



An interpretable Bayesian deep learning-based approach for sustainable clean energy

Dalia Ezzat^{1,2,5} · Eman Ahmed^{1,5} · Mona Soliman^{1,5} · Aboul Ella Hassanien^{3,4,5}

Received: 19 August 2023 / Accepted: 9 May 2024 / Published online: 6 June 2024
© The Author(s) 2024

Abstract

Sustainable Development Goal 7 is dedicated to ensuring access to clean and affordable energy that can be utilized in various applications. Solar panels (SP) are utilized to convert sunlight into electricity, acting as a renewable energy source. It is important to keep SP clean to obtain the required performance, as the accumulation of snow and dust on SP greatly affects the amount of electricity generated. On the other hand, excessive cleaning has some detrimental effects on the SP, therefore cleaning should only be done when necessary and not on a regular basis. Consequently, it is critical to determine whether the cleaning procedure is necessary by automatically detecting the presence of dust or snow on the panels while avoiding inaccurate predictions. Research efforts have been made to detect the presence of dust and snow on SP, but most of the proposed methods do not guarantee accurate detection results. This paper proposes an accurate, reliable, and interpretable approach called Solar-OBNet. The proposed Solar-OBNet can detect dusty SP and snow-covered SP very efficiently and be used in conjunction with the methods used to clean SP. The proposed Solar-OBNet is based on a Bayesian convolutional neural network, which enables it to express the amount of confidence in its predictions. Two measurements are used to estimate the uncertainty in the outcomes of the proposed Solar-OBNet, namely predictive entropy and standard deviation. The proposed Solar-OBNet can express confidence in the correct predictions by showing low values for predictive entropy and standard deviation. The proposed Solar-OBNet can also give an uncertainty warning in the case of erroneous predictions by showing high values of predictive entropy and standard deviation. The proposed Solar-OBNet's efficacy was verified by interpreting its results using a method called Weighted Gradient-Directed Class Activation Mapping (Grad-CAM). The proposed Solar-OBNet has achieved a balanced accuracy of 94.07% and an average specificity 95.83%, outperforming other comparable methods.

Keywords Sustainable development goals · Sustainable energy · Solar panels · Bayesian deep learning · Harris Hawks optimizer · Interpretable deep learning

✉ Dalia Ezzat
dalia_ali@cic-cairo.com

Aboul Ella Hassanien
Abo.aly@ku.edu.kw; Aboitcairo@cu.edu.eg

¹ Faculty of Computers and Artificial Intelligence, Cairo University, Cairo, Egypt

² School of Computer Science, Canadian International College (CIC), Cairo, Egypt

³ College of Business Administration (CBA), Kuwait University, Kuwait City, Kuwait

⁴ Faculty of Computers and Artificial Intelligence (IT Department), Cairo University, Cairo, Egypt

⁵ Scientific Research School of Egypt (SRSEG), Cairo, Egypt

1 Introduction

The United Nations adopted the 2030 Agenda to ensure sustainable development, which includes 17 goals and 169 targets. Goal 7 is focused on producing and dealing with clean and affordable energy. Due to population expansion and industrial development, the demand for electrical energy has significantly expanded. Long ago, traditional energy sources including coal, natural gas, and crude oil were regarded as the primary sources of electricity [1, 2]. However, the use of these traditional energy sources has had a severe impact on the environment, which has significantly increased global warming [3]. Accordingly, the reliance on renewable energy to satisfy global energy

demands is growing. In 2016, 24% of all electricity was produced by renewable energy sources, which have been outpacing natural gas since 2013 [4, 5]. Examples of these sources include solar, wind, hydro, and biomass. Solar energy is transformed into electrical energy utilizing SP [6]. Environmental factors, including operating temperature, sky condition, snow, leaves, and dust on the PS affect its performance and efficiency. Dust particles on SP and those in the atmosphere can reflect, scatter, and absorb sunlight. As a result of dust particles, less light may be available for photovoltaic (PV) cells to absorb and convert into photocurrents. One of the main sources of energy loss for SP, particularly in dry desert environments, is dust buildup or soiling on SP [7]. Similarly, accumulating snow affects the performance of the panels by reducing the quantity of solar irradiation that reaches the PV cells, leading to significantly reduced or nonexistent energy production [4].

Numerous studies looking at the impact of deposited dust on SP found that soiling rates are highest in tropical climates, particularly in Asia. Between 20 and 25% of the power generated by PV cells could be lost during the process, based on an Indian study on the dust effect [8]. A study in Saudi Arabia revealed that the yield of SP could drop by 26–40% due to dusty deposition [9], and other studies have shown that dust deposition on SP has a significant impact on their efficiency [10–17]. On the other hand, the amount of electricity lost owing to snow cover has been estimated to reach up to 34% of annual generation [18] but is usually less than 10% [19–23]. However, due to the accumulation of snow on SP during the winter, 90–100% of planned generation may be lost, as has been proven in many studies [23–27].

There are numerous methods that are utilized to clean SP periodically, including manual, automatic and coating methods. However, each method has some drawbacks, for example, the manual cleaning method is slow, labor-intensive, and has risks to human life. The most important disadvantage of the automatic cleaning method is that its maintenance costs are high, while the most important disadvantage of the coating method is that it may cause a decrease in electrical efficiency [13, 28]. Hence, detecting when it is necessary to undergo the cleaning process automatically rather than cleaning frequently is essential to mitigate the burden of redundant cleaning processes. There is a need to rely on accurate methods for detecting dusty SP and snow-covered SP because any error will result in a significant loss of energy.

Some of the research efforts have been directed to take advantage of the significant advances made by artificial intelligence (AI) methods, especially deep learning (DL) methods in various fields, in the automatic detection of dusty SP and snow-covered SP. DL methods have already

achieved effective results in identifying dusty SP and snow-covered SP [6, 29]. Despite the impressive successes made by DL methods, these methods do not ensure accurate detection outcomes unless the test data roughly matches the distribution of the training data. Unfortunately, practical detection applications where test data frequently diverges from training data make it challenging to satisfy this condition. This issue is referred to as the out-of-distribution (OOD) issue [30]. When working on test data, traditional DL models have a propensity to be overconfident when they make incorrect predictions without providing any kind of warning of uncertainty. The inability of these models to provide a quantitative estimate of the predictive uncertainty in their results reveals a lack of theoretical comprehension of the underlying mechanisms of DL models. Hence, they are often called “black boxes” [31]. The predictive uncertainty is the sum of the uncertainty in the DL model’s parameters (epistemic uncertainty) and the uncertainty in the data (aleatoric uncertainty) [32]. The probabilistic version of classical DL models, known as Bayesian DL models, can offer estimates of uncertainty because they offer a posterior distribution across the outputs for a given test sample [33]. Monte Carlo dropout (MCD) is an effective technique to efficiently transform any traditional DL model into a Bayesian variant without increasing the computational time [34]. MCD technique can be applied to any DL model such as convolutional neural network (CNN) by adding a dropout layer with an appropriate dropout rate, training the model, and then activating the dropout layer during inference as well. By activating the dropout during inference and making multiple forward passes for each test sample, the posterior distribution is obtained. It has been observed that the chosen dropout rate significantly influences the obtained posterior distribution, as choosing a high dropout rate results in obtaining a highly diverse posterior distribution. Choosing a low rate will result in very similar posterior distribution. On the other hand, even using the Bayesian DL models and improving the reliability and confidence of the results, there is still a need to use methods to interpret the results. Several methods can be used to generate visual interpretations of the results. For example, Grad-CAM [35] can be used if dealing with image classification tasks. This paper makes contributions toward maintaining the sustainability and efficiency of SP which can be summarized as follows:

1. This paper introduces an approach called Solar-OBNet for the accurate detection of dusty SP and snow-covered SP by classifying SP into three categories: clean SP, dusty SP, or snow-covered SP. Clean SP indicates that no cleaning is required whereas either

- dusty SP or snow-covered SP means that cleaning is needed.
2. The proposed Solar-OBNet is relying on the use of a Bayesian CNN model by employing the MCD technique.
 3. The best value for the dropout rate is determined via an optimization algorithm called Harris Hawks Optimizer (HHO) due to its significant impact on the posterior distribution.
 4. Two measures of uncertainty, standard deviation, and predictive entropy, are utilized to assess the predictive uncertainty over the posterior distribution. These two measures serve as a warning indicator for uncertainty if predictions turn out to be incorrect.
 5. To interpret the results of the proposed Solar-OBNet to ensure its efficiency and avoid being a black box, Grad-CAM is used.

The following are the remaining sections of the paper. Section 2 highlights related work that provides AI models for the identification of dusty SP and snow-covered SP. Section 3 provides the theoretical backgrounds underlying the proposed Solar-OBNet, and Sect. 4 discusses the dataset utilized in this study. The proposed Solar-OBNet is discussed in Sect. 5, while Sect. 6 shows and describes the proposed Solar-OBNet's results. Lastly, the conclusions are presented in Sect. 7.

2 Literature review

There are research efforts that have been made to enhance the efficiency of SP and mitigate the negative effects of dust and snow on SP. Researchers have used a variety of techniques for the classification and detection of dust in different applications, such as k-nearest neighbors (KNN) [36], random forest [37] with 90% accuracy. Other researchers have taken to measuring the dimensions of dust particles using high-resolution images [38]. A model called SolNet was proposed in [6] to detect dusty SP. SolNet achieved an average accuracy of 98.5%, outperforming VGG-16, AlexNet, and ResNet-50 on a newly proposed dataset of images of SP collected from Bangladesh.

To improve the efficiency of large-scale industrial cogeneration facilities, this research [39] offers a novel image classification technique that simultaneously detects and categorizes dust and soil on PV arrays. The suggested method uses high-resolution camera images of PV arrays, seasonal weather data for the industry, and cogeneration power plant operational parameters as training data. In order to perform combined classification and detection tasks, the training data is then input into the DL framework, which comprises CNN-based models and long-short-term

memory (LSTM) models. The suggested method got 96.54% accuracy. Both SqueezeNet and AlexNet transfer learning techniques are utilized for image categorization in [40]. The Fluke TiS60 Thermal Imager is used in this study to identify hotspots caused by dust buildup and to classify faulty and healthy photos. AlexNet outperformed SqueezeNet with an accuracy of 99.3%. To anticipate the concentration of unevenly accumulated dust, a DL network was employed in combination with image processing [41], where 3.67% mean absolute error (MAE) is obtained. Additionally, AI techniques have been used to predict the performance of the SP. Binary labeling was utilized to capture 30,000 photos, and the power loss was estimated while maintaining the same irradiance level using CNN LeNet model in [42]. This model achieved 80% accuracy and a mean squared error (MSE) of 0.0122. Similarly, the AlexNet, LeNet, and VGG-16 models were applied in [43] to a dataset of 599 images for detecting panels defects, with AlexNet outperformed the rest with an accuracy of 93.3%. Researchers have introduced an approach called DeepSolarEye based on CNNs to estimate power loss and soiling localization; they presented bi-directional input-aware fusion (BiDIAF) block to enhance CNNs localization abilities [44]. For snow detection, VGG-19, ResNet-50, and InceptionV3 have been applied on drone, augmented image dataset of SP with snow layers in [29]. They reported the f1-score of the three models with 100% for Inception V3, 99% for VGG-19, and 91% for ResNet-50. In [45], multiple ML models were utilized to detect the presence of different sources of pollution on SP surfaces: dust, branch, leaves, and powder. Each ML model was trained on a numerical dataset collected by manually polluting SP with these types of pollution. Various parameters are then measured taking into account weather parameters including temperature and irradiation. An accuracy of 97.37% and a f1-score of 83.2% were attained by the suggested ensemble model [45].

Despite the good results of the models proposed in the literature for detecting fouling of solar panels that affect their efficiency and discussed above, no reliable model has been presented that can express the level of confidence in its predictions [6, 29, 36–45]. In important applications such as energy-related applications, predictive systems that are responsible, transparent, and reliable must be relied upon. Quantifying uncertainty is a critical first step to delivering transparent, responsible, and reliable systems, and Bayesian DL is one of the most important ways to achieve this.

3 Basics and background

This section discusses the theoretical background on which the proposed Solar-OBNet is based, namely BCNN and HHO algorithm.

3.1 Convolutional neural networks

In the fields of image recognition, CNNs are the most widely applied DL networks. This is a result of its two-part architecture, the convolutional part dedicated to extracting the most important features from input images, and the classification part dedicated to classifying images using the extracted features. Convolutional layers, activation layers and pooling layers are the fundamental set of layers that make up the convolutional part. The core set of layers that comprise the classification part are flatten layer, dense layers and activation layers. To enhance generalization, and prevent overfitting of CNNs, dropout layers can be added to both the convolutional and classification layers. To acquire the best outcomes in a fair length of time, CNNs must be trained on a vast quantity of labeled data and provide efficient computing resources. However, in many fields, availability of sufficient quantities of labeled data to train CNN from scratch is a major challenge. The most widely used method at the moment to address these flaws is the transfer learning method. This method is to utilize a CNN model trained on a dataset for a base task and take advantage of the features the model has learned on that dataset to serve as a starting point for customizing that model for another processing data. In order to customize the pre-trained CNN model for a different processing data, the classification part of the pre-trained CNN model must first be changed to one appropriate for the new processing data and then trained. To adapt the pre-trained CNN model more appropriately to the new processing data and improve performance, the last convolutional layers of the pre-trained CNN's convolutional part are unfrozen and retrained [46–48]. There are many pre-trained CNN models that can be leveraged and adapted for various other processing data such as DenseNet169 [49], MobileNet [50], and EfficientNetB0 [51].

In CNN models that are employed in classification tasks, the learning procedure entails iteratively optimizing a loss function such as the cross-entropy (CE) to find the best values for weights and biases (parameters) of the model. The optimization of the CE is based on the use of optimization algorithms such as Adam [52].

3.2 Bayesian convolutional neural networks

In a typical CNN model, the model's parameters are limited to a point estimate, producing specific outputs for the test inputs [53]. In contrast, the parameters of Bayesian CNN models (θ) are expressed as probability distributions and learned utilizing Bayesian inference. Given a dataset $x = \{x_1, x_2, \dots, x_n\}$ and corresponding labels $y = \{y_1, y_2, \dots, y_n\}$, the posterior distribution which can help Bayesian CNN models capture uncertainty of the model's parameters can be computed utilizing Bayes' rule as Eq. 1.

$$p(\theta|x, y) = \frac{p(y|x, \theta)p(\theta)}{p(y|x)} \quad (1)$$

Because of the extreme nonlinear nature of CNN models, which results from the presence of a large number of hidden layers and various activation functions, an accurate calculation of $p(\theta|x, y)$ is not possible. To overcome this, alternative methods such as MCD can be utilized to approximate Bayesian inference. In the MCD method, the posterior distribution can be acquired by activating the MCD at the test time and each test sample is passed to the model K times. With MCD activated during the testing time, a different output is generated with each forward pass of the same input. The predictive mean (P_m) of the obtained posterior predictive distribution (PPD) is the basis for measures that can be used to determine model uncertainty for input samples and can be computed as shown in Eq. 2 [54]. Predictive entropy (P_E) and standard deviation (SD) which are the measures used to quantify uncertainty in model predictions are calculated as shown in Eqs. 3 and 4, respectively. P_E reflects how a model's prediction is far from the actual label, therefore, the lower the P_E value, the more confident the model is in its prediction. SD refers to the amount of variance in different results obtained by passing test samples K forward times, and the lower this variance, the higher the degree of confidence in the results [55, 56].

$$P_m = \frac{1}{K} \sum_k p(\hat{y} = x, \hat{\theta}_k) \quad (2)$$

$$P_E = - \sum_c P_m \log P_m \quad (3)$$

$$SD = \sqrt{\frac{1}{K} \sum_k (p(\hat{y} = c|x, \hat{\theta}_k) - P_m)^2} \quad (4)$$

where K represents the number of forward passes of the test samples, x is the test sample, $\hat{\theta}_k$ is the model's parameters on a specific k th forward pass, c is the Softmax

output, and $p(\hat{y} = c|x, \hat{\theta}_k)$ represents the probability that \hat{y} belongs to c .

3.3 Harris Hawks optimizer algorithm

Harris Hawks optimizer (HHO) is a mathematical algorithm that simulates the dynamics of Harris hawk predator–prey relationship [57, 58]. This algorithm is relatively new and was introduced in [57]. In [57], HHO has achieved the best global optimum for multi-dimensional problems compared to eleven optimizers, including the particle swarm optimization (PSO) [59], the moth flame optimizer (MFO) [60], and the flower pollination algorithm (FPA) [61]. In addition, HHO achieved a reasonably fast performance in finding the best solutions. This is because HHO uses a variety of search techniques based on adaptive parameters to select the optimal movement. Furthermore, by using time-varying parameters, HHO is able to tackle a search space's challenges, such as local optimal solutions, and misleading optima. Therefore, in the context of DL and the high-dimensional data it deals with, HHO is well suited for solving problems related to DL models [62–64].

The HHO algorithm is divided into three phases: exploring, transition and exploitation. Initially, A population of N positions of Harris Hawks are randomly initialized $z_i, i = 1, \dots, N$. In the exploration phase, Hawks use the following two strategies to locate its prey: they either move according to the positions of other family members, this is in order to be close enough to them at the time of attacking, or according to the rabbit position. The selection between the two strategies is modeled using random variable st such that if $st < 0.5$, the hawks change their locations based on the rabbit position, whereas for $st \geq 0.5$, they choose their locations randomly inside the group's home range. The positions are updated as follows in Eq. 5:

$$z(t+1) = \begin{cases} z_{\text{rand}}(t) - r_1 |z_{\text{rand}}(t) - 2r_2 z(t)| & st \geq 0.5 \\ z_{\text{rabbit}}(t) - z_{\text{mean}}(t) - r_3(L - r_4(U - L)) & st < 0.5 \end{cases} \quad (5)$$

where $z(t+1)$ is the Hawk's positions vector of the next iteration and $z(t)$ is the current positions vector of the current iteration. t is the current iteration number. $z_{\text{rand}}(t)$ is a random position chosen from the population, and $z_{\text{rabbit}}(t)$ is the rabbit. L, U are the corresponding lower and upper bounds for generating random locations inside the Hawks' house. All st, r_1, r_2, r_3 , and r_4 are random numbers between 0 and 1. $z_{\text{mean}}(t)$ is the mean position of current Hawk's individuals in the population, and it is calculated as shown in Eq. 6.

$$z_{\text{mean}}(t) = \frac{1}{N} \sum_{i=0}^N z_i(t) \quad (6)$$

where $z_i(t)$ is the position vector of the i th hawk in the population for the current iteration t . N denotes the number of Hawks in the population. In the transition phase, the energy of the escaping rabbit controls whether the Hawks perform exploration or exploitation. The energy of the rabbit is given by:

$$E_{\text{rabbit}} = 2E_0 \left(1 - \frac{t}{\text{max_iter}} \right) \quad (7)$$

where E_0 represents the initial energy of the rabbit, and it is randomly produced between $[-1, 1]$. max_iter denotes the maximum number of iterations. When $|E_{\text{rabbit}}| \geq 1$, the exploration phase continues, whereas the exploitation phase begins when $|E_{\text{rabbit}}| < 1$. In the exploitation phase, the Hawks attempt to attack the rabbit. At the same time, the rabbit tries to escape with a 50% chance of success and failure denoted as c . The escaping rabbit energy specify how the Hawks will accomplish its goal, such that, if $|E_{\text{rabbit}}| \geq 0.5$ and $c \geq 0.5$, soft besiege is done. During soft besiege, the Hawks gently surround the rabbit to tire it out before launching an unexpected attack. The following is a mathematical formulation of the soft besiege.

$$z(t+1) = z_{\text{diff}}(t) - E_{\text{rabbit}} |J z_{\text{rabbit}}(t) - z(t)| \quad (8)$$

where $z_{\text{diff}}(t)$ is the difference between the hawk position and the rabbit position at the current iteration. $J = 2(1-r_5)$ stands for the random jump strength of the rabbit during the escaping process. To replicate the characteristics of rabbit motions, the J value varies randomly with each iteration. In case $c < 0.5$ and $|E_{\text{rabbit}}| \geq 0.5$, the rabbit has enough energy to successfully escape, soft besiege with progressive rapid dives is performed. The Hawks are able to select the best possible dive. Lévy flight is utilized to simulate the zigzag deceptive movements of the escaping rabbit. To determine whether the dive is good, the next move of the Hawks is calculated as follows:

$$\text{mov}_{\text{next}} = z_{\text{rabbit}}(t) - E_{\text{rabbit}} |J z_{\text{rabbit}}(t) - z(t)| \quad (9)$$

Then, in order to determine whether or not the dive would be successful, they compare the potential outcome of such a movement to the prior dive. If the prior dive proves to be ineffective, the Hawks will dive using Levy flight L pattern as follows

$$z_{\text{levy}} = \text{mov}_{\text{next}} + w \text{Levy}(D) \quad (10)$$

where D is the problem dimension, w denotes a random vector of size D and L is the function representing Levy flight.

$$\text{Levy}(x) = 0.01 \left(\frac{u \times \sigma}{|v|^{\frac{1}{b}}} \right) \quad (11) \quad z(t+1) = \begin{cases} \text{mov}_{\text{nxt}}, & \text{if } F(\text{mov}_{\text{nxt}} < z(t)) \\ z_{\text{levy}}, & \text{if } (z_{\text{levy}} < z(t)) \end{cases} \quad (13)$$

$$\sigma = \left(\frac{\Gamma(1+b) \times \sin(\frac{\pi b}{2})}{\Gamma(\frac{1+b}{2}) \times b \times 2^{\frac{(b-1)}{2}}} \right)^{\frac{1}{b}} s \quad (12) \quad z(t+1) = z(t) - E_{\text{rabbit}} |Jz_{\text{diff}}(t)| \quad (14)$$

Algorithm 1 Pseudocode of HHO algorithm

Inputs: N, max_iter
Output: The rabbit's location z_{rabbit} and its fitness value.

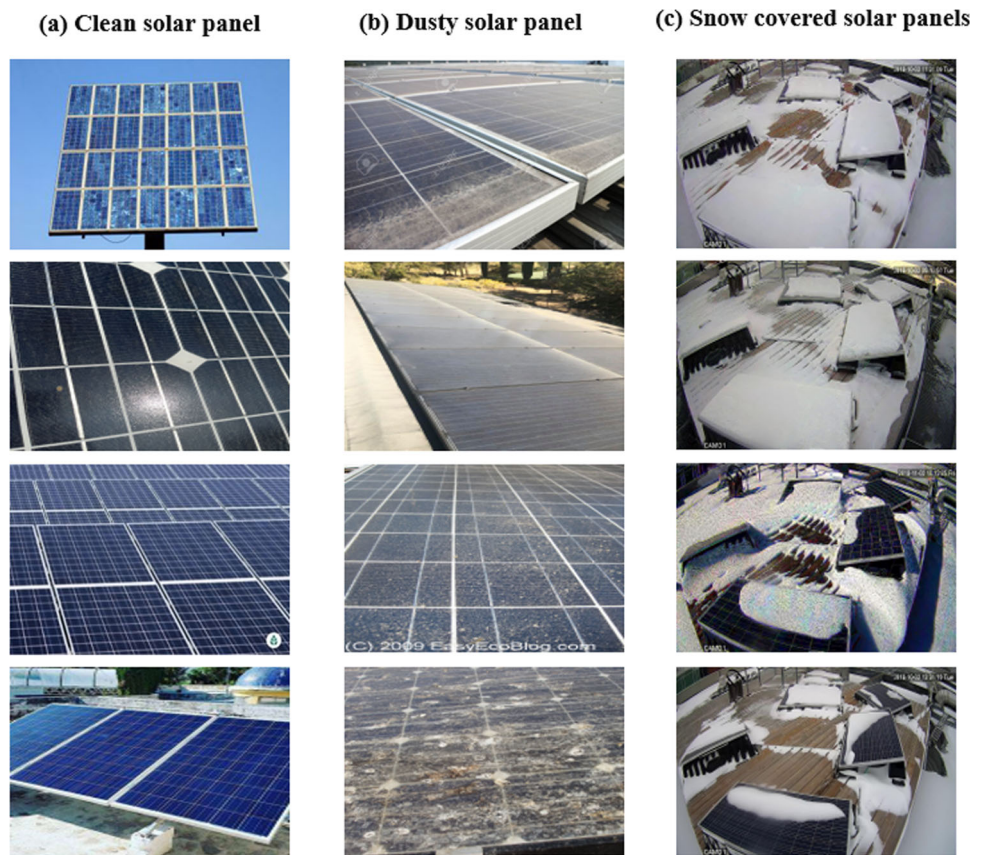
1. Initialize $Z_i, i = (1 \dots N)$ randomly.
2. **While** (stopping condition is not met), **Do**
3. Calculate the fitness value of each hawk.
4. **Set** Z_{rabbit} as the rabbit position (best position)
5. **For** each hawk, **Do**
6. Update energy E_{rabbit} utilizing Eq. 7 and jump strength $J = 2(1 - r_5)$
7. **If** $E_{\text{rabbit}} \geq 1$, **Then**
8. // Exploration
9. Update hawk's position utilizing Eq. 5
10. **Else:**
11. **If** ($E_{\text{rabbit}} \geq 0.5$ and $c \geq 0.5$) **Then**
12. // Soft Besiege
13. Update hawk position utilizing Eq. 8
14. **Else If** ($E_{\text{rabbit}} \geq 0.5$ and $c < 0.5$) **Then**
15. // Soft Besiege with progressive rapid dives
16. Update hawk position utilizing Eq. 13 with parameters calculated utilizing Eq. 9 and Eq. 10.
17. **Else If** ($E_{\text{rabbit}} < 0.5$ and $c \geq 0.5$), **Then**
18. // Hard Besiege
19. Update hawk position utilizing Eq. 14
20. **Else**
21. // Hard Besiege with progressive rapid dives
22. Update hawk position utilizing Eq. 13 with parameters calculated utilizing Eq. 15 and Eq. 10.
23. **End If**
24. **End For**
25. **End While**
26. **return** Z_{rabbit}

where u and v are random numbers from 0 to 1, and b is a constant equal to 1.5. The final soft besiege progressive rapid dives are updated utilizing Eq. 13. On the other hand, when $|E_{\text{rabbit}}| < 0.5$ and $c \geq 0.5$, the hard besiege is done in which the rabbit is so tired and has a low escaping energy. Furthermore, the Harris' Hawks barely circle the selected rabbit before making their final unexpected attack. Hard besiege can be described as Eq. 14. When $|E_{\text{rabbit}}| < 0.5$ and $c < 0.5$, the rabbit lacks the energy to escape and a hard besiege is organized before the unexpected attack to capture and murder the rabbit. This is called hard besiege with progressive rapid dives, in which $z(t+1)$ is calculated as Eq. 13 except that mov_{nxt} is calculated as Eq. 15. Algorithm 1 summarizes the HHO algorithm.

4 Dataset description

The solar panel soiling (SPS) dataset presented in this paper is used to detect clean SP, dusty SP, and snow-covered SP. The dataset includes 725 images; of which 300 are images of clean SP, 300 images of dusty SP, and 125 images of SP completely and partially covered with snow. The images in the first column in Fig. 1 are some images of clean SP, the images in the second column are some images of dusty SP, and the images in the third column are some images of snow-covered SP. All images in the SPS dataset were selected from two public datasets. The first dataset is available on Kaggle [65], it has 1493 images of clean SP and 1069 images of dusty SP. The second dataset has been published in [29] and available on [66], it contains 270 images of SP without snow on them, 27 images of SP

Fig. 1 Few samples of clean SP, dusty SP, and snow-covered SP in the SPS dataset



partially covered in snow, and 98 images of SP completely covered with snow. Images of clean SP and dusty SP in the SPS dataset were selected from the first dataset, while images of SP completely and partially covered with snow were selected from the second dataset.

5 Proposed Solar-OBNet

As depicted in Fig. 2, the proposed Solar-OBNet primarily includes three phases: data preparation phase, optimization and training phase, and inference phase. Explanations of each of these phases can be found in the ensuing subsections.

5.1 Data preparation phase

The SPS dataset is separated into three sets: training, validation, and test. 70% of each of the clean solar panel images, dusty solar panel images, and snow-covered solar panel images were assigned to the training set, at this phase. 15% of the clean solar panel images, dusty solar panel images, and snow-covered solar panel images were assigned to the test set and the same to the validation set. In other words, there are 507 images in the training set; 210 of

these images are of clean SP, 210 images of dusty SP, and 87 images of snow-covered SP. On the other hand, both the test and validation sets contain 109 images; 45 of them are from the clean solar panel category, 45 images are from the category of dusty SP, and 19 images are from the category of snow-covered SP, as depicted in Fig. 3. Before starting the second phase of the proposed approach, all samples in training, validation, and test sets are normalized and data augmentation techniques were employed to the training set to avoid overfitting. This phase is also explained in steps 1–3 in Algorithm 2.

5.2 Optimization and training phase

At this phase, multiple pre-trained CNNs, including VGGNet, DenseNet169, MobileNet, ResNet101V2 and EfficientNetB0, were tested in order to determine which would be the best at detecting solar panel soiling. According to preliminary results, DenseNet169 was able to detect soiling of SP with higher accuracy. The pre-trained DenseNet169 was adapted to the SPS dataset by preserving its convolutional part and changing the classifier with another four-layer classifier, as depicted in steps 4–8 in Algorithm 2. The four layers of the new classifier part are one flatten layer, two fully connected (FC) layers

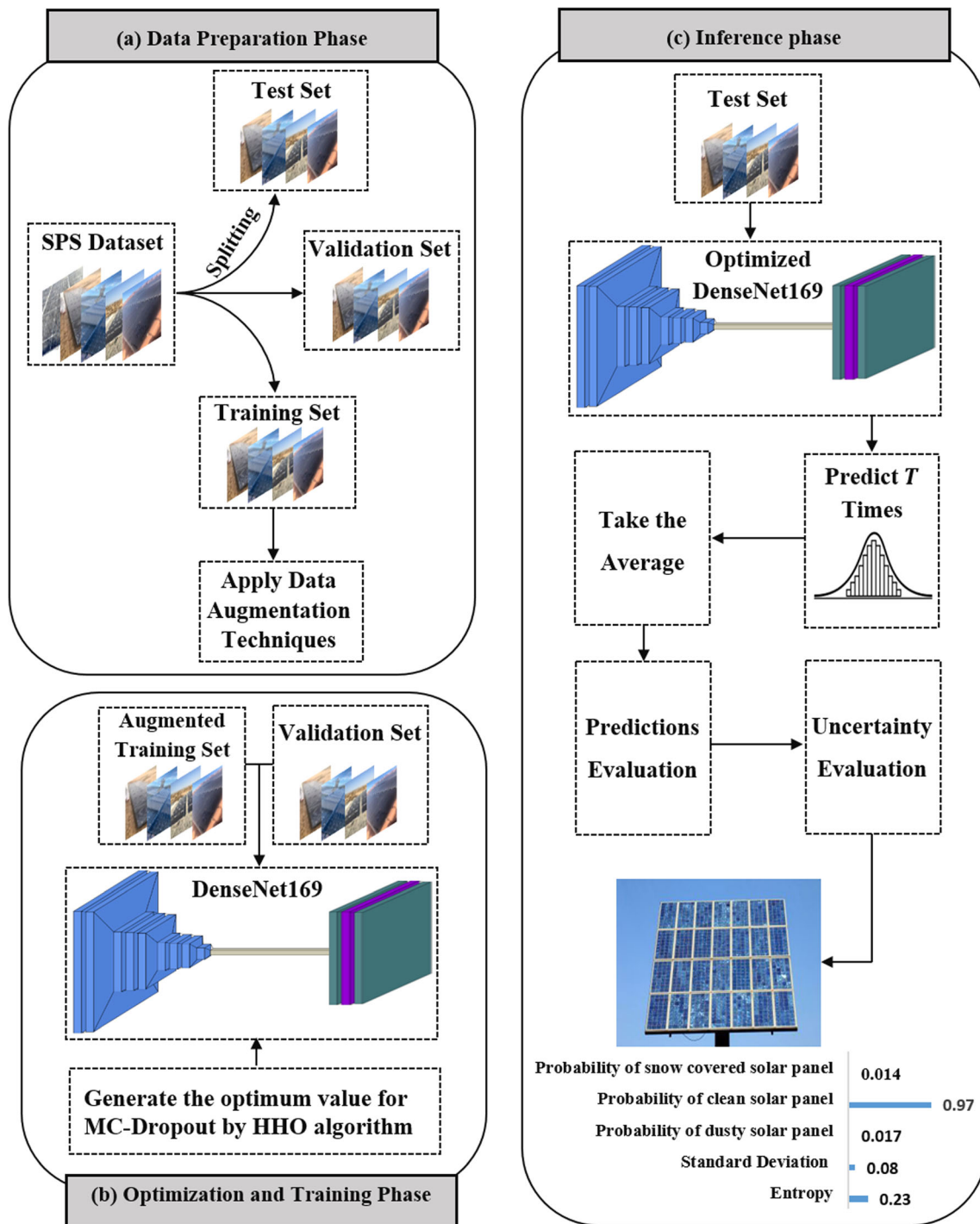


Fig. 2 Structure of the proposed Solar-OBNet. SPS Dataset, solar panel soiling dataset

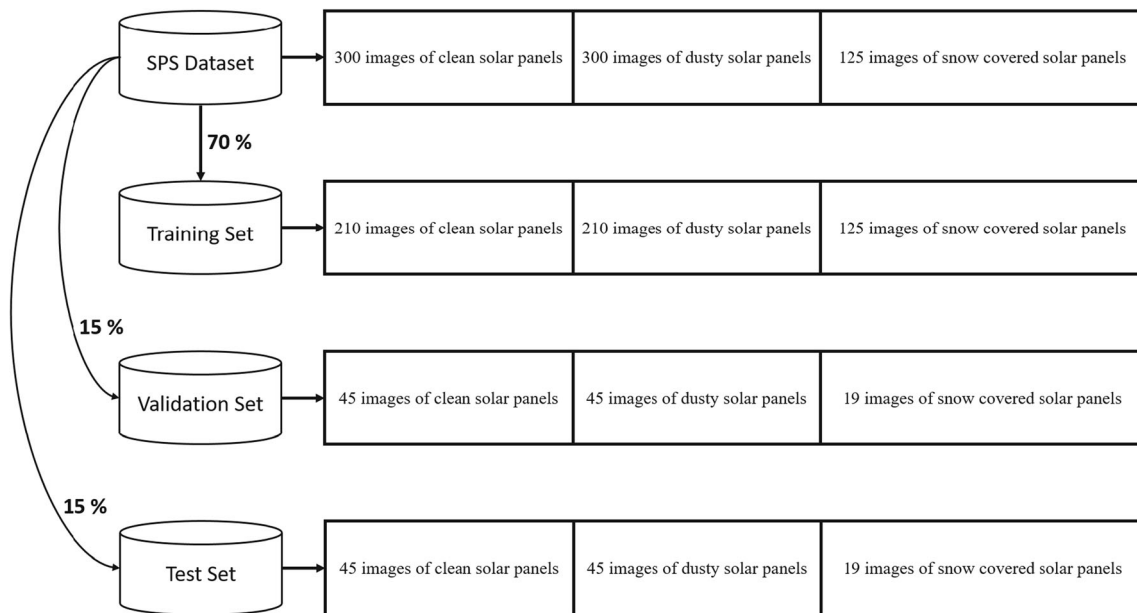


Fig. 3 Illustrative structure of the data distribution in the training set, validation set, and test set

interspersed with one MCD layer to make it Bayesian. Adding one MCD layer is a simple trick, but it has a huge impact on converting the adapted DenseNet169 to a Bayesian network. The PPD obtained from the Bayesian network is significantly affected by the MCD rate. The PPD will be divergent if the MCD rate is quite high. In contrast, the PPD will be convergent if the MCD rate is relatively low. Therefore, MCD rate must be chosen carefully. In such a situation where there are no theoretical guidelines to determine the appropriate MCD rate, relying on an optimization algorithm to determine the MCD rate is an effective solution instead of the random selection method [67]. Therefore, in this phase the optimal MCD rate is determined using HHO algorithm, as depicted in step 7 in Algorithm 2. The HHO algorithm generates the MCD rate and feeds it into the adapted DenseNet169. The training procedure for the adapted DenseNet169 begins as soon as it receives the generated MCD rate. The MCD's optimal rate is established when the HHO algorithm has completed its maximum number of iterations. The adapted DenseNet169 is then trained using the MCD value that was obtained by the HHO algorithm, as shown in step 15 in

Algorithm 2. The adapted DenseNet169 model is trained by keeping the convolutional part of the adapted DenseNet169 frozen, while the new classifier is trained for a predetermined number of iterations. For best results, the adapted DenseNet169 was trained again by unblocking the training of the last four layers of the convolutional part and training them together with the classifier part for a specified number of iterations.

5.3 Inference phase

In this phase, the adapted DenseNet169 is evaluated after being trained in the previous phase using the test set while keeping the MCD activated. Each test sample is predicted T times to get the PPD, then P_m is calculated according to Eq. 2, as shown in steps 19–22 in Algorithm 2. Hence, two uncertainty metrics are generated to quantify the proposed Solar-OBNet's predictions uncertainty, as shown in steps 23 and 24 in Algorithm 2. In addition, the proposed Solar-OBNet predictions are evaluated using several evaluation metrics, as shown in step 25 in Algorithm 2.

Algorithm 2 Pseudocode for the proposed Solar-OBNet

Input: Dataset (D), Pre-trained model DenseNet169
Output: Prediction results with the degree of uncertainty
Data Preparation Phase
1. Split D into 70 % training set (D_{train}), 15% validation set ($D_{validation}$), and 15% test set (D_{test}) 2. Normalize all samples in D_{train} , $D_{validation}$, D_{test} 3. Apply data augmentation techniques (D_{train})
Optimization and training phase
// Optimization phase 4. Model= Convolutional_part (DenseNet169) 5. Model. Add (flatten layer) 6. Model. Add (FC dense layer with 80 neurons and RELU) 7. Model. Add (MCD layer with a rate that would be set by HHO algorithm (Algorithm 1)) 8. Model. Add (output layer with 3 neurons and Softmax) 9. Model. Compile (Optimizer=Nadam (learning_rate=2e-5), Loss=categorical CE) 10. Train the model on D_{train} , and evaluate it on $D_{validation}$ for 5 epochs 11. Evaluate the MCD rate set by HHO algorithm on $D_{validation}$ // Training phase 12. Model= Convolutional_part (DenseNet169) 13. Model. Add (flatten layer) 14. Model. Add (FC dense layer with 80 neurons and RELU) 15. Model. Add (MCD layer with the rate set by HHO algorithm) 16. Model. Add (output layer with 3 neurons and Softmax) 17. Model. Compile (Optimizer=Nadam (learning_rate=2e-5), Loss=categorical CE) 18. Train the model on D_{train} , and evaluate it on $D_{validation}$ for 50 epochs with applying early stopping and step decay
Inference phase
19. for $k = 1$ to 100 20. Model. Predict (D_{test}) 21. end for 22. Calculate P_m according to Eq. 2 23. Calculate P_E according to Eq. 3 24. Calculate SD according to Eq. 4 25. Evaluate Model results using several evaluation metrics

6 Experimental results and discussion

This section shows a detailed presentation of the experimental setup for the three phases of the proposed Solar-OBNet that were discussed in Sect. 5. The results of all phases are also presented, and a comparison between the results of state-of-the-art (SOTA) methods and the proposed Solar-OBNet is made in this section. On a Kaggle notebook with 16 GB GPU and 14 GB RAM, the proposed Solar-OBNet phases were executed utilizing Python and Keras library. Utilizing this specific environment, the phases of the proposed Solar-OBNet were implemented in six hours, with most of this time taken up in the optimization and training phase.

6.1 Experimental setup

6.1.1 Data preprocessing

The training set was expanded with more samples by applying six data augmentation technique before it was used in the proposed Solar-OBNet's last two phases. Augmentation techniques generate many versions of the images which greatly helps in avoiding overfitting. Rotation augmentation, zoom augmentation, shear augmentation, height shift augmentation, width shift augmentation, and horizontal flip augmentation are the data augmentation techniques employed. The corresponding values of the augmentation techniques are rotation angle: 10° zoom range: 0.25, shear range: 0.3, height shift: 0.2, width shift: 0.2, and horizontal flip: True. Figure 4 displays some samples of the augmented images.

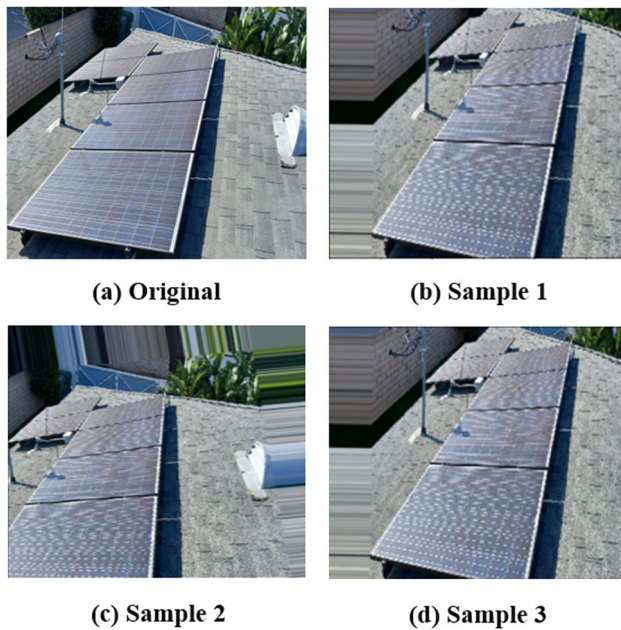


Fig. 4 Some images of dusty SP generated from applying various augmentation techniques. **a** The original dusty solar panel image, and **b–d** are the images that were produced by applying certain data augmentation techniques

6.1.2 Optimization and training phase

To find the optimal settings for the hyper-parameters of the adapted DenseNet169, which vary depending on the problem to be addressed, several experiments based on random selection were conducted. The hyper-parameters that significantly influence the performance of the pre-trained models are the batch size (BS), learning rate (LR), activation function, optimization function, number of neurons in the FC layers, and input image size. The experiments conducted have indicated that the adapted DenseNet169 begins to achieve good performance when the hyper-parameter settings are as stated in Table 1. It was

Table 1 Values assigned to the hyper-parameters of the adapted DenseNet169 model

Hyper-parameter	Corresponding setting
Input size	200
Count of units in the first FC layer	80
The first FC layer's activation function	RELU
Count of units in the second FC layer	3
The second FC layer's activation function	Softmax
Loss function	Categorical CE
BS	4
Optimization function	Nadam
The initial value of the LR	$2e-5$
MCD rate	0.37

found that the results of the adapted DenseNet169 are much better when the input size is 280×280 pixels. Since there was no discernible impact on the outcomes when the number of units or neurons in the first FC layer was more than 80 units and only the time the model needed to train increased, the number of units in the first FC layer was set to 80. For the first FC layer, the RELU activation function [68] was employed, as depicted in step 6 in Algorithm 2. The output layer, which is the second FC layer, contains as many units as there are classes in the SPS dataset. For the second FC layer, the SoftMax activation function was employed. With a LR of $2e-5$, Nadam [69] was shown to update model weights and converge more quickly and effectively than others in this task after trying a variety of optimization algorithms, including Adam [70], AdaMax [69], and Nadam. The loss function used in the architecture of the adapted DenseNet169 is the categorical CE, as shown in step 9 in Algorithm 2.

After selecting suitable hyper-parameters for the adapted DenseNet169, an MCD layer was added after the first FC layer and then the adapted DenseNet169 was combined with the HHO algorithm to determine the ideal MCD rate. The HHO's parameter values were chosen randomly, the max_iter and population size of HHO algorithm were set to 5 and 10, respectively. The search range for the MCD value, whose value will be set by HHO algorithm, is restricted to [0.1, 0.9]. It was noticed that the optimization procedure takes exponentially more time when training the adapted DenseNet169 more than five epochs. Therefore, the number of training epochs for the adapted DenseNet169 model was set to five, as shown in step 10 in Algorithm 2. The HHO algorithm's fitness function is to reduce the validation set's loss rate as possible, as shown in step 11 in Algorithm 2. The ideal MCD rate was found by the HHO algorithm to be 0.37 after the pre-specified number of iterations of this algorithm were over, as depicted in Table 1.

After completing the optimization phase, in which the optimal rate of the MCD is determined, the adapted DenseNet169 model is trained with this determined value for 50 epochs while keeping the convolutional part of the adapted DenseNet169 frozen, as shown in step 18 in Algorithm 2. The adapted DenseNet169 is trained again by unblocking the training of the last four layers of the convolutional part for 50 epochs to improve the results. The early stopping technique [71] was utilized to compel the training to end before the specified number of training epochs had been completed if no progress was noticed for 10 epochs. The purpose of using this technique is to reduce the possibility of overfitting. The LR decay [72] is utilized to improve the model's performance, as the value of the LR falls by 0.2 every five epochs in the event that accuracy does not rise.

6.1.3 Inference phase

The PPD is obtained at this phase by making 100 predictions for each test sample, after which the average of those predictions is then calculated. The P_E and SD of the PPD are calculated as Eqs. 3 and 4, respectively, to assess the predictive uncertainty of the proposed Solar-OBNet. P_E demonstrates the degree of uncertainty in the forecast, while SD demonstrates variances in the predictions of the proposed Solar-OBNet. Six performance metrics are used to evaluate Solar-OBNet's results using the average of those predictions. These performance metrics include the accuracy, precision, sensitivity, specificity, f1-score, and balanced accuracy, which are calculated as Eqs. 16–21.

$$\text{Accuracy} = \left(\frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \right) \times 100 \quad (16)$$

$$\text{Balanced Accuracy} = \left(\frac{\text{Sensitivity} + \text{Specificity}}{2} \right) \times 100 \quad (17)$$

$$\text{Sensitivity or Recall} = \left(\frac{\text{TP}}{\text{TP} + \text{FN}} \right) \times 100 \quad (18)$$

$$\text{Precision} = \left(\frac{\text{TP}}{\text{TP} + \text{FP}} \right) \times 100 \quad (19)$$

$$\text{Specificity} = \left(\frac{\text{TN}}{\text{FP} + \text{TN}} \right) \times 100 \quad (20)$$

$$F1 - \text{Score} = 2 \left(\frac{\text{Sensitivity} \times \text{Precision}}{\text{Sensitivity} + \text{Precision}} \right) \times 100 \quad (21)$$

where TN = True Negative, and TP = True Positive, FP = False Positive, and FN = False Negative.

6.2 Results of the proposed Solar-OBNet

In this experiment, the test set includes 109 samples, of which 45 samples out of 109 belong to the clean solar panel category, 45 samples belong to the dusty solar panel category, and 19 samples belong to the snow-covered solar panel category. As shown in Fig. 5a, which displays the confusion matrix showing the proposed Solar-OBNet's performance in classifying test samples, it misclassified only eight samples out of the total test samples. By examining the samples that were misclassified by the proposed Solar-OBNet, it can be observed that there is an overlap between the clean solar panel category and the dusty solar panel category. This is due to the difficulty of differentiating between these two categories, as some samples in the category of dusty SP contain a very small amount of dust, which made the proposed Solar-OBNet misclassify them.

Although the proposed Solar-OBNet misclassified some test samples, it is able to show uncertainty in its predictions for these samples through the high values of P_E and SD as shown in Fig. 6. Figure 6 presents samples mistakenly classified by the proposed Solar-OBNet, and the top two samples are from the clean SP class but classified as the dusty SP class. The last two samples are from the dusty SP class but were mistakenly classified as the clean SP class. As shown in Fig. 6, below each misclassified sample is a detailed list of results obtained from the proposed Solar-OBNet, where the probability of each category and the P_E and SD values are listed. The ability of the proposed Solar-OBNet to reflect great uncertainty in the misclassification is demonstrated by the high values of SD and P_E for each sample.

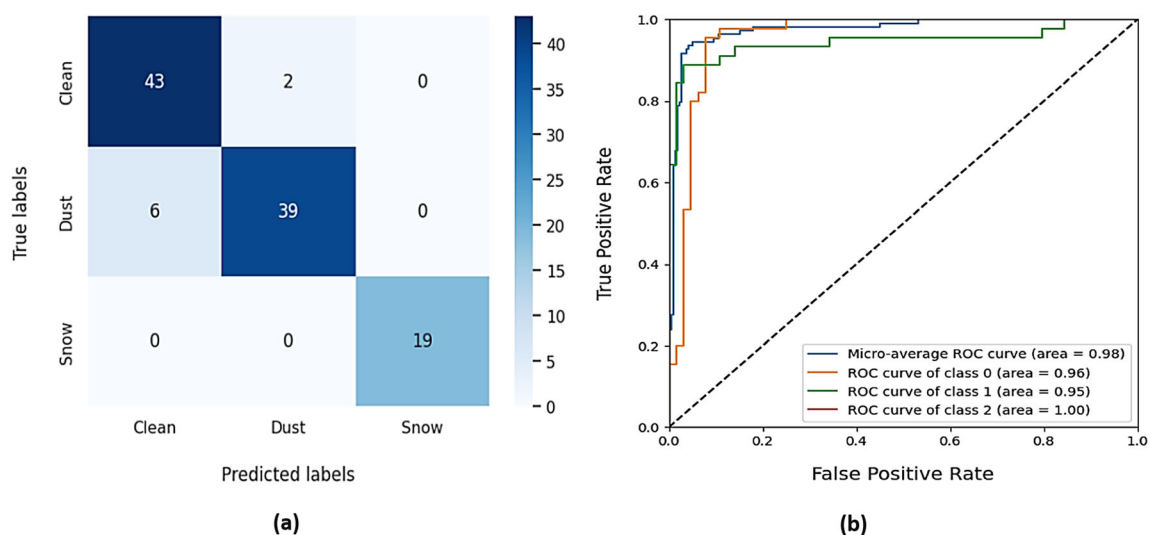


Fig. 5 Proposed Solar-OBNet's performance on the test set: **a** confusion matrix, **b** ROC-AUC curve, class 0 = clean SP class, class 1 = dusty SP class, and class 2 = snow-covered SP class

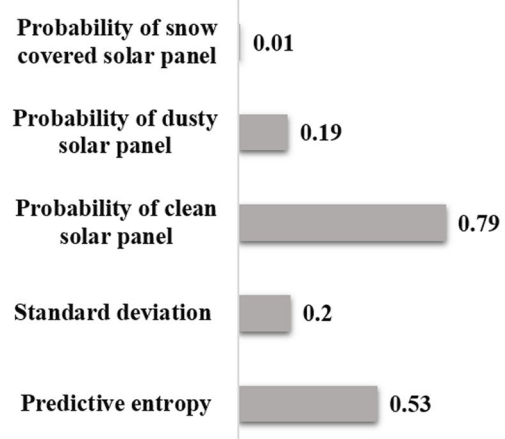
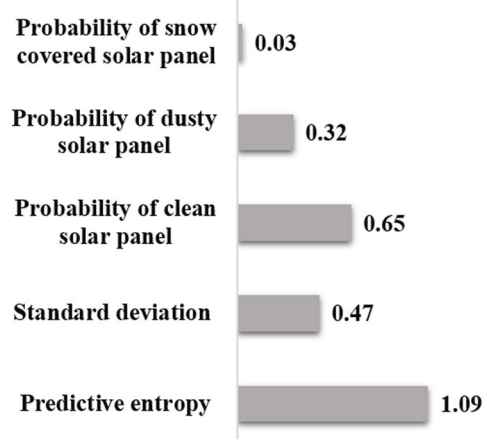
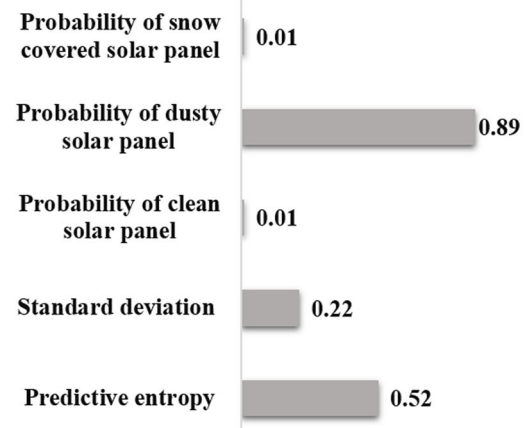
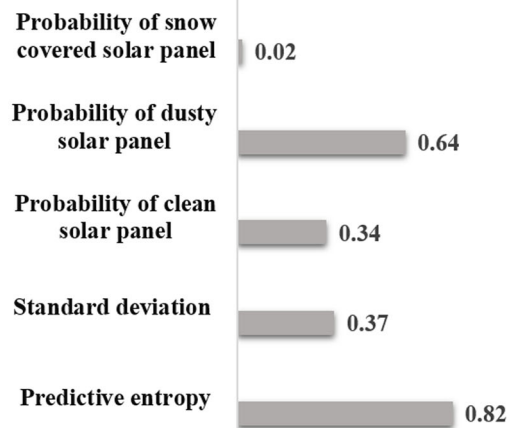


Fig. 6 Four incorrectly classified test samples by the proposed Solar-OBNet with stating each class's probability, standard deviation, and predictive entropy for each sample below it

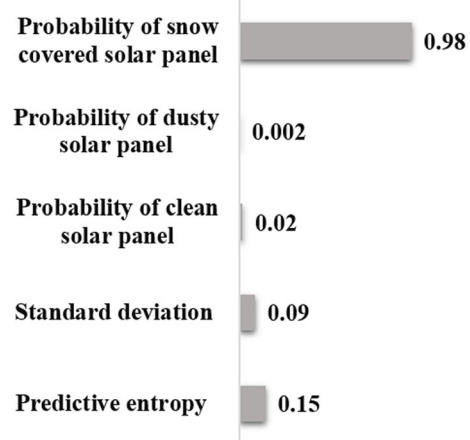
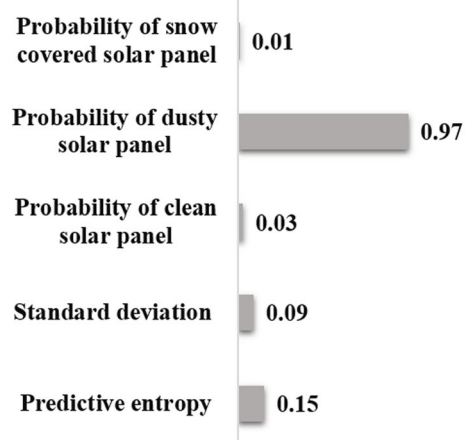
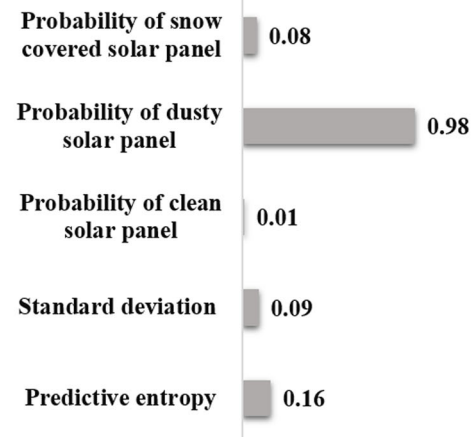
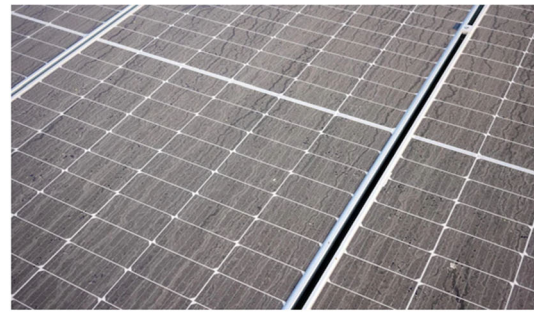
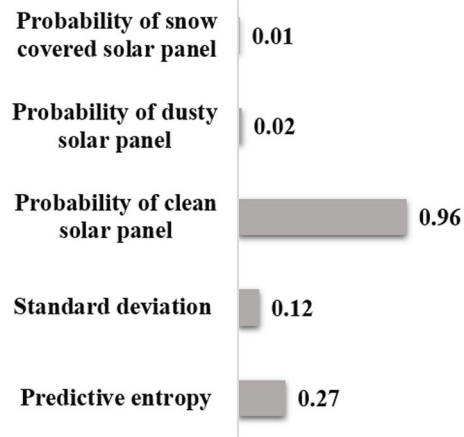
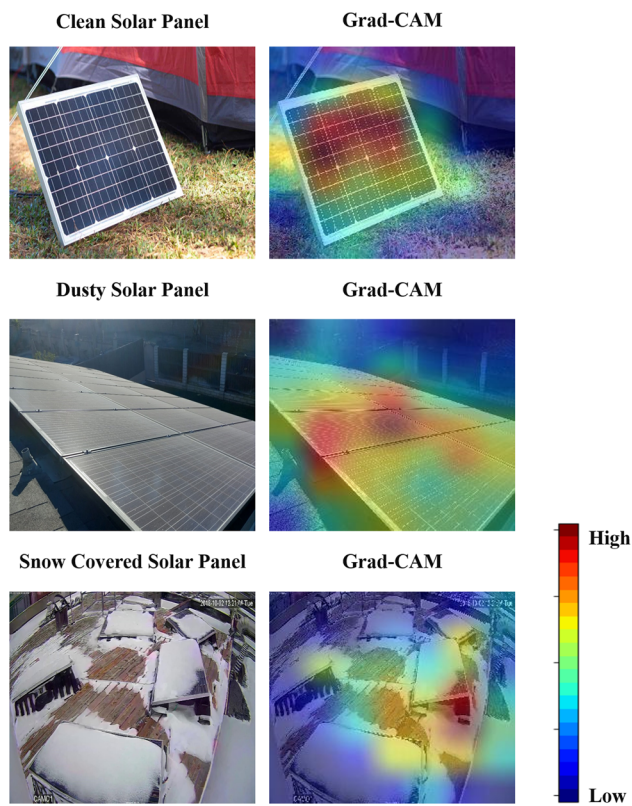


Fig. 7 Four correctly classified test samples by the proposed Solar-OBNet with stating each class's probability, standard deviation, and predictive entropy of each sample below it

Table 2 Performance results of the proposed Solar-OBNet on the test set

Performance metric	Value
Accuracy	92.66%
Average sensitivity (or recall)	94.07%
Average precision	94.29%
Average f1-score	94.06%
Average specificity	95.83%
Balanced accuracy	94.07%
P_E	0.39
SD	0.165

**Fig. 8** Grad-CAM visualization of some test samples correctly classified by the proposed Solar-OBNet. The color indicates the degree of activation from the most significant feature (red), and significant feature (yellow), to low importance feature (green) and the least important feature (blue) (color figure online)

Given Fig. 7, which presents some samples that were classified correctly by the proposed Solar-OBNet, it can be observed that the values of the P_E and SD are low. Low values for P_E and SD reflect the confidence of the proposed Solar-OBNet in its predictions. The first sample in Fig. 7 is an image of a clean solar panel that was classified correctly by the proposed Solar-OBNet. The second and third samples in Fig. 7 are images of dusty SP that were classified

correctly by the proposed Solar-OBNet. The last sample in Fig. 7 is an image of a snow-covered solar panel that has been correctly classified by the proposed Solar-OBNet. The small values of P_E and SD for each sample in Fig. 7 demonstrate how well the proposed Solar-OBNet can give a high degree of confidence in the correct predictions. This implies that the proposed Solar-OBNet's predictions with low P_E and SD are substantially accurate, and do not require human intervention.

As given in Table 2, the proposed Solar-OBNet yielded an accuracy of 92.66%, a balanced accuracy of 94.07%, an average f1-score of 94.06%, an average sensitivity of 94.07%, an average precision of 94.29%, and an average specificity of 95.83%. The balanced accuracy result demonstrates that the proposed Solar-OBNet can address the issue of data imbalance. The proposed Solar-OBNet produced an overall SD of 0.16, showing that its predictions are not very variable. As depicted in Table 2, the proposed Solar-OBNet yielded an overall P_E value of 0.39 indicating high confidence and low uncertainty in its predictions. Additionally, the proposed Solar-OBNet's performance was also examined utilizing a receiver operating characteristic (ROC) curve, with the X and Y axes representing the false positive rate (FPR) and true positive rate (TPR) with values between 0 and 1, as shown in Fig. 5b. The proposed Solar-OBNet's performance is more successful the closer the ROC curve is to the upper left corner. In addition, the area under the ROC curve (AUC) that is within the range [0.5, 1.0] can be utilized to verify the proposed Solar-OBNet's performance. The higher the AUC value, the more accurate the proposed Solar-OBNet is in differentiating between classes. As is well known, the ROC curve and AUC are typically for binary class classification tasks. In order to apply it for multiclass tasks, TPR and FPR are calculated for each class versus all others (assumed as one) and then the TPRs and FPRs of all classes are averaged to get the micro-average ROC curve and AUC. As can be observed in Fig. 5b, the ROC curve of the snow-covered SP class is so close to the upper left corner that it is not clear, and the AUC value is 1. The AUC of the ROC curve for the clean SP class, and the dusty SP class are 0.96, and 0.95, respectively. The micro-average AUC of the micro-average ROC curve is 0.98, as shown in Fig. 5b.

To make sure the proposed Solar-OBNet's robustness and interpret its predictions, Grad-CAM was employed. Grad-CAM is an algorithm applied to the last convolutional layer to highlight specific regions in the input image from which the DL model's predictions can be interpreted. This algorithm could be used as a screening method to see what features DL models use in making their predictions. In other words, Grad-CAM can act as a utility to see if the proposed Solar-OBNet is already using detailed and correct

Table 3 Performance evaluation of the proposed Solar-OBNet in comparison to other approaches

Evaluation metric	Proposed Solar-OBNet without optimization phase	Adapted DenseNet169	Proposed Solar-OBNet
Accuracy	88.99%	87.15%	92.66%
Average precision	91.33%	89.33%	94.33%
Average recall	91%	89.33%	94.33%
Average f1-score	91%	89.33%	94%
Average specificity	93.75%	92.71%	95.83%
Balanced accuracy	92.38%	91.02%	95.08%
P_E	0.76	–	0.39
SD	0.35	–	0.165

Bold indicates the best results

features and can provide understandable visual information about the proposed Solar-OBNet's performance. As illustrated in Fig. 8, there are three test samples that have been identified correctly by the proposed Solar-OBNet and next to each sample is the Grad-CAM generated for it. The first sample is a clean solar panel image with the generated Grad-CAM image next to it, and the second sample is a dusty solar panel image with the generated Grad-CAM image next to it. The third sample is a snow-covered solar panel with the generated Grad-CAM image next to it. Grad-CAM is used to visualize and highlight significant regions and features upon which the proposed Solar-OBNet is based to differentiate between clean SP, dusty SP, and snow-covered SP. The significant regions highlighted in red show that the proposed Solar-OBNet can successfully identify the image's class using the most useful features.

6.3 Comparison with SOTA methods

The Solar-OBNet's performance with the use of the optimization phase was compared with the performance without the use of the optimization phase. To evaluate the optimization phase's efficiency (the phase in which the optimal rate of MCD is set by the HHO algorithm) on the performance of the proposed Solar-OBNet. The MCD rate in the proposed Solar-OBNet without the optimization phase was randomly chosen to be 0.8. Looking at the results obtained from the proposed Solar-OBNet without the optimization phase in Table 3, the proposed Solar-OBNet's performance was severely affected when the MCD rate was set to this randomly chosen value. The proposed Solar-OBNet without the optimization phase obtained a very high P_E value of 0.76 in comparison with the proposed Solar-OBNet, which implies very high levels of uncertainty in the results. The SD value is greater than that achieved by the Solar-OBNet with the optimization phase as shown in Tables 2 and 3. Higher SD indicates higher variance in predictions of the Solar-OBNet without

the optimization phase. As given in Table 3, the Solar-OBNet without the optimization phase performed worse than the proposed Solar-OBNet in terms of all evaluation measures.

The proposed Solar-OBNet's results were compared with the results of the adapted DenseNet169 where no MCD layer was added. To compare the proposed Solar-OBNet's performance with that of the non-Bayesian approach. As shown in Table 3, the adapted DenseNet169 performed worse than the proposed Solar-OBNet in terms of all evaluation measures. This is an indication that the proposed Solar-OBNet is not only useful in providing uncertainty in predictions, but also achieves a higher level of accuracy. According to the comparison, it can be concluded that Bayesian inference-based approaches are superior to approaches that are not based on Bayesian inference and reveal the degree of confidence in the predictions. In case of concentrating on approaches that rely on Bayesian inference, obtaining the best results depends greatly on choosing an appropriate value for the MCD.

The proposed Solar-OBNet's performance was compared with that of other SOTA methods introduced in [6, 45], as shown in Table 4. The SolNet model that was proposed in [6] is only able to detect the presence of dust on SP, as it was trained on a dataset containing images of clean SP and dusty SP. This model achieved accuracy, average precision, average recall, and average f1-score of 98.2%, 96.6%, 96.69%, and 96.12%, respectively. The proposed Solar-OBNet is superior to the SolNet model in its ability to detect the presence of dust and snow on SP and not just detect dust as is the case in the SolNet model. The stacked ensemble model proposed in [45] can detect the presence of four types of pollution on SP: dust, branch, leaves, and powder. Each model in this ensemble model was trained on a numerical dataset containing SP-specific features and weather-specific features. The stacked ensemble model presented in [45] outperformed the proposed Solar-OBNet in terms of its ability to detect a greater

Table 4 Comparison with SOTA models

Criterion	SolNet [6]	Stacking ensemble model [45]	Proposed Solar-OBNet
Data	Images	Numerical	Images
Pollution type that can be detected	Dust	Dust, branch, leaves, and powder	Dust, and snow
Accuracy	98.2%	97.37%	92.66%
Average precision	96.6%	90%	94.29%
Average recall	96.69%	78.2%	94.07%
Average f1-score	96.12%	83.2%	94.06%

number of pollutants that could be present on the surface of SP. However, the stacked ensemble model did not achieve high performance in terms of average precision, average recall, and average f1-score, as shown in Table 4.

7 Conclusion and future work

Renewable energy is an important source for meeting energy needs, in addition to being a clean source of energy through which the seventh goal of SDGs can be achieved. Solar energy is the most important source of renewable energy, and this energy is converted into electrical energy using SP. In order to obtain the largest amount of electrical energy from the SP, it must always be ensured that they are clean. However, environmental factors such as dust and snow impede the fulfillment of this condition. Therefore, SP are always cleaned periodically, but most of the methods used to clean SP at an affordable cost have negative effects either on human health or on the efficiency of SP. To lessen the negative effects of frequent cleaning operations, SP must be automatically cleaned only when necessary. Additionally, since any inaccurate prediction would result in a significant loss of energy, reliable methods must be used to identify dusty SP and snow-covered SP. This paper presents a reliable approach capable of detecting the presence of dust or snow on SP. The proposed Solar-OBNet, named Solar-OBNet, is based on a Bayesian CNN, and the Bayesian approximation was performed using the MCD method. In this method, a dropout layer is added to the CNN with an appropriate dropout rate, the network is trained, and then the dropout layer is activated during testing. To select the appropriate dropout rate, the HHO algorithm was used. In addition to the high balanced accuracy of the results of the proposed Solar-OBNet, which reached 94.07%, the proposed Solar-OBNet provides the quantity of confidence in its predictions. The proposed Solar-OBNet shows high confidence with correct predictions and high uncertainty warning with false predictions, which in this case requires human examination. The approach proposed in this paper helps to preserve the

efficiency of SP from some effects of environmental factors: dust and snow, but there are other environmental factors that can also significantly affect the efficiency of SP, such as temperature, shade, and leaves. Providing AI-based approaches to detect exposure of SP to these other environmental factors with high accuracy is also an important issue to be addressed in future studies. Future research directions could involve working on improving the proposed approach and applying it to real-world scenarios.

Funding Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB).

Data availability Data utilized in this paper is available at the following links: <https://www.kaggle.com/datasets/hemanthai7/solar-panel-dust-detection> <https://doi.org/10.5683/SP2/ITQPHZ>.

Declarations

Conflict of interest There is no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Ridha HM, Gomes C, Hizam H, Ahmadipour M, Heidari AA, Chen H (2021) Multi-objective optimization and multi-criteria decision-making methods for optimal design of standalone photovoltaic system: a comprehensive review. *Renew Sustain Energy Rev* 135:1364–2321. <https://doi.org/10.1016/j.rser.2020.110202>

2. Zhu B, Su B, Li Y (2018) Input–output and structural decomposition analysis of India's carbon emissions and intensity, 2007/08–2013/14. *Appl Energy* 230:1545–1556. <https://doi.org/10.1016/j.apenergy.2018.09.026>
3. Perera ATD, Nik VM, Chen D, Scartezzini JL, Hong T (2020) Quantifying the impacts of climate change and extreme climate events on energy systems. *Nat Energy* 5:150–159. <https://doi.org/10.1038/s41560-020-0558-0>
4. Pawluk RE, Chen Y, She Y (2019) Photovoltaic electricity generation loss due to snow—a literature review on influence factors estimation and mitigation. *Renew Sustain Energy Rev*. <https://doi.org/10.1016/j.rser.2018.12.031>
5. International Energy Agency (2017) Market Report Series: Renewables. <https://www.iaea.org/publications/renewables2017/>;2017. Accessed 5 May 2023
6. Onim MSH, Sakif ZMM, Ahnaf A, Kabir A, Azad AK, Oo AMT, Afreen R, Hridy ST, Hossain M, Jabid T et al (2023) SolNet: a convolutional neural network for detecting dust on solar panels. *Energies* 16(1):155. <https://doi.org/10.3390/en16010155>
7. Sangchul Oh (2019) Analytic and Monte-Carlo studies of the effect of dust accumulation on photovoltaics. *Sol Energy*. <https://doi.org/10.1016/j.solener.2019.07.011>
8. Ghosh A (2020) Soiling losses: a barrier for india's energy security dependency from photovoltaic power. *Challenges* 11:9
9. Benghanem M, Almohammadi A, Khan MT, Al-Masraqi A (2018) Effect of dust accumulation on the performance of photovoltaic panels in desert countries: a case study for Madinah, Saudi Arabia. *Int J Power Electron Drive Syst* 9:1356
10. Ullah A, Imran H, Maqsood Z, Butt NZ (2019) Investigation of optimal tilt angles and effects of soiling on PV energy production in Pakistan. *Renew Energy* 139:830–843
11. Paudyal BR, Shakya SR (2016) Dust accumulation effects on efficiency of solar PV modules for off grid purpose: a case study of Kathmandu. *Sol Energy* 135:103–110
12. Costa SC, Diniz ASA, Kazmerski LL (2018) Solar energy dust and soiling R&D progress: literature review update for 2016. *Renew Sustain Energy Rev* 82:2504–2536
13. Kazem HA, Chaichan MT, Al-Waeli AH, Sopian K (2020) A review of dust accumulation and cleaning methods for solar photovoltaic systems. *J Clean Prod* 276:123187
14. Ramli MA, Prasetyono E, Wicaksana RW, Windarko NA, Sedraoui K, Al-Turki YA (2016) On the investigation of photovoltaic output power reduction due to dust accumulation and weather conditions. *Renew Energy* 99:836–844
15. Kabir MA, Islam R, Nazifa S, Choudhury R (2021) Dust effect on photovoltaic output performance: comparative analysis and a case study in Dhaka, Bangladesh. In: *Proceedings of the 2021 international conference on automation, control and mechatronics for industry, Rajshahi, Bangladesh*, pp 1–6
16. Gholami A, Khazaee I, Eslami S, Zandi M, Akrami E (2018) Experimental investigation of dust deposition effects on photovoltaic output performance. *Sol Energy* 159:346–352
17. Kazem HA, Chaichan MT (2016) Experimental analysis of the effect of dust's physical properties on photovoltaic modules in Northern Oman. *Sol Energy* 139:68–80
18. Heidari N, Gwamuri J, Townsend T, Pearce JM (2015) Impact of snow and ground interference on photovoltaic electric system performance. *JPHOTOV* 5:1680–1685
19. Marion B, Schaefer R, Caine H, Sanchez G (2013) Measured and modeled photovoltaic system energy losses from snow for Colorado and Wisconsin locations. *Sol Energy* 97:112–121
20. Taylor MP (2003) Statistical relationship between photovoltaic generation and electric utility demand in Minnesota (1996–2002). In: *Proceedings of the solar 2003 conference*
21. Andrews RW, Pollard A, Pearce JM (2013) A new method to determine the effects of hydrodynamic surface coatings on the snow shedding effectiveness of solar photovoltaic modules. *Sol Energy Mater Sol Cells* 113:71–78
22. Sugiura T, Yamada T, Nakamura H, Umeiya M, Sakuta K, Kurakawa K (2003) Measurements, analyses and evaluation of residential PV systems by Japanese monitoring program. *Sol Energy Mater Sol Cells* 75:767–779
23. Matthews T (2016) Solar photovoltaic reference array report. Northern Alberta Institute of Technology, Edmonton
24. Powers L, Newmiller J, Townsend T (2010) Measuring and modeling the effect of snow on photovoltaic system performance
25. Townsend T, Powers L (2011) Photovoltaics and snow: an update from two winters of measurements in the SIERRA
26. Ryberg D, Freeman J (2017) Integration, validation, and application of a PV snow coverage model in SAM. National Renewable Energy Laboratory
27. Wirth G, Weigl T, Weizenbeck J, Zehner M, Schroedter-Homscheidt M, Becker G (2009) mapping of snow cover periods for yield assessment and dimensioning of PV Systems
28. Myyas REN, Al-Dabbasa M, Tostado-Véliz M, Jurado F (2022) A novel solar panel cleaning mechanism to improve performance and harvesting rainwater. *Sol Energy* 237:19–28
29. Ozturk O, Hangun B, Eyecioglu O (2021) Detecting snow layer on solar panels using deep learning. In: *10th International conference on renewable energy research and application*. <https://doi.org/10.1109/ICRERA52334.2021.9598700>
30. Li X, Zhang W, Ma H, Luo Z, Li X (2020) Domain generalization in rotating machinery fault diagnostics using deep neural networks. *Neurocomputing* 403:409–420
31. Abdar M, Samami M, Mahmoodabad SD, Doan T, Mazouze B, Hashemifesharaki R, Liu L, Khosravi A, Acharya UR, Makarenkov V, Nahavandi S (2021) Uncertainty quantification in skin cancer classification using three-way decision-based Bayesian deep learning. *Comput Biol Med* 135:104418
32. Junhwan C, Seokmin O, Joongmoo B (2022) Uncertainty estimation in AVO inversion using Bayesian dropout based deep learning. *J Pet Sci Eng* 208:109288. <https://doi.org/10.1016/j.petrol.2021.109288>
33. Abdar M, Pourpanah F, Hussain S, Rezazadegan D, Liu L, Ghavamzadeh M, Fieguth P, Cao X, Khosravi A, Acharya UR, Makarenkov V (2021) A review of uncertainty quantification in deep learning: techniques, applications and challenges. *Inf Fusion* 76:243–297. <https://doi.org/10.1016/j.inffus.2021.05.008>
34. Gal Y, Ghahramani Z (2016) Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In: *International conference on machine learning*, pp 1050–1059
35. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2017) Grad-CAM visual explanations from deep networks via gradient-based localization. In: *Proceedings of the IEEE international conference on computer vision*, pp 618–626
36. Proietti A, Panella M, Leccese F, Svezia E (2015) Dust detection and analysis in museum environment based on pattern recognition. *Measurement* 66:62–72
37. Maitre J, Bouchard K, Bédard LP (2019) Mineral grains recognition using computer vision and machine learning. *Comput Geosci* 130:84–93
38. Igathinathane C, Melin S, Sokhansanj S, Bi X, Lim CJ, Pordesimo LO, Columbus EP (2009) Machine vision based particle size and size distribution determination of airborne dust particles of wood and bark pellets. *Powder Technol* 196(2):202–212
39. Ayyagari KS, Munian Y, Inupakutika D, Reddy BK, Gonzalez R, Alamaniotis M (2022) Simultaneous detection and classification of dust and soil on solar photovoltaic arrays connected to a large-scale industry: a case study. In: *18th International conference on European energy market (EEM)*. IEEE, pp 1–6
40. Aman R, Rizwan M, Kumar A (2023) Fault classification using deep learning based model and impact of dust accumulation on

- solar photovoltaic modules. *Energy Sources Part A Recovery Util Environ Eff* 45(2):4633–4651
41. Fan S, Wang Y, Cao S, Zhao B, Sun T, Liu P (2022) A deep residual neural network identification method for uneven dust accumulation on photovoltaic (PV) panels. *Energy* 239:122302
 42. Maity R, Shamaun Alam M, Pati A (2020) An approach for detection of dust on solar panels using CNN from RGB dust image to predict power loss. *Cognit Comput Hum Cognit Perspect Appl*. <https://doi.org/10.1007/978-3-030-48118-6>
 43. Zyout I, Oatawneh A (2020) Detection of PV solar panel surface defects using transfer learning of the deep convolutional neural networks. In: 2020 Advances in science and engineering technology international conferences (ASET). IEEE, pp 1–4
 44. Mehta S, Azad AP, Chemmengath SA, Raykar V, Kalyanaraman S (2018) DeepSolarEye: power loss prediction and weakly supervised soiling localization via fully convolutional networks for solar panels. In: IEEE winter conference on applications of computer vision (WACV), Lake Tahoe, pp 333–342. <https://doi.org/10.1109/WACV.2018.00043>
 45. Khan PW, Byun YC, Jeong OR (2023) A stacking ensemble classifier-based machine learning model for classifying pollution sources on photovoltaic panels. *Sci Rep* 13(1):10256
 46. AlZoubi A, Lu F, Zhu Y, Ying T, Ahmed M, Du H (2024) Classification of breast lesions in ultrasound images using deep convolutional neural networks: transfer learning versus automatic architecture design. *Med Biol Eng Comput* 62(1):135–149
 47. Patel R, Thakkar P, Ukani V (2024) CNNRec: convolutional neural network based recommender systems—a survey. *Eng Appl Artif Intell* 133:108062
 48. Kumar GSC, Kumar RK, Kumar KPV, Sai NR, Brahmaiah M (2024) Deep residual convolutional neural network: an efficient technique for intrusion detection system. *Expert Syst Appl* 238:121912
 49. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708
 50. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H (2017) Mobilenets efficient convolutional neural networks for mobile vision applications. *arXiv preprint* [arXiv:1704.04861](https://arxiv.org/abs/1704.04861)
 51. Tan M, Le Q (2019) EfficientNet: rethinking model scaling for convolutional neural networks. In: International conference on machine learning. PMLR, pp 6105–6114
 52. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. *arXiv preprint* [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
 53. Han Q, Qian X, Xu H, Wu K, Meng L, Qiu Z, Weng T, Zhou B, Gao X (2024) DM-CNN: dynamic multi-scale convolutional neural network with uncertainty quantification for medical image classification. *Comput Biol Med* 168:107758
 54. Hamedani-KarAzmoddehFar F, Tavakkoli-Moghaddam R, Tajally AR, Aria SS (2023) Breast cancer classification by a new approach to assessing deep neural network-based uncertainty quantification methods. *Biomed Signal Process Control* 79:104057
 55. Tai Y, Tan Y, Zou E, Lei B, Fan Q, He Y (2024) Where to model the epistemic uncertainty of Bayesian convolutional neural networks for classification. *Neurocomputing*. <https://doi.org/10.1016/j.neucom.2024.127568>
 56. Stark T, Wurm M, Zhu XX, Taubenbock H (2024) Quantifying uncertainty in slum detection: advancing transfer-learning with limited data in noisy urban environments. *IEEE J Sel Top Appl Earth Obs Remote Sens*. <https://doi.org/10.1109/JSTARS.2024.3359636>
 57. Heidari AA, Mirjalili S, Faris H, Aljarah I, Mafarja M, Chen H (2019) Harris Hawks optimization: algorithm and applications. *Futur Gener Comput Syst* 97:849–872
 58. Abdel-Basset M, Ding W, El-Shahat D (2021) A hybrid Harris Hawks optimization algorithm with simulated annealing for feature selection. *Artif Intell Rev* 54:593–637
 59. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks, vol 4. IEEE, pp 1942–1948
 60. Mirjalili S (2015) Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. *Knowl Based Syst* 89:228–249
 61. Abdel-Basset M, Shawky LA (2019) Flower pollination algorithm: a comprehensive review. *Artif Intell Rev* 52:2533–2557
 62. Costache R, Pal SC, Pande CB, Islam ARMT, Alshehri F, Abdo HG (2024) Flood mapping based on novel ensemble modeling involving the deep learning, Harris Hawk optimization algorithm and stacking based machine learning. *Appl Water Sci* 14(4):78
 63. Hadadi S, Arabani SP (2024) A novel approach for Parkinson's disease diagnosis using deep learning and Harris Hawks optimization algorithm with handwritten samples. *Multimed Tools Appl*. <https://doi.org/10.1007/s11042-024-18584-3>
 64. Dangi R, Lalwani P (2023) Harris Hawks optimization based hybrid deep learning model for efficient network slicing in 5g network. *Clust Comput* 27(1):1–15
 65. <https://www.kaggle.com/datasets/hemanthsai7/solar-panel-dust-detection>
 66. <https://borealiSData.ca/dataset.xhtml?persistentId=10.5683/SP2/ITQPHZ>
 67. Ezzat D, Hassanien AE (2023) Optimized Bayesian convolutional neural networks for invasive breast cancer diagnosis system. *Appl Soft Comput* 147:110810
 68. Glorot X, Bordes A, Bengio Y (2011) Deep sparse rectifier neural networks. In: Proceedings of the 14th international conference on artificial intelligence and statistics, Fort Lauderdale, pp 315–323
 69. Dozat T (2016) Incorporating Nesterov momentum into Adam
 70. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. *arXiv:1412.6980*
 71. Prechelt L (2002) Early stopping-but when? Neural networks: tricks of the trade. Springer, Berlin
 72. Senior A, Heigold G, Ranzato M, Yang K (2013) An empirical study of learning rates in deep neural networks for speech recognition. In: Proceedings of IEEE international conference on acoustics, speech and signal processing, pp 6724–6728