



## AI-Based Smart Real-Time PV Panels Soiling Recognizing System Using Deep Neural Network Framework on NVIDIA Jetson Nano Embedded GPU

Hajar El Karch<sup>1\*</sup>, Rachid El Gouri<sup>1</sup>, Youssef Natij<sup>2</sup>, Mohamed Benally<sup>2</sup>, Abdelkader Mezouari<sup>2</sup>

<sup>1</sup>Laboratory of Advanced Systems Engineering National School of Applied Sciences, Ibn Tofail University, Kenitra 14000, Morocco

<sup>2</sup>Laboratory of Electronic Systems, Information Processing, Mechanics and Energetics, Faculty of Sciences, Ibn Tofail University, Kenitra 14000, Morocco

Corresponding Author Email: [Hajar.elkarch@uit.ac.ma](mailto:Hajar.elkarch@uit.ac.ma)

Copyright: ©2024 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/isi.290503>

### ABSTRACT

**Received:** 27 December 2023

**Revised:** 1 May 2024

**Accepted:** 7 June 2024

**Available online:** 24 October 2024

#### Keywords:

*Deep Learning Networks, CNN, image processing, YOLOv8, PV panel inspection, soiling detection, embedded systems, NVIDIA Jetson Nano*

On a global scale, the dust accumulation and soiling of solar photovoltaic (PV) systems, are the factors that after irradiation, have the most serious impact on system performance. In arid regions, soiling can severely affect large utility-scale PV plants, requiring major cleaning and mitigation efforts. This mitigation leads to increased operating and capital expenditures, thereby reducing overall revenues. This research paper introduces a novel solution to this issue by integrating the YOLOv8 Deep Neural Network (DNN) with advanced image processing techniques for dust and soiling classification to suggest the appropriate cleaning intervention strategy. Our proposed system explores the synergy of artificial intelligence algorithms and embedded electronic platforms to present an innovative system designed for high-performance, autonomous, and real-time dust and soiling classification and enhances the accuracy and efficiency of solar panel status inspection. Our approach was implemented on the NVIDIA Jetson Nano platform, which demonstrates its powerful performance capabilities, low power consumption, and efficient processing time. Through our evaluation of these embedded systems, we contribute to identifying optimal hardware for our application and provide valuable insights for researchers implementing AI-based solutions in solar energy. With an accuracy rate exceeding 90%, our system effectively detects dirt and dust in various scenarios.

## 1. INTRODUCTION

The monitoring and maintenance of photovoltaic panels have become crucial, presenting challenges that require innovative inspection and diagnostic systems to enhance their electrical output, efficiency, dependability, and stability. Therefore, researchers in the solar photovoltaic systems field are focused on improving their performance, enabling them to generate more electricity efficiently and in an environmentally friendly way. The study of dust and soiling impact on the power productivity of photovoltaic (PV) modules or systems is crucial for maximizing the efficiency of PV generators. Several studies by Hamid et al. [1], Semaoui et al. [2], and Quansah and Adaramola [3] highlight the significance of addressing dust and soiling in PV systems in arid regions of Africa and compare the energy efficiency and output performance of photovoltaic modules under varying environmental and operational conditions in Zeedan et al. [4], Mustafa et al. [5] and Salimi et al. [6] provide a study of the impact of dust soiling on solar panel performance in dusty regions of the Middle East. Table 1 presents the performance degradation percentages due to dust soiling in Africa and the Middle East.

Another experimental study [7] examined the impact of dust

on azimuth tracking solar PV systems in the desert regions of Sharjah, United Arab Emirates. This demonstrates that dust accumulation during daily exposure periods is the main factor reducing the productivity of solar energy technologies. In a review of failures of photovoltaic modules, Köntges et al. [8] found that soiling negatively influences power production. It reduces the transmittance of incident radiation through the cover glass of PV modules, thereby decreasing the amount of solar radiation available for conversion into electricity.

Those studies have consistently demonstrated that the accumulation of dust particles on PV panels significantly reduces their performance and underscores the critical need to address dust and soiling in solar systems to ensure maximum energy efficiency. Various cleaning techniques and technologies are used for the maximum operating efficiency of PV panels. Electrostatic cleaning was proposed by Kawamoto and Shibata [9] for airborne dust particles using wire electrodes and single-phase high-voltage to create an electrostatic force. Kawamoto and Guo [10] enhanced their electrostatic cleaning technique three years later using a two-phase, high-voltage and low-frequency approach. The results of their experiments indicated that this improved electrostatic cleaning method achieved a cleaning rate of more than 60% for very fine dust particles. Simiyu [11] optimizes the cleaning

process by determining the ideal number of modules to be cleaned and the frequency of cleaning events per year using machine cleaning for PV panels. Al-badra et al. [12] tested a complex self-cleaning system in Egypt, combining a mechanical vibrator with an antistatic hydrophilic nano-coating. This system effectively removed dust particles, resulting in a 12.94% reduction in electrical efficiency compared to non-cleaned panels.

**Table 1.** The percentage of photovoltaic performance reduction due to dust accumulation

Country	Performance Degradation	Exposure Time
Egypt [1]	53% of the conversion efficiency of the PV unit $\eta_{pv}$	75 days
Algeria [2]	8.79% of the $I_{sc}$ per day	1 month
Ghana [3]	8%-13.8% of the Performance Ratio /month	14 months
Qatar [4]	43% of the cumulative produced power	6 months
Jordan [5]	8.8% of the produced power/ day	21 days
Iran [6]	8%-12% of produced power/ month	45 days

In the last several years, there has been substantial growth in the adoption of photovoltaic technologies and development in utilizing techniques of deep learning, especially Convolutional Neural Networks (CNNs) and artificial intelligence (AI), for evaluating the state of photovoltaic (PV) systems. These advanced technologies offer innovative solutions for monitoring, assessing, and optimizing the performance of PV installations and significantly improve the efficiency, reliability, and maintenance of PV systems.

One of the early works about the applicability of this new evaluation method has been proposed by Ul Mehmood et al. [13], which introduces an efficient and cost-effective solution for measuring the soiling ratio on PV panels by combining Internet of Things (IoT), cloud-based infrastructure, low-cost sensors, and ANN models. The system adopts an IoT framework, enabling connectivity and data collection from distributed sensors. The Low-cost sensors are used to gather data related to soiling deposits on the PV panels. These sensors help minimize the hardware requirements for a soiling monitoring station. The cloud-based centralized approach allows for remote monitoring of soiling deposits on PV panels, eliminating the need for physical presence at the installation site. The results of the soiling ratio measurements are transmitted from the cloud server to a graphical user interface (GUI). This GUI allows users to access and visualize the soiling data, making monitoring the status of PV panels remotely accessible. Another study by Cavieres et al. [14] proposed an approach based on Convolutional Neural Networks (CNN) to estimate the power loss of PV modules affected by soiling, dust and dirt accumulation, and partial shading. The methodology uses visible spectrum RGB images of multiple solar panels and environmental data to individually predict the performance of each module. This methodology offers a comprehensive approach to assessing the performance of PV modules by considering the specific effects of partial shading and soiling on each module within a solar installation. By leveraging CNNs and image analysis techniques, the algorithm can accurately estimate power loss for each module displayed in the image. Tan et al. [15] present a solution in their study by building an image acquisition system designed to assess the dust accumulation status on photovoltaic panels.

The proposed methodology is based on a novel approach that uses Denoising Convolutional Neural Networks (DnCNN) for image analysis and evaluation. The system captures images of PV panels to assess the extent of dust accumulation in real-time. The study compares different combinations of DnCNN with popular deep-learning models like AlexNet, ResNet, and VGG-16. The results suggest that a serial connection of the ResNet-50 model and the DnCNN enables real-time monitoring tasks and quantitative assessment of the state of dust accumulation to be carried out with greater accuracy and better performance in time.

Li et al. [16] perform a systematic study on the application of Artificial Neural Networks (ANN) for photovoltaic fault detection and diagnosis. Three types of Neural Networks for PV Fault Detection and Diagnosis were discussed: Shallow Neural Networks, Deep Neural Networks, and hybrid models, which combine ANNs with other machine learning methods. The review paper provides a clear framework for discussing the use of ANNs in PV FDD and the associated challenges and prospects in the field. The paper summarizes the effectiveness of ANNs in PV fault detection and diagnosis and highlights the common architectural choices for shallow and deep ANNs. This study provides a clear direction for future research interested in applying machine learning techniques to improve the reliability and maintenance of PV systems.

Several studies [17-19] have focused on exploring the use of Deep Learning Networks and embedded systems for detection tasks using YOLO algorithms. This research paper presents an innovative real-time approach that integrates YOLOv8 deep learning-based object detection techniques with image processing methods. Our approach allows the YOLOv8 to perform instant identification and classification of various types of detected soiling on the surfaces of PV panels. This process involves training the YOLOv8 network on a large dataset with multiple images, including a wide array of soiling types, enabling the system to recognize and classify different types of soiling accurately. Image processing techniques using OpenCV are used to extract the covered dust area.

The paper is organized into the following sections: System methodology, the core of our paper, details our novel approach. It introduces the status evaluation method based on the YOLOv8 algorithm and image processing approaches for soiling and dust detection. We also discuss the training dataset, which forms the foundation of our system's capability to detect and classify different types of soiling accurately. Section 3 presents an experimental case study where we apply our novel system to real-world scenarios. We also analyze the results obtained through the system's implementation. This part of the paper demonstrates the practicality and effectiveness of our approach in real-world situations. The final section of the paper draws conclusions based on our research results. We outline potential directions for future research to develop and conceive an intelligent inspection system for real-time soiling recognition and autonomous cleaning of solar modules.

## 2. METHODOLOGY AND MATERIALS

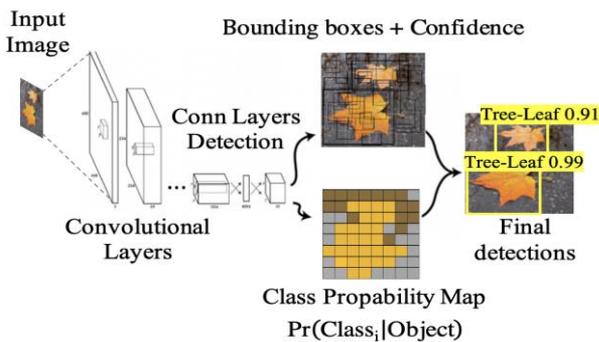
The proposed system uses two distinct methods to identify soiling and detect dust layers on the surface of solar panels. One method relies on the deep learning algorithm YOLOv8, while the other is based on image processing techniques. The YOLOv8 algorithm is a state-of-the-art object detection algorithm that excels in real-time applications. YOLOv8 is

trained to recognize and localize instances of soiling and dust on photovoltaic panels, providing a powerful tool for automated detection. Simultaneously, the system leverages image-processing techniques to enhance its capability to identify and analyze dust areas. Image processing involves manipulating the visual data captured by the HD camera to extract meaningful information. Various image-processing algorithms may be employed to enhance contrast, reduce noise, and highlight specific features associated with soiling or dust deposition. This complementary approach ensures a comprehensive and robust system for real-time monitoring of the solar panel's cleanliness.

## 2.1 Object detection using YOLO detection network

Object detection has gained popularity in computer vision due to its wide range of applications. It is a fundamental aspect of computer vision in various real-world scenarios such as security systems, autonomous driving, video monitoring, remote sensing target detection, robotics, etc. The primary objective of the object detection target is to identify and classify visual objects in images or videos. These objects are classified into specific object classes and highlighted by drawing bounding boxes [20]. Deep learning-based object detection methods can be classified into two categories: Two-One-Stage detectors and two Stage detectors. In the context of this research paper, the object detection methods employed are based on the YOLOv8 algorithm proposed in a previous research work [21].

YOLO (You Look Only Once) is a one-stage detector widely used algorithm famous for object detection robustness. Redmon et al. [22] introduced the first YOLO version in 2015. Then, the YOLO framework has gone through a series of advancements and updates, with each version introducing improved performance and capabilities in object detection. YOLO predicts the bounding box coordinates and class probabilities of objects in an image in a single pass. The main steps used by the YOLO algorithm to detect objects in an image are shown in Figure 1.



**Figure 1.** The main idea of the YOLO network for object detection [22]

In the first step, the algorithm splits the input image into an  $S \times S$  grid, predicts  $B$  bounding boxes and their confidence scores for each grid cell, and discards any bounding boxes with confidence scores below a specified threshold. In the final step, YOLO's regression algorithm estimates the bounding boxes for the entire image simultaneously. Ultimately, Non-maximum suppression (NMS) is applied to eliminate overlapping bounding boxes and obtain the final detection. The YOLO network has seen a series of significant

improvements from its original version, YOLOv1, to the latest iteration, YOLOv8. Each version of YOLO aimed to improve accuracy, speed, and the ability to handle various object scales and types. Table 2 describes the main improvements made from the initial YOLOv1 and beyond [23]:

**Table 2.** The main YOLO framework's improvement

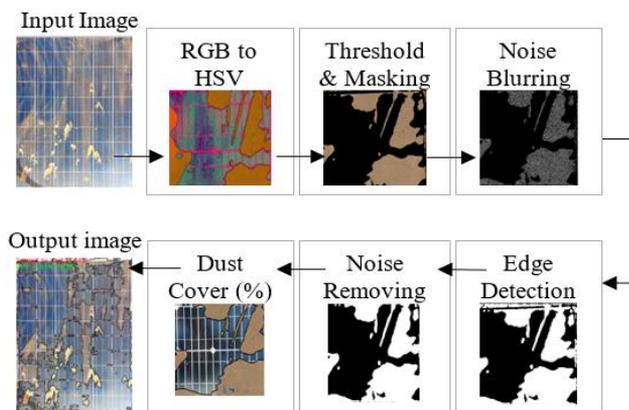
YOLO Framework's	Main Improvement
YOLOv1	Detection relies on grid division. Confidence loss is a primary focus.
YOLOv2	Introduced anchor boxes determined through K-means clustering. Adopted a two-stage training process. Transitioned to a full convolutional network architecture.
YOLOv3	Implemented multi-scale detection by integrating Feature Pyramid Networks (FPN). Incorporated Spatial Pyramid Pooling (SPP). Used the MISH activation function.
YOLOv4	Introduced data augmentation techniques like Mosaic and Mixup. Introduced the GIOU (Generalized Intersection over Union) loss function.
YOLOv5	Offered flexible control over model size. Employed the Hardswish activation function. Further improved data augmentation strategies. Opts for the anchor-free methods, unlike the previous version, which used anchor-based methods.
YOLOv6	Uses the EfficientRep backbone consisting of CSPStackRep blocks, RepConv, RepBlock. Used the Efficient Decoupled Head. Two loss functions are required: Varifocal Loss (VFL) for classification and Distribution Focal Loss (DFL) for box regression.
YOLOv7	We propose several architectural reforms and a series of enhancements, collectively known as a bag-of-freebies, to improve model performance. These reforms include the implementation of E-ELAN (Extended Efficient Layer Aggregation Network) and the adoption of model scaling specifically tailored for concatenation-based models. The bag of freebies utilized in this approach encompasses planned re-parameterized convolution, which optimizes the convolutional operations, and a dual approach to loss calculation, employing coarse adjustments for auxiliary loss and fine-tuning for the primary lead loss.
YOLOv8	C3 Module Replacement: Replace the C3 module with the C2f module. Convolutional Kernel Adjustment: In the backbone, the first 6x6 Conv layer is replaced with a 3x3 Convolutional layer. Convolution Deletion: Two Convolutional layers (No.10 and No.14 in the YOLOv5 configuration) are removed from YOLOv8. Bottleneck Convolution Change: The first 1x1 Convolutional layer in the Bottleneck is replaced with a 3x3 Convolutional layer. Decoupled Head and Objectness Branch Deletion: YOLOv8 adopts a decoupled head, separating certain network components and eliminating the objectness branch.

## 2.2 YOLOv8 model structure

In this paper, we present a novel approach using the latest



identification of dust accumulation. This method can be seamlessly integrated into automated inspection and maintenance systems, contributing to the efficient monitoring and cleaning of solar installations.



**Figure 4.** The block diagram of image processing techniques applied to the dusty panel

After image acquisition, a series of image processing techniques were applied to extract the dusty region of the PV panel image. Figure 4 presents an illustrative block diagram detailing the main image-processing methods for dust layer detection. Different techniques are involved in preprocessing: thresholding, binarization, edge detection, and morphological transformations to improve the image quality and get accurate dust layer extraction from the input image. In the first step, the RGB matrix of the input image was converted to HSV (Hue, Saturation, and Value) space by using the *cv.cvtColor()* function. Then, to identify the pixels within a desired range of values (pixels with dust) from the obtained HSV image, the *cv.inRange()* function was used to perform basic thresholding in the second step, this function takes three parameters to extract the dust layer we are interested in. To remove a false detection along the image border, *cv.GaussianBlur()* function was used.

Gaussian filtering has been intensively studied in image processing and computer vision to reduce noise and preserve edges [31]. Finally, the canny edge detection method based on OpenCV was used for dust edge detection. First, erosion followed by dilatation was performed to preserve the shape and size of the dust area in the image [32]. To calculate the percentage of the dust-covered regions corresponding to the number of non-zero pixels in the binary image Eq. (1), we used the function *cv.countNonZero()*. Total pixels are the number of white pixels (whose value is 255) and black pixels (whose value is 0) inside the binary image Eq. (2). The height and width of the image are calculated using *img.shape()* function.

By combining these pre-processing techniques, the overall aim is to improve the quality of the input image and make it more suitable for the subsequent analysis stages, eventually leading to more accurate dust layer extraction. Each technique plays a specific role in treating different aspects of the image, collectively contributing to the efficiency of the entire pre-processing process.

Based on the image processing techniques, we achieved satisfactory results by segmenting the regions of interest and isolating dust particles from the background of the input image. A decision is made considering the level of dust identified on the PV panels and the type nature of soiling. This decision

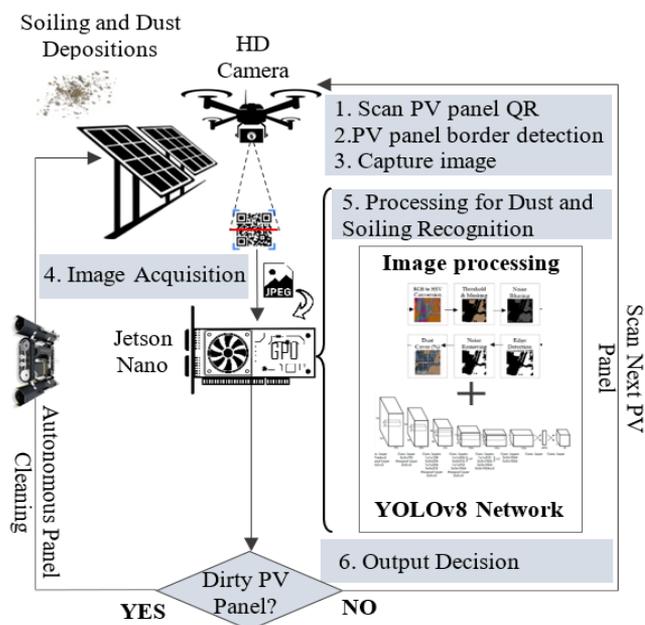
involves initiating the cleaning process for the soiled panels or capturing the next PV panel image for the new processing process in case of a clean panel. This decision is relayed to the autonomous cleaning system to provide a choice of cleaning operations; more details in future work. Different types of cleaning methods and techniques are proposed and inspired by cleaning methods summarized in Younis and Onsa [33] paper to remove or mitigate the soiling and dust from PV surfaces depending on the type and classification of the soiling.

$$Dust\ percentage\ (\%) = \frac{NonZero_{pixels}}{Total_{pixels}} \quad (1)$$

$$Total_{pixels} = Height * Width \quad (2)$$

## 2.4 The proposed system architecture

This section describes the proposed system architecture used in this paper for soiling and dust identification and detection. The proposed system consists of two main components: The software combines the YOLOv8 algorithm and computer vision techniques, efficiently detecting and analyzing soiling and dust on PV panels. The system's hardware components include an HD camera used for capturing images of the PV panels and the Nvidia Jetson Nano, which serves as the central processing unit to perform the core processing tasks, enabling rapid and real-time analysis of the captured images. Figure 5 shows the architecture of the proposed system for real-time soiling and dust detection.

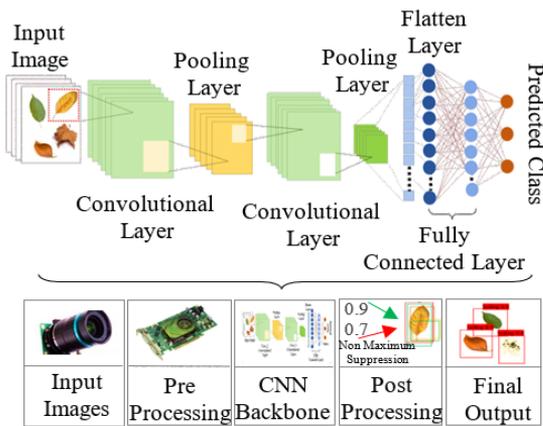


**Figure 5.** The architecture of the proposed system for soiling and dust detection on solar panels

In the first stage, an HD camera captures the PV panel image after scanning the QR code related to the PV panel and detecting its borders. After image acquisition, a PV panel state evaluation method based on a YOLOv8 Algorithm, then image processing techniques are performed for soiling and dust identification to make an optimal decision depending on the types of the detected soiling and the percentage of the dust layer. The YOLO algorithm consists of several steps, from the input image to the final output. Figure 6 shows the architecture of the proposed YOLO-CNN Algorithm for soiling detection

tasks. The input image containing the objects to be detected (a) is preprocessed for the neural network algorithm through the preprocessing step (b). Common preprocessing steps include resizing the image to a fixed size and normalizing values to a specific range. The first algorithm step divides the input image into an  $S \times S$  grid. The preprocessed image is passed through a Convolutional Neural Network (CNN) architecture as its backbone (c). The CNN is responsible for extracting features from the image. The CNN architecture typically consists of multiple convolutional layers followed by fully connected pooling layers, normalization, and activation functions [27]. These layers are designed to capture hierarchical features, including edges, textures, and object parts.

This design enables the network to predict three main components for each grid cell. Bounding Box Predictions, including the coordinates  $(x, y, w, h)$  of the bounding box's center. Class Predictions: for each bounding box, the network predicts the conditional class probabilities for all possible classes and Confidence Scores, each bounding box also has an associated confidence score, which represents how confident the network is that this bounding box contains an object. After obtaining predictions from the CNN, post-processing steps (d) are applied to refine and filter the predictions. Only the bounding box with the greatest Intersection Over Union (IOU) correlation to the ground truth is selected by the Non-Maximum Suppression (NMS) algorithm. The final output (e) of the YOLO algorithm is a set of bounding boxes, each associated with the class label and with the highest conditional probability. A confidence score that indicates how likely the predicted box contains an object. These bounding boxes represent the detected soiling in the input image.



**Figure 6.** The architecture of the YOLO-CNN algorithm for Soiling detection tasks

The application of the YOLOv8-based object detection framework yielded unsatisfactory results for dust detection. The primary reason for this discrepancy is the heterogeneous distribution of dust on various surfaces. The YOLOv8 algorithm is optimized for object detection tasks, assuming a relatively uniform distribution of objects within the image, which is not true for dust layer detection.

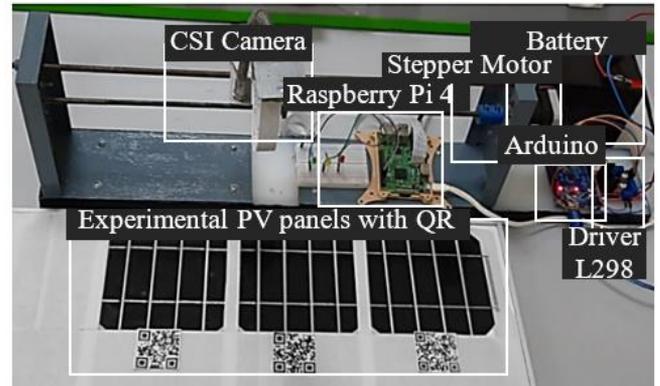
To overcome this issue, various image-processing techniques provided by OpenCV for dust layer detection were proposed in this paper.

### 2.5 Prototype system for testing

A prototype has been developed to assess and validate a

method for scanning QR codes and capturing images related to solar panels. Figure 7 illustrates a mechanical arm with a CSI camera programmed to move systematically for QR code scanning and image capture for each corresponding solar panel. A stepper motor controlled by a microcontroller facilitates the arm's movement. The Raspberry Pi 4 board has been used for image acquisition and transfer. The stepper motor ensures precision in positioning the camera. The microcontroller plays a central role in controlling the movements of the stepper motor, ensuring precise synchronization during QR code scanning and image capture.

The main purpose of this prototype is to validate the reliability of the proposed QR code scanning and image-capturing method, potentially leading to future developments and improvements based on testing outcomes.



**Figure 7.** The prototype for the experimental QR scan test

### 2.6 YOLOv8 training dataset

The main objective of the proposed system is to accurately classify various types of soiling and quantify the dust coverage on the PV panel surface. By training the YOLOv8 algorithms, the system aims to achieve the ability to not only detect the presence of soiling and dust but also classify them into different categories based on their types. This classification enables a more detailed analysis and understanding of the specific types of soiling affecting the PV panels. Training the YOLOv8 algorithms involves providing a labelled dataset that includes images of PV panels with different types of soiling and dust coverage.

The proposed YOLOv8 pipeline for soiling detection involves two stages for final soiling recognition, as shown in Figure 8. In the initial data collection phase, the focus is on collecting a vast data set, including videos and images covering several types of soiling that affect the efficiency and performance of solar PVs (Bird dropping, Tree branches, tree-dropping leaves, etc.). Rigorous attention must be given to dataset curation, which includes eliminating duplicates and applying appropriate labels denoting the name of each soiling. These labels can be generated through automated labeling tools or manual annotation. In the second data preparation stage, the dataset containing soiling types of images and videos collected in the previous step is prepared for training and testing our soiling detection system. This step involves two phases: Data labeling involves labeling the images and videos by surrounding bounding boxes for each type of soiling. It can be accomplished manually and expedited through labeling software. Roboflow software annotated and classified images with multiclass annotation [34]. Roboflow provides

tools and services for managing and annotating datasets for machine learning [35].

During the data splitting step, labelled data are partitioned into training and testing sets. This division is a crucial strategy in the machine learning workflow for several reasons. The testing set, separate from the training data, serves to evaluate the model's generalization ability. It comprises examples that the model hasn't encountered during training, providing a reliable measure of its performance on new and unseen data. The ultimate goal is to ensure that the model can effectively apply its learned knowledge to real-world scenarios. With this balanced approach, we can rigorously train and evaluate our YOLOv8-based model, enhancing its reliability and effectiveness in detecting a wider range of dirt types.

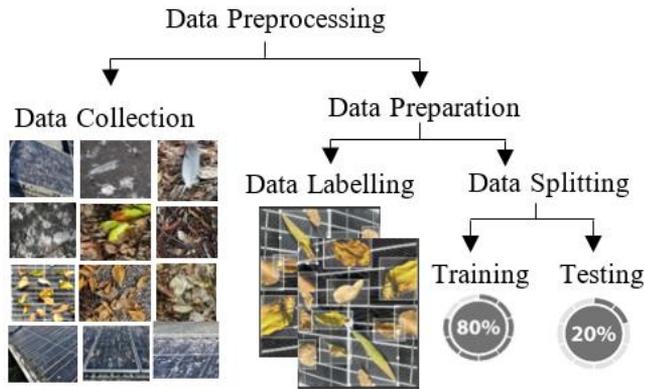


Figure 8. Data Preprocessing steps for soiling detection

In this study, we constructed a training model by curating and annotating a comprehensive database comprising over 3000 images, each sized at (1088×1920) pixels. These images were captured using a high-resolution camera and encompassed various types of soiling.

To ensure effective model training, we partitioned the dataset into 80% for training and retained the remaining 20% for test validation purposes. The database was meticulously annotated using an image-labeling tool for object detection, thus creating a suitable dataset for training the model.

The annotations were formatted in YOLO format, and both the training model and annotated images were uploaded to the Google Drive platform for accessibility and sharing. For training, we utilized Google Colab, an Integrated

Development Environment (IDE) known for its support of machine learning and deep learning tasks. Google Colab provides access to Graphics Processing Units (GPUs) and Tensor Processing Units (TPUs), crucial for accelerating model training. Additionally, it comes pre-installed with popular libraries such as PyTorch, TensorFlow, Keras, and OpenCV, commonly used in deep learning research. This setup facilitated efficient model development and experimentation. The photovoltaic system used for the image collection of the training dataset is a standalone of four panels made of a poly-crystalline silicon model. Figure 9 depicts the PV system installed on the roof of the physics department building at Ibn Tofail University.



Figure 9. Field experiment of PV panel images for soiling and dust accumulation detection

### 2.7 Model performance comparison

To evaluate the performance of our model based on the enhanced YOLOv8 version, we conducted comparative experiments with two widely used object detection models. These models include the previous version YOLOv5 (Ref) as well as the one-stage anchor-based algorithm, MobileNetV2-SSD. The experiments were conducted under identical experimental conditions and on similar datasets. We selected several PV panel images taken by an onboard camera for soiling detection and recognition.

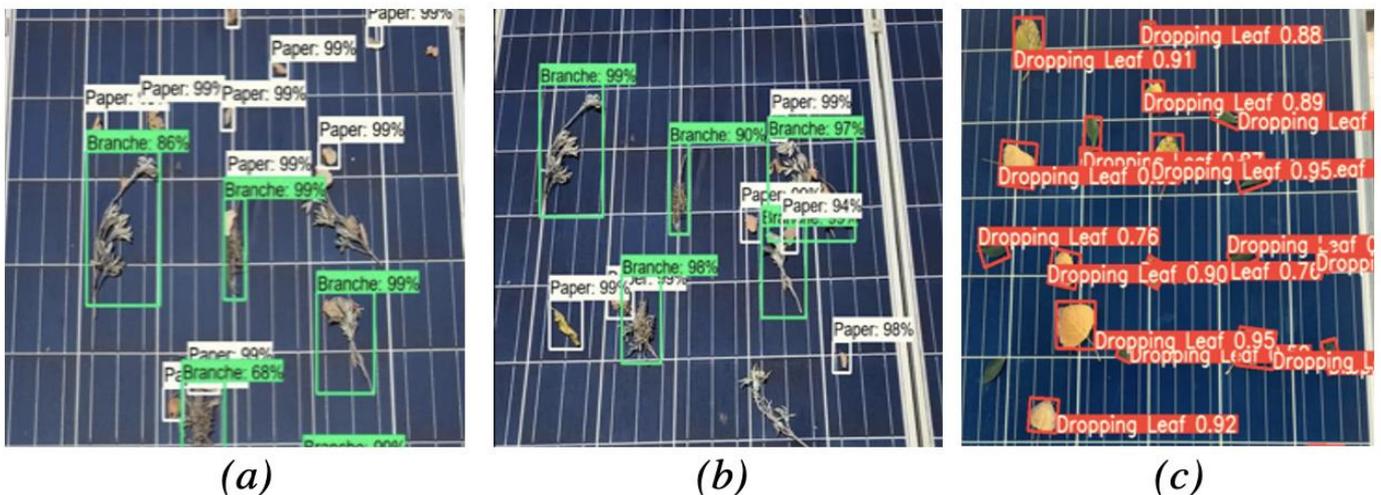


Figure 10. Comparative experiment results for MobileNetV2 SSD (a), (b) and YOLOv5 (c)

In the experiment, our YOLOv8 model outperformed the other models in terms of detection speed and accuracy. Our model exhibited higher detection accuracy than the others, with no false detections and no missed detections. In contrast, the SSD model encountered issues with false detections. While the YOLOv5 model successfully detected almost all dirt, it also had minimal missed detection and reduced false detection rates. The detection results of the different models in the experiment can be observed in Figure 10. Our model produced more accurate bounding boxes and effectively identified all targets in the images captured.

Our model YOLOv8 achieved a Map@50 accuracy of 90.9% and an F1 score of 0.88, significantly outperforming other object detection models. These results show the improved performance of our model when compared to YOLOv8 and other object detection models, demonstrating its ability to substantially improve dirt detection in various scenarios. Additionally, our model exhibits a significant reduction in parameter overhead and computation time compared to the original YOLOv8, making it more efficient and practical for use in Real-world applications.

### 2.8 Processing system

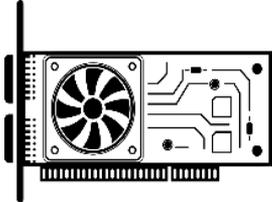
The YOLOv8 and image processing algorithms are implemented in Jetson Nano with a GPU 128 Cuda Cores Cortex-A57 64-bit. Implementing YOLOv8 and image processing algorithms on Jetson Nano with a GPU provides sufficient computational power and memory to operate these algorithms. Table 3 shows the technical specifications of the processing systems that were used for implementation.

The significance of selecting Jetson Nano as the hardware platform for implementing YOLOv8 and image processing algorithms for real-time and low-power application scenarios can be further understood through various factors. Jetson Nano's GPU with 128 CUDA cores provides significant parallel processing capabilities, crucial for accelerating deep learning tasks like object detection with YOLOv8, enabling real-time processing of high-definition images. Additionally, its low-power design makes it suitable for energy-efficient operations in resource-constrained environments. NVIDIA's optimized deep learning frameworks like TensorFlow, PyTorch, and TensorRT take advantage of Jetson Nano's GPU, maximizing performance while minimizing power consumption. Its compact form factor allows seamless integration into space-limited environments, making it ideal for onboard vehicles or small monitoring devices. Moreover, Jetson Nano's cost-effectiveness compared to higher-end platforms, coupled with its vibrant developer community and rich ecosystem of software tools, documentation, and tutorials, ensures simplified development and troubleshooting, making it a compelling choice for real-time and low-power applications requiring sophisticated image processing and deep learning algorithms like YOLOv8.

The NVIDIA Jetson Nano device used as the main processing unit in the system is a low-power embedded system specifically designed for running deep learning models and performing real-time detection tasks. It provides sufficient computational power to process the images captured by the HD camera and run the YOLOv8 Deep Learning Network and computer vision algorithms for soiling and dust detection [17, 18]. However, it is important to note that the system's performance may vary depending on the complexity of the algorithms and the size of the images being processed. Even

with Jetson Nano's capabilities, two critical parameters must be considered for building a high-performance power system: energy consumption and processing time.

**Table 3.** Technical specifications of NVIDIA Jetson Nano used for implementing

NVIDIA® Jetson Nano™	Processing Systems Features	
	CPU	64-bit Quad-core ARM A57 @ 1.43GHz
	GPU	128- CUDA cores NVIDIA Maxwell @ 921MHz
	Memory	4GB 64-bit LPDDR4 @ 1600MHz   25.6 GB/s

## 3. RESULTS AND DISCUSSION

### 3.1 Performance analysis of YOLOv8 on custom dataset

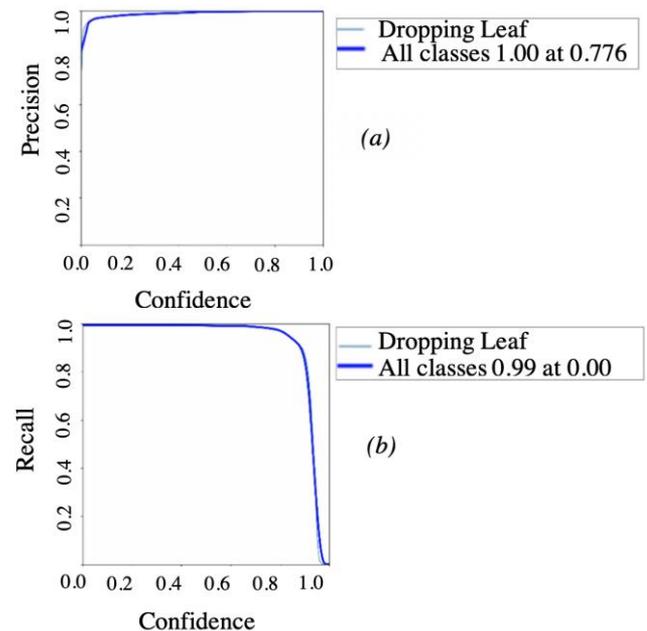
The selection of appropriate evaluation metrics is crucial for measuring a model's performance relative to the problem under investigation. These metrics, influenced by the dataset characteristics and domain, have been carefully chosen to provide a comprehensive view of the model's strengths and limitations.

Precision denoting the positive predictive values. The Recall, also known as sensitivity, signifies the ratio of true positives in the predictions. Eq. (3) and Eq. (4) shows the formulas used to calculate these performance metrics [36].

$$\text{Precision} = \frac{TP}{TP+FP}; \tag{3}$$

$$\text{Recall} = \frac{TP}{TP+FN}; \tag{4}$$

where, TP=true positives, TN=true negatives, FP=false positives and FN=false negatives.



**Figure 11.** (a) Precision curve (P-Curve); (b) Recall curve (R-Curve) of our trained model

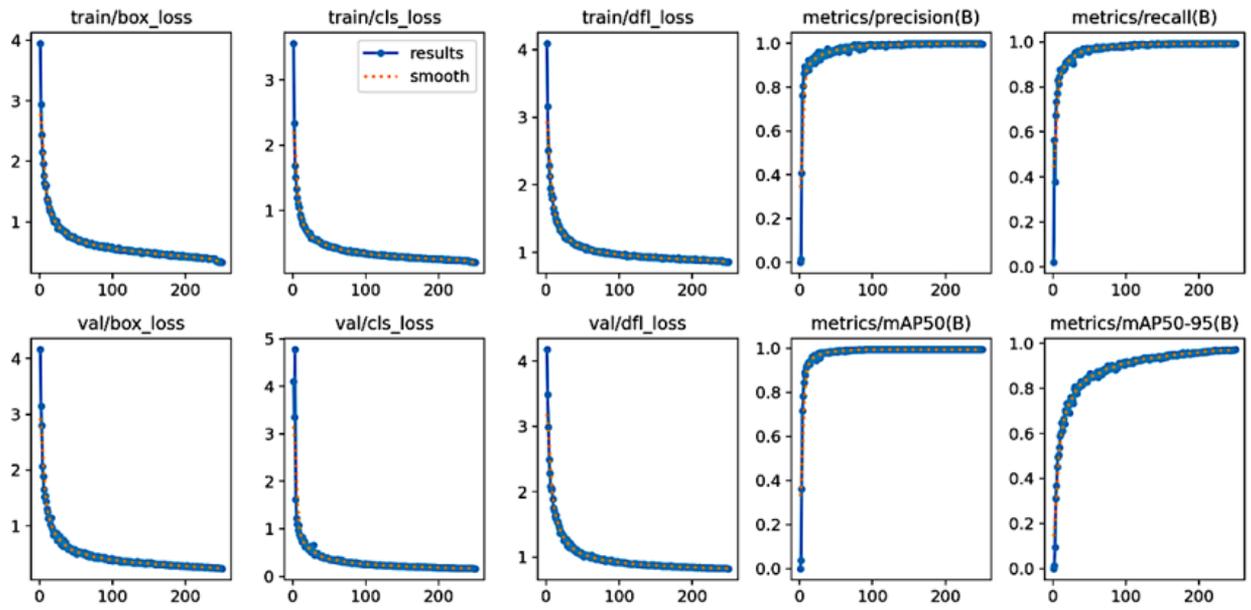
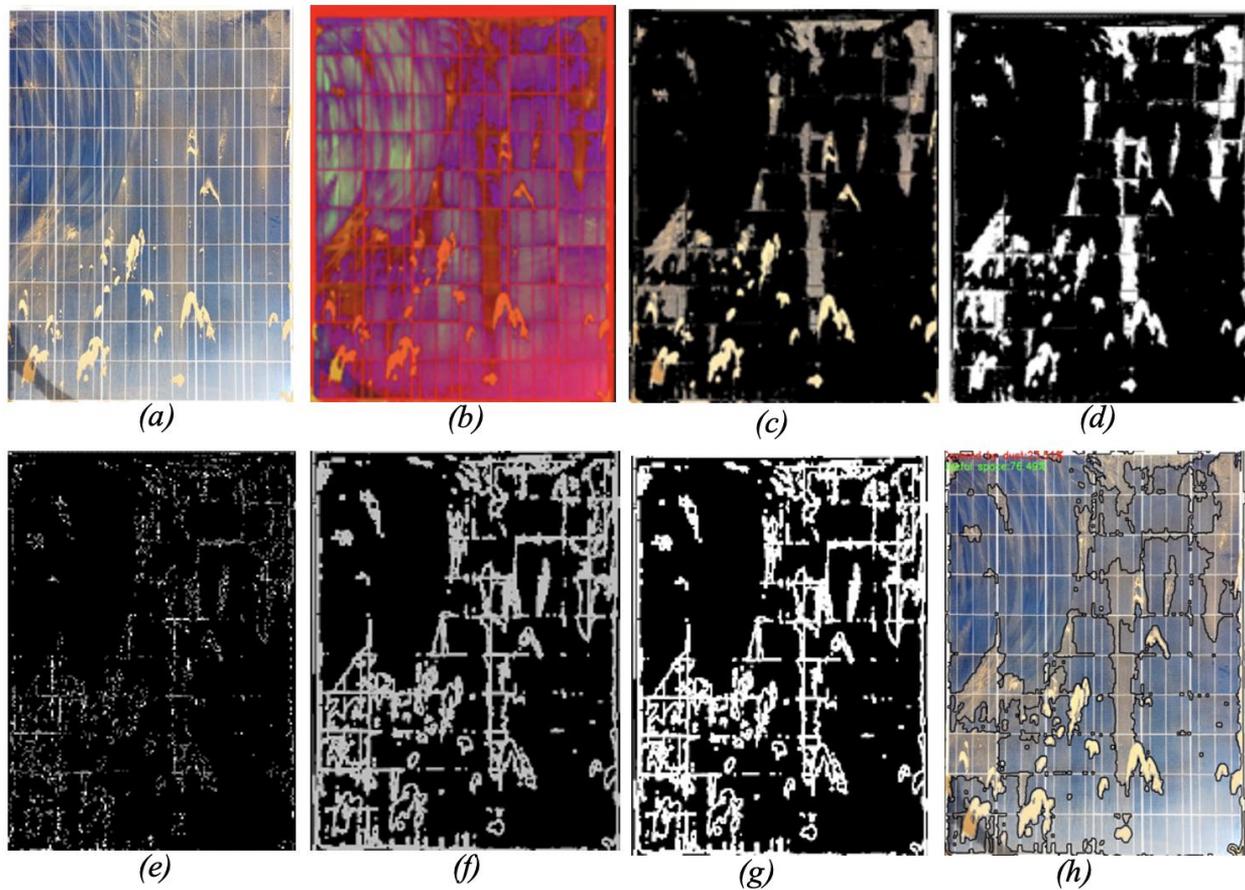


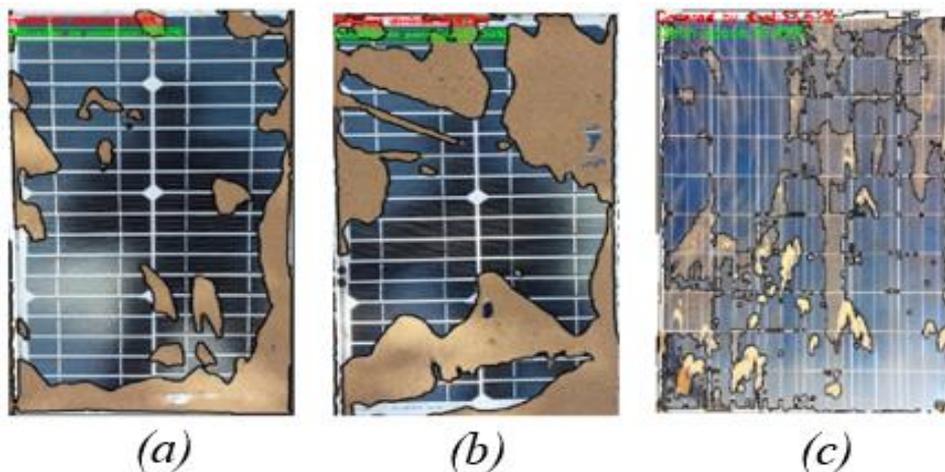
Figure 12. The loss function and metrics (300 epochs) of YOLOv8 model



Figure 13. YOLOv8 soiling detection results



**Figure 14.** Examples of dust image detection for two PV panels,  $PV_1$  and  $PV_2$ : (a) input image, (b) HSV image, (c) Thresholded image for a range of dust colors, (d) Grayscale image, (e) Dust edge boundary image (f)(g) Eroding and Dilating image (h) % covered dust area



**Figure 15.** The percentage calculation result of dust accumulation and dust cover % diagram of three PV panels

Figure 11 presents the performance achieved by YOLOv8 when trained on our custom dataset. The P-curve and R-curve independently portray the evolution of precision and recall, across varying confidence thresholds.

Our model, trained to identify tree leaves in photovoltaic (PV) images, performs remarkably well, as shown by an analysis of its precision-recall curve (P-Curve).

Precision, representing the accuracy of tree leaves identified, is exceptionally high, minimizing false positives and guaranteeing reliable classification of foliage regions.

The training presented in Figure 12 appears promising, with

the training and validation losses decreasing steadily over time. The box loss, which measures the localization error of the bounding boxes, shows a clear downward trend for both training and validation data, reaching around 4.0 for training and 5.0 for validation at epoch 150. Similarly, the classification loss, which measures the error in predicting object classes, exhibits a consistent decline, stabilizing around 3.0 for training and 4.0 for validation. These trends suggest that the model is effectively learning to localize and classify objects in the data.

### 3.2 YOLOv8 performance and dust extraction results

#### 3.2.1 Soiling detection with YOLOv8 algorithm

To assess the proposed method results for soiling detection using a YOLOv8 framework. Soiling was manually simulated by placing falling tree leaves on different photovoltaic panels. Figure 13 displays images of successful detection results performed by YOLOv8 networks of five experimental PV panels. Each detected soiling on the PV panel is surrounded by a Bounding Box.

The results demonstrated that the YOLOv8 algorithm successfully identified over ten classes of tree leaves on the photovoltaic panel images.

The accuracy of the YOLO algorithm for soiling detection demonstrated its efficiency, achieving a global rate exceeding 90%. That means that in 90% of the examined cases, the algorithm accurately detected soiling on the photovoltaic panels. Expanding the capabilities of our model to detect a wider range of soiling types that could potentially decrease the PV installation performance, we aim to improve our training dataset by incorporating more diverse examples of soiling scenarios. By collecting and annotating additional data that includes different forms of soiling, such as dust, bird droppings, and other environmental contaminants, we aim to expose the model to a broader spectrum of challenges.

#### 3.2.2 Dust layer detection

Figure 14 shows the successful application of image processing techniques to extract dust accumulation from the experimental PV panels covered with dust. Using the OpenCV-Python library, a series of image-processing techniques are employed to extract the dust from a preprocessed image.

Both Eq. (1) and Eq. (2) are used to calculate the percentage of the panel area covered by dust, and the resulting calculation is presented in Figure 15. The analysis reveals that approximately Twenty-five percent of the solar panel surface is affected by dust, causing a reduction in energy production.

Based on the calculated percentage of dust coverage, it is possible to estimate the impact on panel efficiency and yield. Figure 14 illustrates that dust covers more than 20% of the panel surface, negatively affecting the electrical performance and resulting in a significant decrease in voltage and current output compared to clean panels. The proposed system automates the quantification of dust on PV panels and facilitates rapid decision-making to perform intelligent maintenance to achieve optimal power production.

The new proposed embedded system provides a decision based on the obtained results to make informed decisions and program a cleaning process according to the specific types of detected soiling. Future work will focus on exploring the implementation of the proposed system for autonomous Photovoltaic cleaning robots. The robot employs various algorithms to analyze the type of soiling and extract relevant information, such as the severity of the soiling, the distribution of the soiling on the surface, and any other pertinent features to select and perform the suitable cleaning method.

### 3.3 Implementation results

To choose a compatible system for processing to implement our approach, there are two main parameters to consider the system consumption energy and processing average time. Figure 16, shows the implementation of the proposed

algorithm in the NVidia Jetson Nano platform, which has adequate performance for implementing our method. The Jetson Nano system shows relatively low energy consumption compared to other processing systems.

This lower energy requirement makes it an interesting option for building an optimized power system with enhanced energy efficiency. Additionally, the Jetson Nano system offers a low average processing time, indicating that it can efficiently process the required algorithms and image processing tasks within a reasonable timeframe. This aspect is crucial for maintaining system response and ensuring real-time performance. Table 4 shows the results and technical specifications of the Jetson processing systems.



Figure 16. NVidia Jetson Nano processing system

Table 4. Results of the used processing system

Processing System	Jetson Nano
Energy Consumption	≈ 5W
Average Processing Time / 1Frame	3.74 ms
Price	250\$
Frame Per Second FPS	32

## 4. CONCLUSION

The popularity of Deep Learning Networks and computer vision is evident across various application domains, owing to their superior accuracy compared to traditional algorithms. Their adaptability has led to wide-spread adoption in a variety of fields.

This paper proposes a real-time embedded system for soiling recognition and dust detection in photovoltaic modules. We proposed a YOLOv8 framework to detect soiling on photovoltaic PV panels and perform operations on visible images and image processing techniques using the computer vision library OpenCV to extract the dust layer on the PV panel for the state PV panel evaluation. The results obtained in this work are favorable and can be used for further research to develop and conceive a robust Real-time system for photovoltaic module inspection. Furthermore, it can significantly improve the accuracy and efficiency of real-time PV panel inspection conditions. The YOLOv8 model was evaluated on more than 20 solar panels, showing promising accuracy with an overall detection rate of 85% for soiling.

In future research, many improvement ideas can be discussed. Increasing the image training dataset was expanded to incorporate additional types of soiling and defects, aiming

to enhance the detection and classification system's accuracy and efficiency. To identify a promising platform with high performance for embedded machine-learning systems. A comparison of the processing system used based on Jetson Nano to similar platforms such as Raspberry Pi 4, FPGA (Field-Programmable Gate Array), and VPU (Vision Processing Unit) is suggested. This comparison will help to evaluate the advantages and limitations of each platform. It can make an optimal decision on which platform will be the best suited for our embedded machine-learning requirements in terms of speed, power consumption, and average processing time.

## ACKNOWLEDGMENT

This research was conducted at the Laboratory of Advanced Systems Engineering, National School of Applied Sciences, University Ibn Tofail, Morocco. It was carried out in collaboration with the Laboratory of Electronic Systems, Information Processing, Mechanics, and Energetics at the Faculty of Sciences for method testing and validation.

## REFERENCES

- [1] Hamid, R.H.A., Elidrissi, Y., Elsamahy, A., Regradui, M., Menoufi, K. (2021). Examining the impact of different technical and environmental parameters on the performance of photovoltaic modules. *Environmental and Climate Technologies*, 25(1): 1-11. <https://doi.org/10.2478/rtuect-2021-0001>
- [2] Semaoui, S., Abdeladim, K., Taghezouit, B., Arab, A.H., Razagui, A., Bacha, S. (2020). Experimental investigation of soiling impact on grid-connected PV power. *Energy Reports*, 6: 302-308. <https://doi.org/10.1016/j.egyr.2019.08.060>
- [3] Quansah, D.A., Adaramola, M.S. (2019). Assessment of early degradation and performance loss in five co-located solar photovoltaic module technologies installed in Ghana using performance ratio time-series regression. *Renewable Energy*, 131: 900-910. <https://doi.org/10.1016/j.renene.2018.07.117>
- [4] Zeedan, A., Barakeh, A., Al-fakhroo, K., Touati, F., Gonzales, A.S.P. (2021). Quantification of PV power and economic losses due to soiling in Qatar. *Sustainability*, 13(6): 3364. <https://doi.org/10.3390/su13063364>
- [5] Mustafa, R.J., Gomaa, M.R., Al-dhaifallah, M., Rezk, H. (2020). Environmental impacts on the performance of solar photovoltaic systems. *Sustainability*, 12(2): 1-17. <https://doi.org/10.3390/su12020608>
- [6] Salimi, H., Mirabdollah Lavasani, A., Ahmadi-Danesh-Ashtiani, H., Fazaeli, R. (2019). Effect of dust concentration, wind speed, and relative humidity on the performance of photovoltaic panels in Tehran. *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, 45(3): 7867-7877. <https://doi.org/10.1080/15567036.2019.1677811>
- [7] Abdelsalam, M.A.M., Ahmad, F.F., Hamid, A.K., Ghenai, C., Rejeb, O., Alchadirchy, M., Obaid, W., El Haj Assad, M. (2021). Experimental study of the impact of dust on azimuth tracking solar PV in Sharjah. *International Journal of Electrical and Computer Engineering (IJECE)*, 11(5): 3671-3681. <https://doi.org/10.11591/ijece.v11i5.pp3671-3681>
- [8] Köntges, M., Kurtz, S., Packard, C.E., Jahn, U., Berger, K.A., Kato, K., Friesen, G. (2014). Review of Failures of Photovoltaic Modules. Technical Report <https://repository.supsi.ch/9645/>
- [9] Kawamoto, H., Shibata, T. (2015). Electrostatic cleaning system for removal of sand from solar panels. *Journal of Electrostatics*, 73: 65-70. <https://doi.org/10.1016/j.elstat.2014.10.011>
- [10] Kawamoto, H., Guo, B. (2018). Improvement of an electrostatic cleaning system for removal of dust from solar panels. *Journal of Electrostatics*, 91: 28-33. <https://doi.org/10.1016/j.elstat.2017.12.002>
- [11] Simiyu, D.S.N. (2020). Optimal Cleaning Strategy for Large Scale Solar Photovoltaic Array Considering Non-Uniform Dust Deposition. University of Pretoria.
- [12] Al-badra, M.Z., Abd-elhady, M.S., Kandil, H.A. (2020). A novel technique for cleaning PV panels using antistatic coating with a mechanical vibrator. *Energy Reports*, 6: 1633-1637. <https://doi.org/10.1016/j.egyr.2020.06.020>
- [13] Ul Mehmood, M., Ulasyar, A., Ali, W., Zeb, K., Zad, H.S., Uddin, W., Kim, H.J. (2023). A new cloud-based IoT solution for soiling ratio measurement of PV systems using artificial neural network. *Energies*, 16(2): 996. <https://doi.org/10.3390/en16020996>
- [14] Cavieres, R., Barraza, R., Estay, D., Bilbao, J., Valdivia-Lefort, P. (2022). Automatic soiling and partial shading assessment on PV modules through RGB images analysis. *Applied Energy*, 306: 117964. <https://doi.org/10.1016/j.apenergy.2021.117964>
- [15] Tan, Y., Liao, K., Bai, X., Deng, C., Zhao, Z., Zhao, B. (2019). Denoising convolutional neural networks based dust accumulation status evaluation of photovoltaic panel. In 2019 IEEE International Conference on Energy Internet (ICEI), Nanjing, China, pp. 560-566. <https://doi.org/10.1109/ICEI.2019.00105>
- [16] Li, B., Delpha, C., Diallo, D., Migan-Dubois, A. (2021). Application of artificial neural networks to photovoltaic fault detection and diagnosis: A review. *Renewable and Sustainable Energy Reviews*, 138: 170-179. <https://doi.org/10.1016/j.rser.2020.110512>
- [17] Elmanaa, I., Sabri, M.A., Abouch, Y., Aarab, A. (2023). Efficient roundabout supervision: Real-time vehicle detection and tracking on Nvidia Jetson Nano. *Applied Sciences*, 13(13): 7416. <https://doi.org/10.3390/app1313741>
- [18] Zhang, Z.D., Tan, M.L., Lan, Z.C., Liu, H.C., Pei, L., Yu, W.X. (2022). CDNet: A real-time and robust crosswalk detection network on Jetson nano based on YOLOv5. *Neural Computing and Applications*, 34(13): 10719-10730. <https://doi.org/10.1007/s00521-022-07007-9>
- [19] Dsa, J., Abhishek, S., Rao, D., Balaji, V., Bangera, V.V. (2023). Jetson nano-based low-cost safety enhancement system for the drivers on the road. *Preprints*, 2023: 040880. <https://doi.org/10.20944/preprints202304.0880.v1>
- [20] More, B., Bhosale, S. (2023). A comprehensive survey on object detection using deep learning. *Revue d'Intelligence Artificielle*, 37(2): 407. <https://doi.org/10.18280/ria.370201>
- [21] Hajar, E.K., Rachid, E., Mohamed, B., Youssef, A., Abdelkader, M. (2023). Application of deep learning based detector YOLOv5 for soiling recognition in photovoltaic modules. In 2023 3rd International

- Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), Mohammedia, Morocco, pp. 1-6. <https://doi.org/10.1109/IRASET57153.2023.10152990>
- [22] Redmon, J. (2016). You Only Look Once: Unified, Real-Time Object Detection. arXiv:1506.02640. <https://doi.org/10.48550/arXiv.1506.02640>
- [23] Hussain, M. (2023). YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection. *Machines*, 11(7): 677. <https://doi.org/10.3390/machines11070677>
- [24] Jocher, G., Chaurasia, A., Qiu, J. (2023). YOLO by Ultralytics. GitHub. Available online: <https://github.com/ultralytics/ultralytics>.
- [25] Terven, J., Córdova-Esparza, D.M., Romero-González, J.A. (2023). A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. *Machine Learning and Knowledge Extraction*, 5(4): 1680-1716. <https://doi.org/10.3390/make5040083>
- [26] Jiang, P., Ergu, D., Liu, F., Cai, Y., Ma, B. (2022). A review of YOLO algorithm developments. *Procedia Computer Science*, 199: 1066-1073. <https://doi.org/10.1016/j.procs.2022.01.135>
- [27] Li, C.Y., Li, L.L., Jiang, H.L., Weng, K.H., Geng, Y.F., Li, L., Ke, Z.D., Li, Q.Y., Cheng, M., Nie, W.Q., Li, Y.D., Zhang, B., Liang, Y.F., Zhou, L.Y., Xu, X.M., Chu, X.X., Wei, X.M., Wei, X.L. (2022). YOLOv6: A single-stage object detection framework for industrial applications. arXiv preprint arXiv:2209.02976. <https://doi.org/10.48550/arXiv.2209.02976>
- [28] Ultralytics. (2023). Brief summary of YOLOv8 model structure. GitHub. <https://github.com/ultralytics/ultralytics/issues/189>.
- [29] Abukhait, J. (2024). Dust detection on solar panels: A computer vision approach. *Ingénierie des Systèmes d'Information*, 29(2): 533-541. <https://doi.org/10.18280/isi.290214>
- [30] Benallal, M.A., Si Tayeb, M. (2023). An image-based convolutional neural network system for road defects detection. *IAES International Journal of Artificial Intelligence (IJ-AI)*, 12(2): 577-584. <https://doi.org/10.11591/ijai.v12.i2.pp577-584>
- [31] Xu, Z., Baojie, X., Guoxin, W. (2017). Canny edge detection based on Open CV. In 2017 13th IEEE international conference on electronic measurement & instruments (ICEMI), Yangzhou, China, pp. 53-56. <https://doi.org/10.1109/ICEMI.2017.8265710>
- [32] Singh, H. (2019). Practical machine learning and image processing: For facial recognition, object detection, and pattern recognition using Python. Apress Berkeley, CA. <https://doi.org/10.1007/978-1-4842-4149-3>
- [33] Younis, A., Onsa, M. (2022). A brief summary of cleaning operations and their effect on the photovoltaic performance in Africa and the Middle East. *Energy Reports*, 8: 2334-2347. <https://doi.org/10.1016/j.egy.2022.01.155>
- [34] Dinesh, Y., Ramalingam, K., Ramani, P., Deepak, R.M. (2023). Machine learning in the detection of oral lesions with clinical intraoral images. *Cureus*, 15(8): e44018. <https://doi.org/10.7759/cureus.44018>
- [35] Brucal, S.G.E., de Jesus, L.C.M., Peruda, S.R., Samaniego, L.A., Yong, E.D. (2023). Development of tomato leaf disease detection using YOLOv8 model via RoboFlow 2.0. In 2023 IEEE 12th Global Conference on Consumer Electronics (GCCE), Nara, Japan, pp. 692-694. <https://doi.org/10.1109/GCCE59613.2023.10315251>
- [36] Alhuseen, Z.A.A., Joda, F.A., Naser, M.A. (2023). Abnormal behavior detection in gait analysis using convolutional neural networks. *Ingénierie des Systèmes d'Information*, 28(5): 1151-1159. <https://doi.org/10.18280/isi.280504>