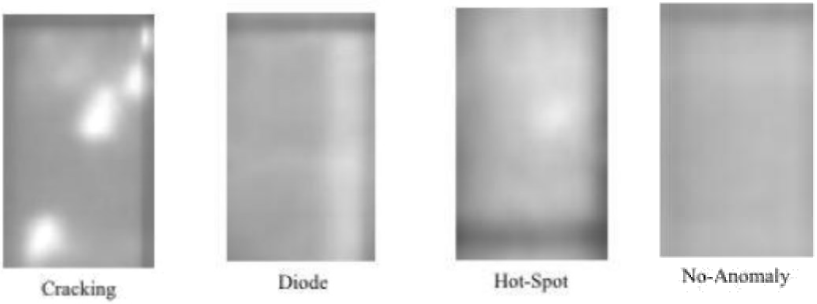


Fig 2. Canonical examples of solar module anomalies observable in infrared imagery.[4]

2) **Image Preprocessing:** The images are already in grayscale, therefore conversion of RGB to grayscale was not required. But the images were put all together without any distinguishing labels for which another program had to be written to create an empty folder for each type of malfunction, fetch images from the main dataset, and classify those images in their respective folders based on information in the JSON file. The batch size being used is 32, and the number of epochs used is 30, which was decided based on the validation dataset accuracy. The input to the model is given using data generators which read the input images and feed them to the image classifier model. Here we have also rescaled the training and testing images in such a way that the pixel value ranges between [0,1]. The target size of the image has been set to (224,224), and the input images are resized to this value. The class mode is set to categorical, given the nature of the output.

Table 2. InfraredSolarModule Dataset Segregation

Serial No.	Input Data	Number of Images
1	Training Images	2310
2	Validation Images	665
3	Testing Images	665

3) **Model Training and Defining Model Parameters**: The learning rate of the model has been set to 5E-05, epoch to 30, and additionally a checkpoint has been created where the model with the best validation accuracy is being saved. The model is compiled using categorical cross-entropy loss, and the optimizer used is Adam. The model uses a pre-trained deep learning network [10] VGG16 as the basis of the solar image classifier model, and then it is retrained on our dataset. The parameter "pooling= 'max' is used because rather than connecting the convolutional base of the VGG16 model to a couple of fully connected layers before the final output layer (which is done in the original VGG16 model) [10] and it takes the average of each feature map and feeds it directly into the SoftMax layer [3], we rather use a max-pooling output as it seemed to work better than average pooling in this case [3], [10]. This approach is an alternative to using fully connected layers to transition from feature maps to an output prediction for the model. This makes the model less prone to overfitting. The model is retrained on all the layers without freezing any of the layers of the VGG16 network.

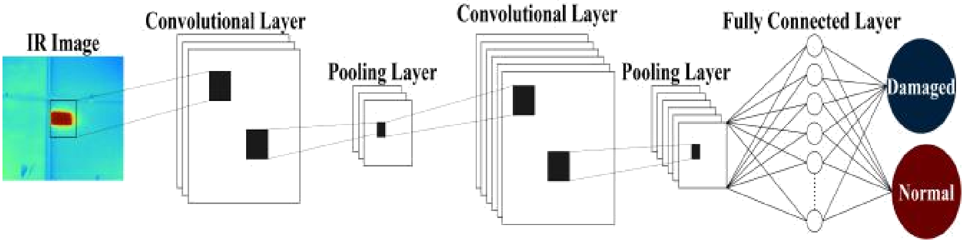
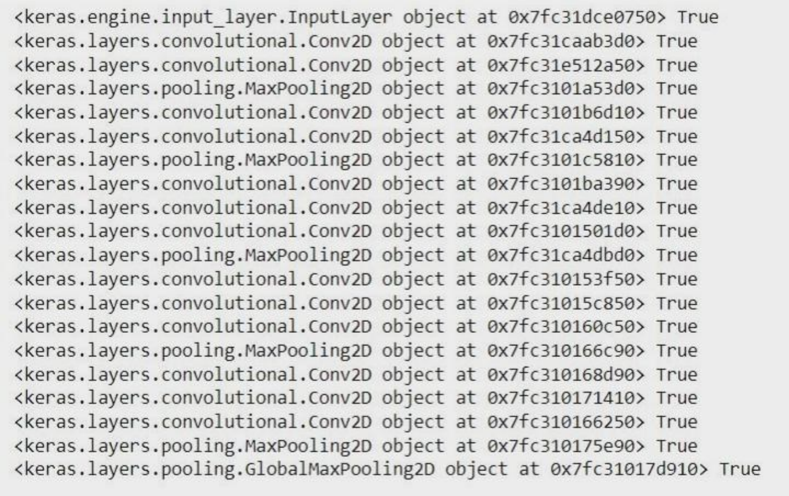


Fig 3. Basic architecture of CNN model for the Fault Detection system.

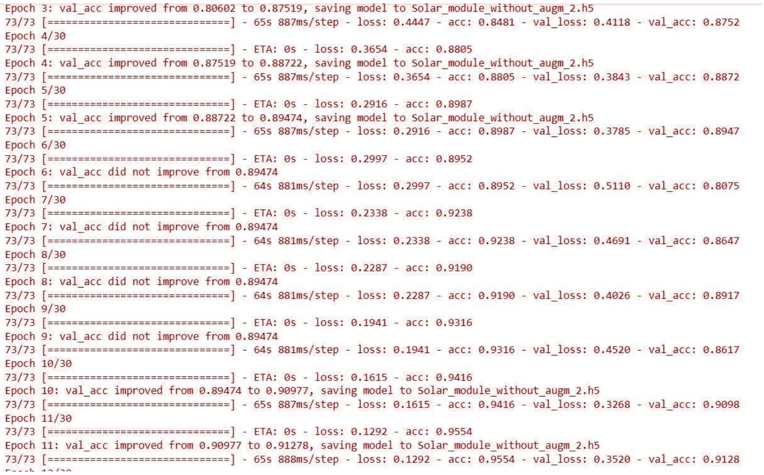
4) **Very Deep Convolutional Neural Network for Large-Scale Image Recognition (VGG16) Architecture:** The input to a VGG16 has to be a fixed 224 x 224 RGB image. The architecture starts with two sets of 2 convolutional layers followed by 1 max-pooling layer, further it has 3 sets of 3 convolutional layers followed by 1 max-pooling layer. After this, there are 3 fully connected layers in the form of 1 x 1 x 1000, i.e the output shows 1000 classes as it was trained on the ImageNet dataset. This final layer can be built as to our preference based on the number of classes in our dataset, which is 4 in this case. VGG16 seemed to capture more complex features as compared to AlexNet because of the deep neural network that it is and due to the increased layers. The convolutional base layers were frozen as they captured the general features. The final classifier was built on top of it according to the problem as it captures problem-specific features. The last layer was to be flattened and then a dense layer was created, with 4 output classes and the activation function used was softmax. The softmax function classes the output probabilities of each class in the range [0,1]. These probabilities add up to 1 and therefore are better used for multinomial logistic regression as compared to binary classification.



```
<keras.engine.input_layer.InputLayer object at 0x7fc31dce0750> True
<keras.layers.convolutional.Conv2D object at 0x7fc31caab3d0> True
<keras.layers.convolutional.Conv2D object at 0x7fc31e512a50> True
<keras.layers.pooling.MaxPooling2D object at 0x7fc3101a53d0> True
<keras.layers.convolutional.Conv2D object at 0x7fc3101b6d10> True
<keras.layers.convolutional.Conv2D object at 0x7fc31ca4d150> True
<keras.layers.pooling.MaxPooling2D object at 0x7fc3101c5810> True
<keras.layers.convolutional.Conv2D object at 0x7fc3101ba390> True
<keras.layers.convolutional.Conv2D object at 0x7fc31ca4de10> True
<keras.layers.convolutional.Conv2D object at 0x7fc3101501d0> True
<keras.layers.pooling.MaxPooling2D object at 0x7fc31ca4dbd0> True
<keras.layers.convolutional.Conv2D object at 0x7fc310153f50> True
<keras.layers.convolutional.Conv2D object at 0x7fc31015c850> True
<keras.layers.convolutional.Conv2D object at 0x7fc310160c50> True
<keras.layers.pooling.MaxPooling2D object at 0x7fc310166c90> True
<keras.layers.convolutional.Conv2D object at 0x7fc310168d90> True
<keras.layers.convolutional.Conv2D object at 0x7fc310171410> True
<keras.layers.convolutional.Conv2D object at 0x7fc310166250> True
<keras.layers.pooling.MaxPooling2D object at 0x7fc310175e90> True
<keras.layers.pooling.GlobalMaxPooling2D object at 0x7fc31017d910> True
```

Fig 4. Trainable layers of VGG16 model summary

5) **Model Accuracy**: The model is fitted using a fit generator function which uses the train generator images and estimates the accuracy for the validation dataset and losses. The model with the highest validation accuracy is then saved to the Solar module file where the final model is saved (Fig.5) Ideally, the validation accuracy must increase for each epoch and the validation loss must keep on decreasing as we keep training the model, and eventually reach a steady value when our model is not able to learn any more useful information from our training data. By analyzing this trend, we can finalize the number of epochs required to train the model. From the output shown above, we see that the loss decreases while the accuracy increases during the training process. Each time the validation accuracy reaches a new maximum value, the checkpoint file is saved (output: "saving model to Solar_module_without_augm.h5". After the training has been completed, we then can load the checkpoint file which has the best validation accuracy during training. This is the final model which can also be used to create an interface, such as a web app to classify the model by simply uploading the images by the user.



```
Epoch 3: val_acc improved from 0.80602 to 0.87519, saving model to Solar_module_without_augm_2.h5
73/73 [==============================] - 65s 887ms/step - loss: 0.4447 - acc: 0.8481 - val_loss: 0.4118 - val_acc: 0.8752
Epoch 4/30
73/73 [==============================] - ETA: 0s - loss: 0.3654 - acc: 0.8805
Epoch 4: val_acc improved from 0.87519 to 0.88722, saving model to Solar_module_without_augm_2.h5
73/73 [==============================] - 65s 887ms/step - loss: 0.3654 - acc: 0.8805 - val_loss: 0.3843 - val_acc: 0.8872
Epoch 5/30
73/73 [==============================] - ETA: 0s - loss: 0.2916 - acc: 0.8987
Epoch 5: val_acc improved from 0.88722 to 0.89474, saving model to Solar_module_without_augm_2.h5
73/73 [==============================] - 65s 887ms/step - loss: 0.2916 - acc: 0.8987 - val_loss: 0.3785 - val_acc: 0.8947
Epoch 6/30
73/73 [==============================] - ETA: 0s - loss: 0.2997 - acc: 0.8952
Epoch 6: val_acc did not improve from 0.89474
73/73 [==============================] - 64s 881ms/step - loss: 0.2997 - acc: 0.8952 - val_loss: 0.5110 - val_acc: 0.8075
Epoch 7/30
73/73 [==============================] - ETA: 0s - loss: 0.2338 - acc: 0.9238
Epoch 7: val_acc did not improve from 0.89474
73/73 [==============================] - 64s 881ms/step - loss: 0.2338 - acc: 0.9238 - val_loss: 0.4691 - val_acc: 0.8647
Epoch 8/30
73/73 [==============================] - ETA: 0s - loss: 0.2287 - acc: 0.9190
Epoch 8: val_acc did not improve from 0.89474
73/73 [==============================] - 64s 881ms/step - loss: 0.2287 - acc: 0.9190 - val_loss: 0.4026 - val_acc: 0.8917
Epoch 9/30
73/73 [==============================] - ETA: 0s - loss: 0.1941 - acc: 0.9316
Epoch 9: val_acc did not improve from 0.89474
73/73 [==============================] - 64s 881ms/step - loss: 0.1941 - acc: 0.9316 - val_loss: 0.4520 - val_acc: 0.8617
Epoch 10/30
73/73 [==============================] - ETA: 0s - loss: 0.1615 - acc: 0.9416
Epoch 10: val_acc improved from 0.89474 to 0.90977, saving model to Solar_module_without_augm_2.h5
73/73 [==============================] - 65s 887ms/step - loss: 0.1615 - acc: 0.9416 - val_loss: 0.3268 - val_acc: 0.9098
Epoch 11/30
73/73 [==============================] - ETA: 0s - loss: 0.1292 - acc: 0.9554
Epoch 11: val_acc improved from 0.90977 to 0.91278, saving model to Solar_module_without_augm_2.h5
73/73 [==============================] - 65s 888ms/step - loss: 0.1292 - acc: 0.9554 - val_loss: 0.3520 - val_acc: 0.9128
```

Fig 5. Training of model

6) **Accuracy and Loss Visualization:**

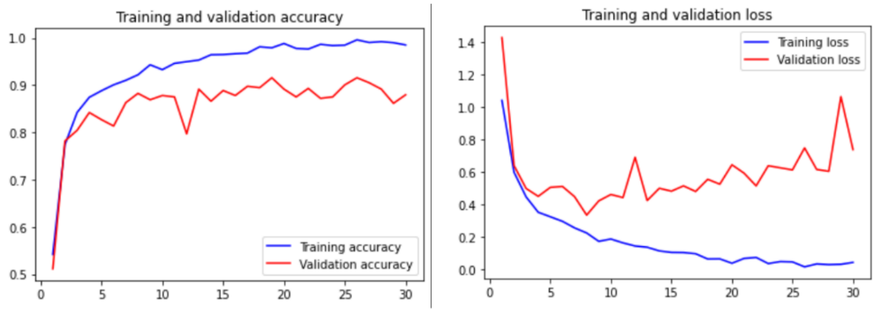


Fig. 6. Training and Validation accuracy and loss obtained

As shown in the above figure. 6, the training accuracy is shown using the blue line, and almost after 20 epochs, it reaches an accuracy of almost 1 (which represents classifying 100% of the training images correctly) and then tapers off from there. However, the validation accuracy is the accuracy measured on the validation set, which is the accuracy we care about. In this case, the accuracy leveled off at around 95%, meaning that we successfully classified almost all of the images in our validation set to the correct category. In a summary, we have finally visualized subsets of testing images to get an idea of how accurately the model can classify the solar panels.

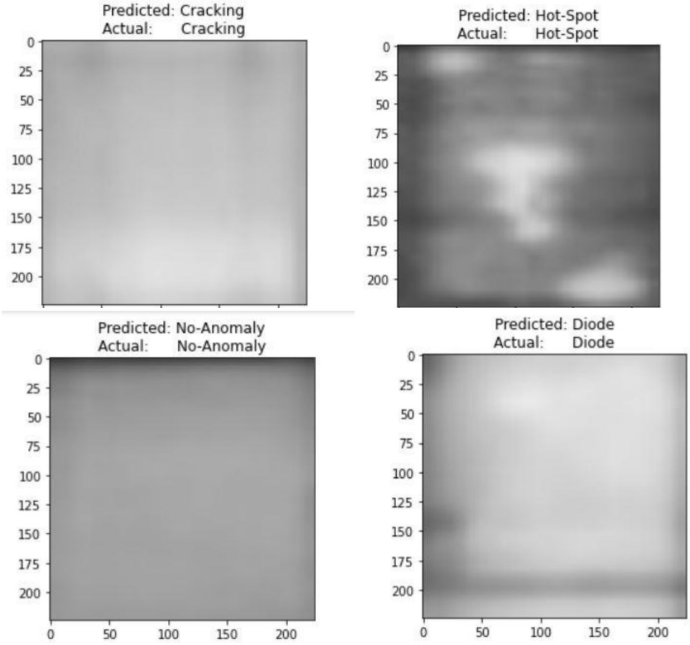


Fig 7. Classification of solar panel images by model

4 Results & Conclusion

In this paper, we propose a deep learning model to classify Photovoltaic modules as malfunctioning or normal. For this method, the model is trained with IR images that are read using data generators, preprocessed by using data augmentation, and built upon the VGG16 model which gives a 90.7% accuracy. The model has used transfer learning by training all the layers of the VGG16 model [10] to fit according to the InfraredSolarModule Dataset [4]. Due to this, the accuracy of the proposed model is better as compared to the accuracy obtained using K-means clustering (85%) or using DBSCAN clustering [8], which gives an accuracy of around 87%. The accuracy of the model is obtained by using the accuracy score function in the sklearn metrics library. The user can save the cost that goes under buying a new panel or repairing the less damaged solar panels by detecting the anomaly in the panel beforehand or in the early stages, which would then increase the lifetime of the solar panel by 5-10 years. The performance of this model can be further increased by adding layers manually to the architecture of the model. Therefore, we recommend using VGG16 along with transfer learning to classify malfunctions more accurately.

References

[1] L. Bommes, T. Pickel, C. Buerhop-Lutz, J. Hauch, C. Brabec, and I. M. Peters, “Computer Vision Tool for Detection, Mapping and Fault Classification of PV Modules in Aerial IR Videos,” Jun. 2021.

[2] M. V. C. V. da Costa *et al.*, “Remote sensing for monitoring photovoltaic solar plants in Brazil using deep semantic segmentation,” *Energies*, vol. 14, no. 10, May 2021

[3] S. Naveen Venkatesh and V. Sugumaran, “Fault Detection in aerial images of photovoltaic modules based on Deep learning,” IOP Conference Series: Materials Science and Engineering, vol. 1012, no. 1, p. 012030, Jan. 2021, doi: 10.1088/1757-899x/1012/1/012030.

[4] M. Millendorf, E. Obropta, and N. Vadhavkar, “INFRARED SOLAR MODULE DATASET FOR ANOMALY DETECTION.”

[5] J. Zhao, Q. Sun, N. Zhou, H. Liu, and H. Wang, “A Photovoltaic Array Fault Diagnosis Method Considering the Photovoltaic Output Deviation Characteristics,” International Journal of Photoenergy, vol. 2020, p. 2176971, 2020, doi: 10.1155/2020/2176971.

[6] A. Appiah, X. Zhang, B. Ayawli, and F. Kyeremeh, “Review and Performance Evaluation of Photovoltaic Array Fault Detection and Diagnosis Techniques,” International Journal of Photoenergy, vol. 2019, pp. 1–19, Feb. 2019, doi: 10.1155/2019/6953530.

[7] O. Menéndez, R. Guamán, M. Pérez, and F. A. Cheein, “Photovoltaic modules diagnosis using artificial vision techniques for artifact minimization,” *Energies*, vol. 11, no. 7, 2018.

[8] IEEE Singapore Section, IEEE Region 10, and Institute of Electrical and Electronics Engineers, *Proceedings of the 2016 IEEE Region 10 Conference (TENCON): November 22-25, 2016, Marina Bay Sands, Singapore.*

[9] International Neural Network Society, IEEE Computational Intelligence Society, and Institute of Electrical and Electronics Engineers, *2015 International Joint Conference on Neural Networks (IJCNN): date 12-17 July 2015*

[10] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” Sep. 2014, [Online]. Available: http://arxiv.org/abs/1409.1556

[11] M. Lin, Q. Chen, and S. Yan, “Network In Network,” Dec. 2013, [Online]. Available: http://arxiv.org/abs/1312.4400

[12] Alsafasfeh, M.; Abdel-Qader, I.; Bazuin, B.; Alsafasfeh, Q.; Su, W. Unsupervised Fault Detection and Analysis for Large Photovoltaic Systems Using Drones and Machine Vision. Energies 2018, 11, 2252. https://doi.org/10.3390/en11092252

[13] Xiaoliang Qian, Heqing Zhang, Huanlong Zhang, Yuanyuan Wu, Zhihua Diao, Qing-E Wu, and Cunxiang Yang. Solar Cell Surface Defects Detection based on Computer Vision [J]. Int J Performability Eng, 2017, 13(7): 1048-1056

[14] Robinson Cavieres, Rodrigo Barraza, Danilo Estay, José Bilbao, Patricio Valdivia-Lefort, Automatic soiling and partial shading assessment on PV modules through RGB images analysis, Applied Energy, Volume 306, Part A, 2022, 117964, ISSN 0306-2619, https://doi.org/10.1016/j.apenergy.2021.117964

[15] K. M. Sundaram, A. Hussain, P. Sanjeevikumar, J. B. Holm-Nielsen, V. K. Kaliappan and B. K. Santhoshi, "Deep Learning for Fault Diagnostics in Bearings, Insulators, PV Panels, Power Lines, and Electric Vehicle Applications—The State-of-the-Art Approaches," in IEEE Access, vol. 9, pp. 41246-41260, 2021, doi: 10.1109/ACCESS.2021.3064360

[16] Antonio Di Tommaso, Alessandro Betti, Giacomo Fontanelli, Benedetto Michelozzi, A multi-stage model based on YOLOv3 for defect detection in PV panels based on IR and visible imaging by unmanned aerial vehicle, Renewable Energy, 2022, ISSN 0960-1481, https://doi.org/10.1016/j.renene.2022.04.046

[17] L. Hernández-Callejo, S. Gallardo-Saavedra, J. I. Morales-Aragonés, V. Alonso-Gómez, A. R. Plaza, and D. F. Martínez, “Methodology for Inspection of Defects in Photovoltaic Plants by Drone and Electroluminescence,” *Communications in Computer and Information Science*, vol. 1555 CCIS, pp. 3–14, 2022, doi: 10.1007/978-3-030-96753-6_1.

[18] Z. Yahya, S. Imane, H. Hicham, A. Ghassane, and E. Bouchini-Idrissi Safia, “Applied imagery pattern recognition for photovoltaic modules’ inspection: A review on methods, challenges and future development,” *Sustainable Energy Technologies and Assessments*, vol. 52, Aug. 2022, doi: 10.1016/j.seta.2022.102071.

[19] U. Otamendi, I. Martinez, M. Quartulli, I. G. Olaizola, E. Viles, and W. Cambarau, “Segmentation of cell-level anomalies in electroluminescence images of photovoltaic modules,” *Solar Energy*, vol. 220, pp. 914–926, May 2021, doi: 10.1016/J.SOLENER.2021.03.058.

[20] Z. Meng, S. Xu, L. Wang, Y. Gong, X. Zhang, and Y. Zhao, “Defect object detection algorithm for electroluminescence image defects of photovoltaic modules based on deep learning,” Energy Science and Engineering, vol. 10, no. 3, pp. 800–813, Mar. 2022, doi: 10.1002/ESE3.1056.