TinyML for fault diagnosis of Photovoltaic Modules using Edge Impulse Platform

Adel Mellit
Department of Electronics
University of Jijel
Jijel, Algeria
adel_mellit@univ-jijel.dz

Nicola Blasuttigh
Department of Engineering and Architecture, and Center for Energy, Environment and Transport Giacomo Ciamician, University of Trieste
Trieste, Italy
nicola.blasuttigh@units.it

Alessandro Massi Pavan
Department of Engineering and Architecture, and Center for Energy, Environment and Transport Giacomo Ciamician, University of Trieste
Trieste, Italy
apavan@units.it

***Abstract*—In this paper a fault diagnosis method for photovoltaic (PV) modules is developed using an open source Machine Learning (ML) platform (Edge impulse). The idea is to develop a TinyML to classify certain defects that can frequently occur on PV modules (e.g. dirty, degradation and dust deposit on PV modules), and then to integrate the impulse into an Edge device for real time application. In this regard a database of infrared thermography image was built and used. The model could be run locally without internet connection. This method could help users to diagnosis their PV modules and make decision about the maintenance schedule (cleaning or replacing of PV modules). Results clearly report the feasibility of the method with a mean accuracy of 93.4 %. The main advantage is that, thanks to this platform, embedded ML model could be developed quickly. Moreover, edge processes are not affected by the latency and bandwidth issues becoming outstanding methods for real-time diagnostics.**

Keywords—photovoltaic, fault classification, machine learning, deep convolutional neural network, edge device.

I. INTRODUCTION

Renewable energy is increasingly the focus of the scientific community, institutions, and public opinion due to the need to reduce the environmental impact of traditional energy sources. Among the various renewable technologies, photovoltaics is certainly one of the most promising, due to its ability to directly transform sunlight into electricity [1]. In recent years, the photovoltaic industry has made significant progress worldwide. In particular, there have been significant improvements in efficiency, durability, reliability and costs. The efficiency of photovoltaic (PV) modules has constantly improved, thanks to research and development of new materials and technologies. In addition, their reliability has also improved due to the increased focus on material quality and production, and the ability to withstand extreme weather conditions such as rain, wind, hail and snow. However, other several factors could affect the photovoltaic production such as thermal degradation, soiling and shading which could lead to hot spot effects. For these reasons, several studies have focused on the PV system monitoring and fault detection to ensure high reliability and reduce maintenance costs and times. Although various methods can be used for this purpose, the evaluation of infrared (IR) images allows fast and non-invasive faults detection and prompt action to solve triggering problems [2].

Several recent literature studies have faced these kinds of issues based on IR images processing in PV fault applications. For example, in [3] a fault detection and classification method for PV modules using thermal images supported by edge detection algorithm and ANN has been proposed. An IR images analysis to detect and localize hotspot has been performed and described in [4] where the type of failure as temporary or permanent is also taken into account. The use of Unmanned Aerial Vehicles (UAVs) for visual or thermal imaging has led to a breakthrough in this field, allowing for a relatively quick scan of the PV plant in real time avoiding its disconnection. In [5], an optimised, two-stage drone flight strategy for fault detection was proposed, allowing flight duration and operational time to be minimised. However, faults are detected based on classical image processing approaches. In [6], a CNN based algorithm (YOLO) has been used to detect and identify PV faults such as hotspots, bypass diode failure and cracks by using aerial images by obtaining a mean average precision of 84%. A deep learning (DL) approach has been used to detect PV faults based on a dataset of 42048 module infrared images in [7], where the performances of the DL model have been evaluated for different types of segmentation. In [8], eleven fault classes have been considered in order to fully describe the PV fault consequences using a CNN-based method for their classification with an accuracy up to 90% in specific failures. A large dataset from 28 sites was collected in [9] resulting in 93220 module IR images. From these, a DL model supported by data augmentation was developed to classify the faults into five different classes with an average F1-score of 94.52%.

However, even if these methods lead to great performances, the cost-effectiveness of the proposed devices and instrumentation is in doubt. Regarding this aspect, recent research development has led the implementation of artificial intelligence algorithms also in low-cost edge embedded devices where the computational performances can be lower. In this aspect, TinyML is one of the most promising embedded machine learning (EML) platform for edge devices [10]. Also in this research field, literature studies have proposed different approaches and methodology in order to detect and classify PV faults on different scales. For example, in [11,12] embedded approaches based on thermal and visual images supported by neural networks has been proposed for PV module diagnosis of different defect classes on low-power edge devices. The studies demonstrate the feasibility of the system to operate in real-time obtaining good accuracy results and the interfacing of these systems with IoT platforms for smart remote monitoring purposes.

The main reasons and advantages to run the model locally, on edge device, are: i) portable device suitability, ii) no need to cloud interfacing, iii) enhances the data security and iv) reduce the latency.

Very limited works are related to the implementation of fault diagnosis of PV modules, based on IR images, into an edge device for real time application.

The main objective of the work is to develop a TinyML model for fault diagnosis of PV module. The optimized model could be easily integrated inside a low-cost and low-power edge device (e.g. Nano 33 LBE sense). In this work we try to verify the feasibility of a such approach in this field.

II. DataBase

The used database consists of 2000 IR thermography images collected for PV modules under normal and faulty conditions. The PV modules present four different fault classes: healthy (N), dirty (D2), degraded (D3) and sand deposit on PV module (D1) as shown in Figure 1.

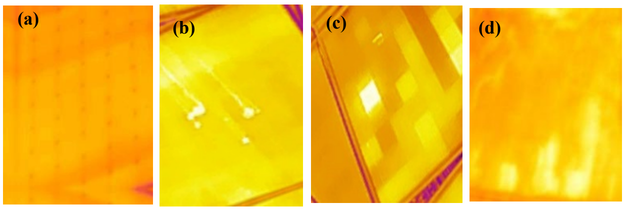


Fig.1. IR thermography images: a) healthy PV module (N), b) dirty PV module (D2), c) degraded PV module (D3) and d) sand deposit on PV module (D1).

Each class contains 500 IR images. The database was collected in a semi-arid location (South of Algeria, in a hot and dry desert location). Figure 2 shows the experimental setup with the examined PV modules and the used IR thermography camera (Uni-T Pro UTi260B) with a resolution of 256×192 pixels. All IR images were collected manually for solar irradiance greater than 500 W/m^2 with a perpendicular position of the camera on the PV module.

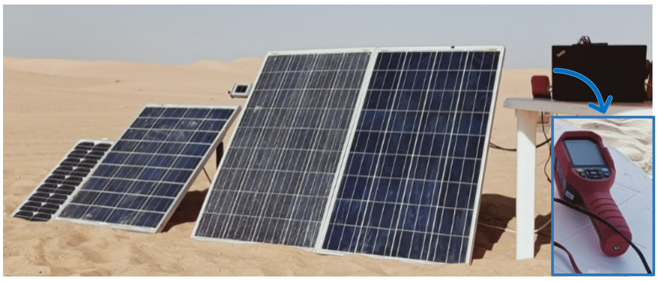


Fig.2 Test facility: the examined PV modules and IR camera

III. Methodology

The methodology used in this study aims to apply a deep convolutional neural network (DCNN) to classify some defects on PV modules and build the developed model for a real time application. To do this we use an open ML platform named Edge impulse. To develop the model, we use a CNN type MobilNetV2. Regarding the edge device, we choose a microcontroller board type Arduino Nano 33 BLE sense. Figure 3 depicts a workflow of the embedding procedure.

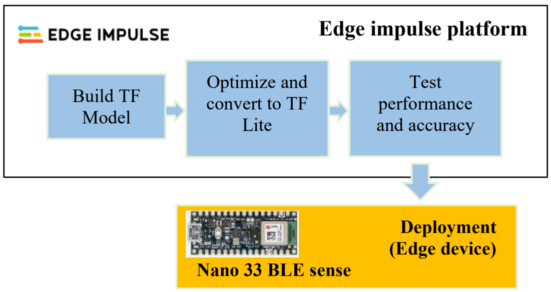


Fig.3. Workflow of embedding procedure

This comprises four blocks: i) build TensorFlow (TF) model, ii) optimize and convert the model to TF Lite, iii) test the performance of the model and iv) integrate the model inside the mentioned edge device and deploy it.

A. *Edge impulse platform*

Edge impulse is the leading development platform for ML on edge devices, founded in 2019 by Zach Shelby and Jan Jongboom. Edge Impulse provides maximum efficiency and speed on a wide range of hardware from MCUs to CPUs [13].

B. *The used neural network (MobileNetV2)*

Figure 4 shows the basic structure of MobielNetV2 [14]. It consists of 53 convolution layers and one average pooling. It comprises two main blocks: i) Inverted residual block and ii) Bottleneck residual block. It is based on an inverted residual structure where the residual connections are between the bottleneck layers. Each block has three layers (Convolution with ReLu6, Depthwise convolution and 1x1 convolution without any linearity). As shown in Fig. 4 there are two types of convolution layers: i) 1x1 convolution and ii) 3x3 Depthwise Convolution.

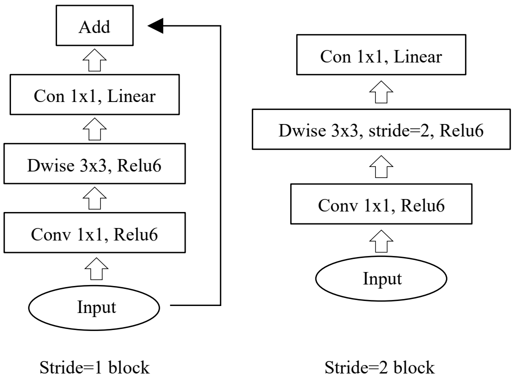


Fig.4 MobilNetV2 configuration

C. *Microcontroller board*

The selected microcontroller board is the Arduino Nano 33 BLE sense [15]. The main feature of this board, besides the impressive selection of sensors, is the possibility of running Edge computing applications (Artificial intelligence techniques) on it using TinyML. It helps to create ML models using TensorFlow™ Lite and upload them to the board using the Arduino IDE.

D. Procedure for model developement based on edge impulse

The procedure for developing the model using Edge impulse platform can be summarized in the following steps:

1) *Data-acquisition*: it consists of upload and label our IR images in four classes (D1, D2, D3 and N)

2) *Impulse design*: it comprises three steps:

   a) *Create impulse*: set up the size of the images, add a processing block and a learning block.

   b) *Image*: in this step we select the type of the IR image (RGB or Grayscale), and we generate the features of the classes.

   c) *Transfer learning*: here we define the CNN parameters (number of training cycle, learning rate, validation set, number of neurons in the end layer, and the value of dropout). Once these parameters are defined, we train the model.

3) *Retrain the model*: this step aims to retrain the model with known parameters.

4) *Live classification:* in this step we test the model with new data (IR images).

5) *Model testing:* set the expected outcome for each IR image to the desired outcome to automatically score the impulse.

6) *Deployment:* this step makes the model run without an internet connection, minimizes latency, and runs with minimal power consumption. Then, we have to select the hardware board (e.g., Nano 33 BLE sense) to build an optimized model for real time application (firmware). Differently, we can create a library and this can turn our impulse into optimized source code to be run on any device.

E. Performance metrics

To evaluate the performance of the classifier, we compute the F1-score and accuracy (*Acc*) given by the following expression:

$$F1-score = 2 * \frac{(Pre*Rec)}{(Rec+Pre)} \quad (1)$$

$$Acc = \frac{\sum_i CM(i,i)}{\sum_i \sum_j CM(i,j)} \quad (2)$$

where *Pre* is the precision, *Rec* is the recall and *CM* is the confusion matrix.

$$Pre = \frac{CM(i,i)}{\sum_j CM(j,i)} \quad (3)$$

$$Rec = \frac{CM(i,i)}{\sum_j CM(i,j)} \quad (4)$$

IV. Results and Discussion

After several experiments, the optimal model parameters are summarized in Table I.

The confusion matrix of the training performance is shown in Table II. As shown from the table, the accuracy is 96.2% and the loss is 0.11. The F1-score ranges between 93% to 99%. Overall, in terms of accuracy, these results are good and demonstrate the correct operation of the proposed solution. To check the effectiveness of the approach, we calculated the confusion matrix during the validation process using unknown IR images. Table III shows the confusion matrix of the validation performance.

TABLE I: MODEL PARAMETERS

Parametrs of the model MobileNetV2	Value
Number of training cycle	30
Learning rate	0.0006
Validation set size	20%
Dropout	0.2
Number of node at the end layer	16

TABLE II: CONFUSION MATRIX (MODEL TRAINING PERFORMANCE)

Class	D1 (%)	D2 (%)	D3 (%)	N (%)	Accuracy (%)	Loss
D1	98.7	1.3	0	0	96.2	0.11
D2	0	92.6	1.2	6.2		
D3	0	1.5	98.5	0		
N	0	4.4	0	95.6		
F1-score	99	93	98	95		

TABLE III CONFUSION MATRIX (MODEL VALIDATION PERFORMANCE)

Class	D1 (%)	D2 (%)	D3 (%)	N (%)	Uncertain	Accuracy (%)
D1	96.8	1.1	0	0	2.1	93.45
D2	1.0	85.1	1.0	6.9	5.9	
D3	0	0	99.1	0	0.9	
N	0	5.8	0	92.2	1.9	
F1-score	98	89	99	93		

The accuracy is 93.45% and the F1-score ranges between 93% to 99%. Also in this case, the accuracy is still good, and the considered faults are correctly classified. Based on the uncertain and the F1-score the class D3 can be classified correctly with an accuracy of 99.1%. The class D2 has a high uncertain value, and it is the worst classified class with an accuracy of 85.1%. Features explore are shown in Fig.5.

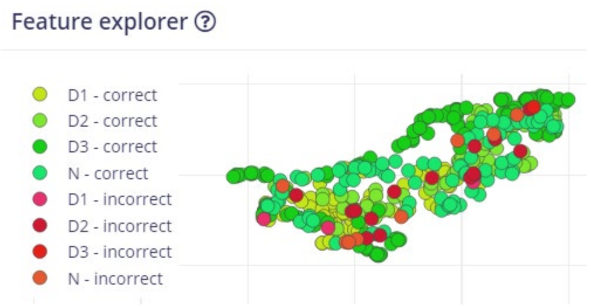
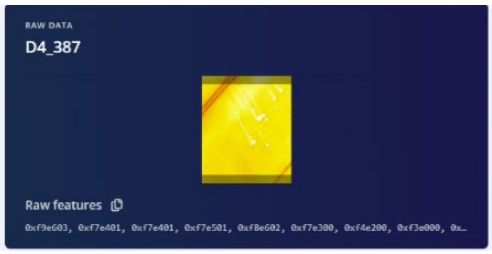


Fig.5. Feature explorer of the four classes during the validation process

Figure 6 shows an example of a live classification. An unknown image, a dirty PV module (class D2) in this case, was selected to check the model ability to classify this image.



RAW DATA

D4_387

Raw features

0xf9e603, 0xf7e401, 0xf7e401, 0xf7e501, 0xf8e602, 0xf7e300, 0xf4e200, 0xf3e000, 0x…

Fig.6 Selected IR image for testing the model

The results are reported in Table IV. As can be seen the model classify the given IR image with good accuracy. The predicted class is D2 belonging to the case of dirty PV modules, as expected.

TABLE IV MODEL VALIDATION USING UNKNOWN IR IMAGE

Name	D4_387
Expected outcome	D2
Class	
D1	0
D2	1
D3	0
N	0
Uncertain	0

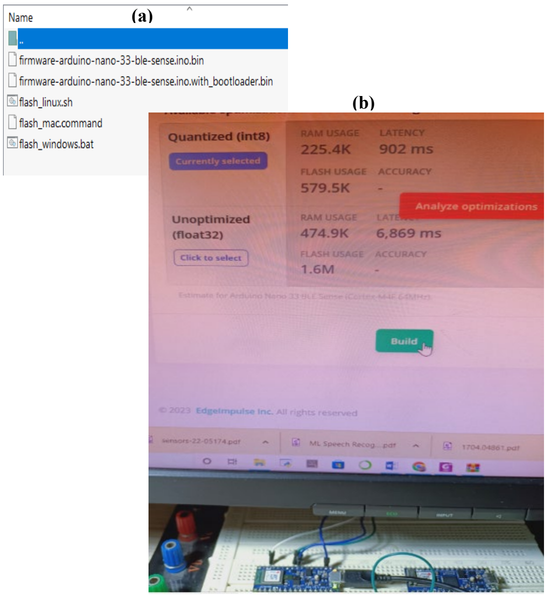
In order to minimize hardware resources and latency, an optimization process of the proposed model has been performed. In this regard, a comparison between the non-optimized and optimized model is shown in Table V where, for each model, the RAM usage (kB), the flash memory usage (MB) and the latency (ms) have been evaluated. For each of them, the percentage reduction is calculated between the optimized and the non-optimized model.

TABLE V COMPARISON BETWEEN OPTIMIZED AND UNOPTIMIZED MODELS

Optimizer	**RAM usage (KB)**	**Flash usage (MB)**	**Latency (ms)**
Non-optimized model (float32)	474.9	1.6	6,600
Optimized model quantized (Int8)	225.4	0.58	904
Reduction	52 %	63 %	86 %

As can be seen from Table V, the optimize model RAM has been reduced compared to the non-optimized model with a percentage reduction of 52%. The flash memory usage is also reduced to 0.58 MB, leading to a percentage reduction of 63%. The latency is also significantly decreased up to 904 ms compared to 6,600 ms for the non-optimized model. This means that the optimized code can be run about 7 times faster than the non-optimized one.

The last step consists of building the optimized model and download it into the selected edge device. Figure 7a shows the optimized files from the build model, and Fig. 7b presents the uploaded model into the Nano 33 BLE sense.



Name

..

firmware-arduino-nano-33-ble-sense.ino.bin

firmware-arduino-nano-33-ble-sense.ino.with_bootloader.bin

flash_linux.sh

flash_mac.command

flash_windows.bat

Quantized (int8) — Currently selected — RAM USAGE 225.4K, LATENCY 902 ms, FLASH USAGE 579.5K, ACCURACY -

Unoptimized (float32) — Click to select — RAM USAGE 474.9K, LATE… 6,869 ms, FLASH USAGE 1.6M, ACCURACY -

Estimate for Arduino Nano 33 BLE Sense (Cortex-M4F 64MHz)

Fig.7. a) The generated files after building the model and b) downloading the code inside the Nano 33 BLE sense.

V. CONCLUSIONS AND PERSPECTIVES

In this work, an embedded solution for PV module fault diagnosis based on thermographic images was proposed. The idea is to incorporate the MobileNetV2 model into a low-power and low-cost (~43 $) edge device (Nano 33 BLE sense) for a real-time application with the purpose of detecting and classifying four different fault classes. The results obtained from the study and the proposed hardware showed that the system works properly with an average classification accuracy of 93.4%, leading to good classification of the considered fault classes. Moreover, the model has been optimized taking into consideration the utilization of available hardware and latency times, resulting in a model RAM size reduction of up to 52%, a flash usage reduction of 63% and a latency reduction of 86%. This shows how the optimization process can lead to a significant improvement in running performance and thus allow the model to work more efficiently.

The feasibility of the proposed solution is justified, and the model is ready to be deployed by using an infrared camera to receive IR image and make diagnosis using the embedded classifier, which is the future plan of our work. Through this method, it is therefore possible to classify the proposed photovoltaic modules faults cost-effectively and, despite the use of limited hardware, with good accuracy. This approach allows this solution to be integrated into several applications. In fact, the edge device could be integrated and used by UAV equipped with an IR camera to diagnose faulty PV modules, saving time and thus money and resources.

ACKNOWLEDGMENT

This work was supported in part by the “Ministero dell’Istruzione, dell’Università e della Ricerca” (Italy) under the Grant PRIN2020–HOTSPHOT 2020LB9TBC. The first author would like to acknowledge support from the ICTP through the Associates Programme (2023–2028).

REFERENCES

[1] IEA (2022), Solar PV, IEA, Paris https://www.iea.org/reports/solar-pv, License: CC BY 4.0

[2] J. A. Tsanakas, L. Ha, and C. Buerhop, "Faults and infrared thermographic diagnosis in operating c-Si photovoltaic modules: A review of research and future challenges", *Renewable and Sustainable Energy Reviews*, vol. 62, pp. 695–709, Sep. 2016

[3] V. S. Bharath. Kurukuru, A. Haque, and M. A. Khan, "Fault Classification for Photovoltaic modules using Thermography and Image Processing", in *2019 IEEE Industry Applications Society Annual Meeting*, Sep. 2019, pp. 1–6.

[4] M. Alajmi, K. Awedat, M. S. Aldeen and S. Alwagdani, "IR Thermal Image Analysis: An Efficient Algorithm for Accurate Hot-Spot Fault Detection and Localization in Solar Photovoltaic Systems," 2019 IEEE International Conference on Electro Information Technology (EIT), Brookings, SD, USA, 2019, pp. 162-168.

[5] V. Lofstad-Lie, E. S. Marstein, A. Simonsen, and T. Skauli, "Cost-Effective Flight Strategy for Aerial Thermography Inspection of Photovoltaic Power Plants", *IEEE Journal of Photovoltaics*, vol. 12, no. 6, pp. 1543–1549, Nov. 2022.

[6] N. Prajapati, R. Aiyar, A. Raj and M. Paraye, "Detection and Identification of faults in a PV Module using CNN based Algorithm," 2022 3rd International Conference for Emerging Technology (INCET), Belgaum, India, 2022, pp. 1-5.

[7] D. Rocha, M. Lopes, J. P. Teixeira, P. A. Fernandes, M. Morais and P. M. P. Salome, "A Deep Learning Approach for PV Failure Mode Detection in Infrared Images: First Insights," 2022 IEEE 49th Photovoltaics Specialists Conference (PVSC), Philadelphia, PA, USA, 2022, pp. 0630-0632.

[8] R. H. Fonseca Alves, G. A. de Deus Júnior, E. G. Marra, and R. P. Lemos, "Automatic fault classification in photovoltaic modules using Convolutional Neural Networks", *Renewable Energy*, vol. 179, pp. 502–516, Dec. 2021.

[9] Y. Zefri, I. Sebari, H. Hajji, and G. Aniba, "Developing a deep learning-based layer-3 solution for thermal infrared large-scale photovoltaic module inspection from orthorectified big UAV imagery data", *International Journal of Applied Earth Observation and Geoinformation*, vol. 106, p. 102652, Feb. 2022.

[10] H. Han and J. Siebert, "TinyML: A Systematic Review and Synthesis of Existing Research," 2022 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Jeju Island, Korea, Republic of, 2022, pp. 269-274, doi: 10.1109/ICAIIC54071.2022.9722636.

[11] A. Mellit, "An embedded solution for fault detection and diagnosis of photovoltaic modules using thermographic images and deep convolutional neural networks", *Engineering Applications of Artificial Intelligence*, vol. 116, p. 105459, Nov. 2022.

[12] A. Mellit, M. Benghanem, S. Kalogirou, and A. Massi Pavan, "An embedded system for remote monitoring and fault diagnosis of photovoltaic arrays using machine learning and the internet of things", *Renewable Energy*, vol. 208, pp. 399–408, May 2023.

[13] 'Edge Impulse'. https://www.edgeimpulse.com/ (accessed Apr. 07, 2023).

[14] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. -C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018, pp. 4510-4520.

[15] 'Arduino Nano 33 BLE Sense', *Arduino Official Store*. https://store.arduino.cc/products/arduino-nano-33-ble-sense (accessed Apr. 07, 2023).