

RESEARCH ARTICLE

An Ensemble-Based Deep Learning Framework for Efficient Soiling Detection on Photovoltaic Panels

Musa Balcı^{ID}, Andaç Fındıkçı^{ID}, Mustafa Yasin Erten^{ID}, Hüseyin Aydılek^{ID}

Department of Electrical and Electronics Engineering, Kırıkkale University Faculty of Engineering and Natural Sciences, Kırıkkale, Türkiye

Cite this article as: M. Balcı, A. Fındıkçı, M. Y. Erten, and H. Aydılek, "An ensemble-based deep learning framework for efficient soiling detection on photovoltaic panels," *Turk J Electr Power Energy Syst*. Published online November 17, 2025. doi: 10.5152/tepes.2025.25020.

ABSTRACT

Solar energy plays a pivotal role in renewable energy systems; however, dust accumulation on photovoltaic panels substantially reduces energy production efficiency. Manual cleaning methods at large-scale plants are costly and impractical, highlighting the need for automated detection techniques. This study presents a novel image processing and deep learning-based approach to accurately detect dusty PV panels. Images underwent preprocessing, including Hue, Saturation, Value color space conversion, and morphological operations to precisely segment dust-affected regions. Individual performances of DenseNet169, Xception, and InceptionV3 models were evaluated, and an ensemble model—Deep Solar Ensemble—was developed via soft voting. Experimental results demonstrated that the proposed ensemble achieved a superior classification accuracy of 97.02%, a precision of 97.29%, a recall of 96.56%, and an F1 score of 96.92% on a binary classification task. To address real-world applicability and robustness, the study was extended to include comparisons with lightweight architectures and testing on a more diverse, multi-class dataset containing various fault types, where the ensemble continued to show robust performance. The proposed methodology offers significant potential for automating solar panel maintenance, thereby enhancing operational efficiency, while also considering the trade-offs between accuracy and computational cost for practical deployment.

Index Terms—DenseNet, dust classification, ensemble learning, InceptionV3, photovoltaic panel, Xception

I. INTRODUCTION

In the contemporary world, energy has become a critical necessity due to the growing human population, industrial competition, and the continuous advancement of technology. Energy constitutes a cornerstone of both societal and individual life. It is indispensable for the continuity of industry, agriculture, transportation, technology, and daily household activities. However, approximately 80% of the energy currently utilized is still derived from fossil fuels. The formation of fossil fuels through geological processes spanning millions of years results in their limited reserves. Factors such as finite reserves and the environmental harm caused by their use render fossil fuels unsustainable energy options [1, 2]. The escalating environmental degradation and the consequent issue of climate change have compelled humanity to seek alternative energy sources. Consequently, there has been a shift toward renewable energy sources. Solar, wind, hydroelectric, and geothermal energy sources are advancing and gaining prominence in the energy sector. Among these, solar energy is one of the most frequently preferred energy sources. Despite its significant position among sustainable

energy sources, the effective utilization of solar panels faces certain challenges. In particular, the accumulation of dust and dirt reduces the capacity of photovoltaic (PV) cells to absorb sunlight, thereby decreasing energy efficiency. Additionally, determining the optimal timing for solar panel maintenance enhances energy production efficiency and extends the lifespan of the panels. Onim and colleagues developed the SolNet model, which employs a deep learning-based approach for the early detection of dust accumulation. Utilizing a Convolutional Neural Network (CNN) architecture, the SolNet model detects dust levels on solar panel images, offering recommendations that could increase panel efficiency by 15% [3]. Malik and colleagues [4] implemented an Arduino-based system for the automatic detection of dust accumulation and cloud-based reporting. This system analyzes data from dust sensors to automatically assess the cleanliness of panels, reducing energy production losses by 25%. Abuqaoud and Ferrah [5] developed an innovative sensor-based approach to detect dust and soil on solar panels. This method, employing optical sensors, achieved 90% accuracy in detecting dust accumulation, significantly reducing efficiency losses. Saquib and colleagues [6]

Corresponding author: Mustafa Yasin Erten, mustafaerten@kku.edu.tr



Content of this journal is licensed under a Creative Commons Attribution (CC BY) 4.0 International License.

Received: June 10, 2025
Revision Requested: June 23, 2025
Last Revision Received: July 7, 2025
Accepted: August 5, 2025
Publication Date: November 17, 2025

utilized image processing and artificial neural networks (ANNs) to detect dust accumulation and predict energy production. Their study classified dust density on panels with 88% accuracy and estimated annual energy losses at 20%. Keerthana and colleagues [7] proposed an effective model for the early detection of dust accumulation using image processing techniques. Employing methods such as histogram equalization and edge detection, this model achieved 85% accuracy. Abukhait [8] developed a computer vision-based approach to enable more precise detection of dust accumulation. This approach utilized transfer learning with the ResNet50 model to analyze dust levels, achieving 92% accuracy. Kavya and Keshav [9] proposed a low-cost system for dust detection, offering an optical sensor-based solution that reduced energy losses by 30%. Alatwi and colleagues [10] presented an effective solution for sustainable energy production using deep learning techniques. Their study achieved 86.79% accuracy with the DenseNet169 model, with further improvements through integration with a Support Vector Machine (SVM). In contrast to these studies, the present work preprocessed solar panel images using image processing techniques to segment dusty areas and employed various deep learning models. The preprocessing steps include Hue, Saturation, Value (HSV) color space transformation and morphological operations to ensure precise detection of dusty regions. The performance of DenseNet169, Xception, and InceptionV3 models was evaluated individually, and these models were combined using an ensemble method to propose the Deep Solar Ensemble model. Experimental results demonstrated that the DenseNet169 model achieved 96.12% accuracy, the Xception model 91.67% accuracy, and the InceptionV3 model 94.64% accuracy, while the Deep Solar Ensemble model exhibited the highest performance with 97.02% accuracy, 97.29% precision, 96.56% recall, and 96.92% F1 score. To further assess the model's real-world viability, this work also evaluates its robustness on a challenging multi-class dataset

and compares its performance against lightweight architectures, addressing the critical trade-off between accuracy and computational efficiency. These results indicate that the image processing and ensemble approach surpasses the performance of individual models, offering a higher success rate in detecting dust accumulation on solar panels.

II. METHODS

This study utilized two distinct datasets. The first dataset, sourced from a publicly available repository on Kaggle, comprises solar panel images categorized into two classes: clean and dirty. This dataset contains images specifically curated for binary classification tasks to distinguish between clean and dusty solar panels, capturing various environmental conditions and dust accumulation levels. The second dataset was introduced to address a more complex, multi-class classification problem. Its objective is to detect a wider range of surface anomalies, containing six classes: Clean (194 images), Dusty (191 images), Bird-drop (192 images), Electrical-damage (104 images), Physical-Damage (70 images), and Snow-Covered (124 images). Both datasets include a balanced collection of images, with approximately equal representation across classes, ensuring robust training and evaluation of classification models. The distribution of images in the datasets is illustrated in Fig. 1. Multiple deep learning models were employed to assess their performance. To address potential class imbalances, the weighted class balancing method was applied to ensure equitable representation of all classes in both datasets. The balanced datasets were subjected to preprocessing steps, including image resizing and normalization techniques to standardize input dimensions and pixel intensities for model training. Additionally, to enhance the detection of dusty regions, image processing techniques were applied, involving HSV color space transformation to emphasize dust-related color and brightness variations, followed by morphological operations to refine the segmentation of dusty areas. The preprocessed datasets were used to train the DenseNet169, InceptionV3, and Xception image classification algorithms. The trained models were combined using ensemble learning to form the Deep Solar Ensemble Learning model. The soft voting method

Main Points

- Introduction of Deep Solar Ensemble: A novel ensemble model combining DenseNet169, Xception, and InceptionV3 enhances dust detection on solar panels.
- Superior performance metrics: The Deep Solar Ensemble achieves high accuracy (97.02%), surpassing individual deep learning models.
- Advanced image preprocessing: Hue, Saturation, Value color space transformation and morphological operations improve the segmentation of dusty regions.
- Robustness validation on a multi-class dataset: The framework's effectiveness was further tested on a diverse dataset with six fault classes (e.g., dust, snow, physical damage), demonstrating its robustness in more complex, real-world scenarios.
- Comparative analysis for practical deployment: A comparison with lightweight models like MobileNet was conducted to analyze the critical trade-off between classification accuracy and computational efficiency for real-time applications.
- Practical application: The model enables efficient solar panel maintenance, reducing energy losses due to dust accumulation.

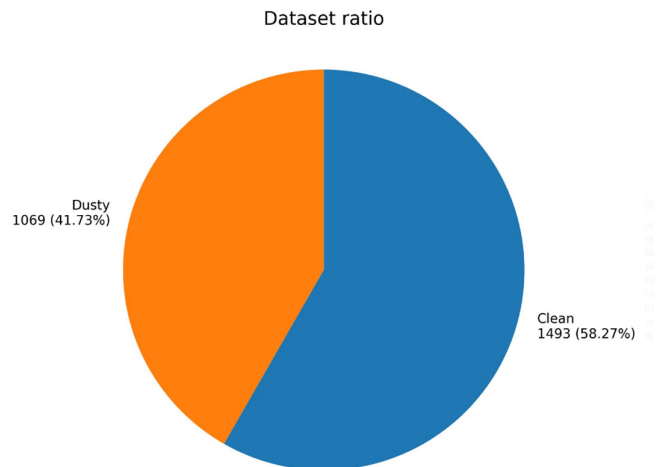


Fig. 1. First dataset ratio.

was adopted as the ensemble learning technique, and the performance of the ensemble model was evaluated on the test datasets. Furthermore, for a comprehensive comparison, the MobileNet and VGG19 models were also trained and evaluated on the same datasets. These models were not included in the ensemble but served as benchmarks to gauge the performance of the primary models. The general workflow of this study is summarized in Fig. 2. Performance evaluation metrics, including Accuracy, Recall, Precision, and F1 score, were employed to assess the individual classifiers, the comparison models, and the ensemble model. The results were analyzed to elucidate the advantages provided by the ensemble learning models and other classifiers.

A. Image Preprocessing

The dataset was subjected to preprocessing steps to standardize input dimensions and enhance the detection of dusty regions. These steps included image resizing and normalization techniques to ensure consistent pixel intensities and input dimensions for model training. Images were resized to a resolution of 224×224 pixels to align with the requirements of deep learning models, and pixel intensities were normalized to the range $[0, 1]$. A representative clean solar panel image from the dataset is shown in Fig. 3. To address potential class imbalances, the weighted class balancing method was applied to ensure equitable representation of clean and dirty panel images during model training. This method assigns higher weights to underrepresented classes in the loss function, thereby mitigating bias toward the majority class and improving model performance on minority classes. Additionally, a series of image processing techniques was applied to highlight color and brightness variations associated with dust and to improve the segmentation of dusty areas. Images were initially converted from Blue, Green, Red (BGR) to HSV color space, as the HSV color space better distinguishes the color

and brightness characteristics of dust. For the detection of dusty areas, a color range was defined in the HSV space with a lower bound of $[H = 0, S = 0, V = 100]$ and an upper bound of $[H = 180, S = 50, V = 200]$, and a binary mask was generated for pixels falling within this range. This mask designates pixels representing dusty regions as white (255) and others as black (0). To enhance the accuracy of the mask, morphological operations were applied; a 5×5 square kernel was used to perform an opening operation (morphological opening), which eliminated small noise artifacts. Subsequently, a closing operation (morphological closing) with the same kernel was applied to fill small gaps within dusty regions. For visualization of dusty areas, the masked regions were highlighted in red (BGR: $[0, 0, 255]$) and overlaid onto the original image with a transparency factor of 0.5. The effectiveness of these preprocessing steps in segmenting dusty regions is illustrated with dirty solar panel images and their processed states in Fig. 4. These preprocessing steps enabled precise segmentation of dusty regions, thereby enhancing the classification performance of deep learning models.

Fig. 4 is a visualization of the dirty solar panel image (left) and the resulting image after applying the preprocessing (right). The dusty regions, determined through the above HSV color space conversion, morphological processes, and mask procedures, are highlighted in red and overlaid on the original image for the model to more easily recognize the regions.

B. Densenet169

DenseNet169 is a variation of the DenseNet (Densely Connected Convolutional Networks) architecture, which is a noteworthy contribution to the CNN community. Architecturally, DenseNet169 has 169 layers and accepts 224×224 pixel RGB images as input. This is followed by convolution and pooling layers, four dense blocks,

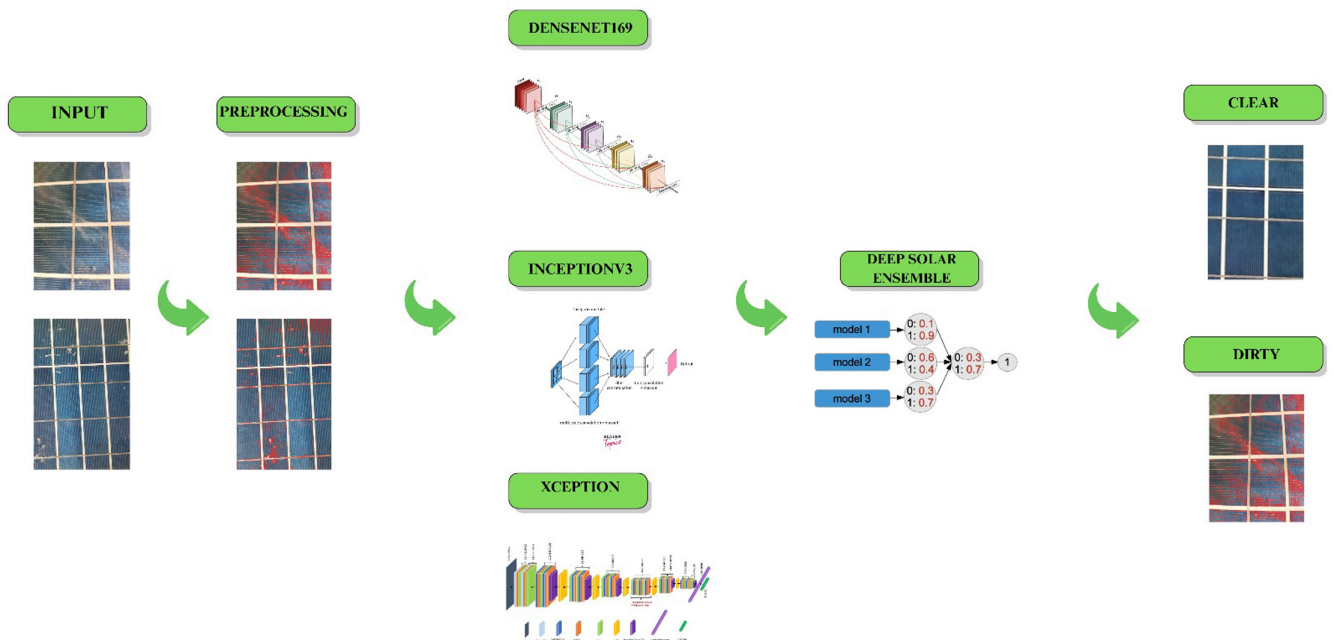


Fig. 2. Workflow of study.

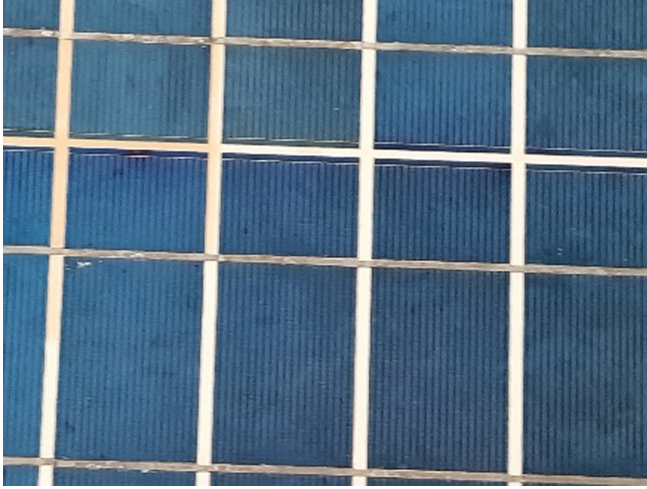


Fig. 3. Clean solar panel.

and three transition layers, decreasing dimensionality between the blocks. Then the classification is accomplished through global average pooling, a fully connected layer, and a Softmax activation function. The DenseNet169 architecture is illustrated in Fig. 5. The most important feature of DenseNet169 is that each layer takes the output of the previous layers as input. This dense connection pattern has two significant advantages: First, it is parameter-efficient since the feature maps are recycled throughout the network. Second, it improves the gradient flow, which essentially diminishes the vanishing gradient problem that is common in deep networks. These features are what make DenseNet169 highly effective at discovering irregular and fine-grained patterns such as dust, dirt, or damage on solar panels. Its ability to learn complex features with fewer parameters allows it to detect surface defects with high accuracy. Therefore, DenseNet169 was selected as part of the base ensemble due to these inherent architectural advantages [11-13].

C. Xception

Xception is a CNN architecture designed by François Chollet, which builds upon the concepts of the Inception model by using more efficient depthwise separable convolutions. The Xception architecture

consists of 71 layers in 14 modules with approximately 23 million parameters. It consumes 224×224 pixel Red, Green, Blue (RGB) input images, which are fed into 36 depthwise separable convolutional layers. Max pooling or strided convolutions are used between modules to downsample the feature map. The architecture concludes with global average pooling and a fully connected layer for classification. The Xception architecture is depicted in Fig. 6. The two building blocks of the model are the depthwise separable convolutions, which decompose an ordinary convolution into two operations: a depthwise operation where one filter convolves all the input channels, and a pointwise operation (a 1×1 convolution) that adds the outputs. This design is particularly helpful because it allows the model to learn high-level spatial hierarchies from panel images—encoding features from low-level edges to complex anomaly shapes—using many fewer parameters. This equates to faster training and inference speeds without a noticeable decrease in performance. Xception was added to the ensemble to take advantage of its newer and highly optimized architecture, providing a strong yet computationally light method. Its design philosophy enables a different and complementary feature extraction approach compared to DenseNet, which was the primary reason for its inclusion in the framework [14-16].

D. InceptionV3

InceptionV3 is a CNN architecture introduced by Christian Szegedy et al., with a focus on creating a deeper and wider network. Its core building blocks are Inception modules, where parallel convolution and pooling operations of different sizes (e.g., 1×1 , 3×3 , 5×5) are executed and combined. One of the innovations of the InceptionV3 architecture is the factorization of larger convolutions—for instance, replacing a 7×7 convolution with two consecutive 3×3 convolutions—to reduce computational cost while maintaining a large receptive field. The model consists of 48 layers and approximately 24 million parameters and accepts RGB images of 224×224 pixels as input. These design choices allow the network to learn effectively across multiple scales and dimensions, improving both performance and efficiency. The architecture of InceptionV3 is shown in Fig. 7. This two-stream, multi-scale processing approach is particularly effective for this task, as defects on solar panels can vary significantly in size—from tiny bird droppings and minute cracks to large clumps of snow or dust. The ability of InceptionV3 to detect both

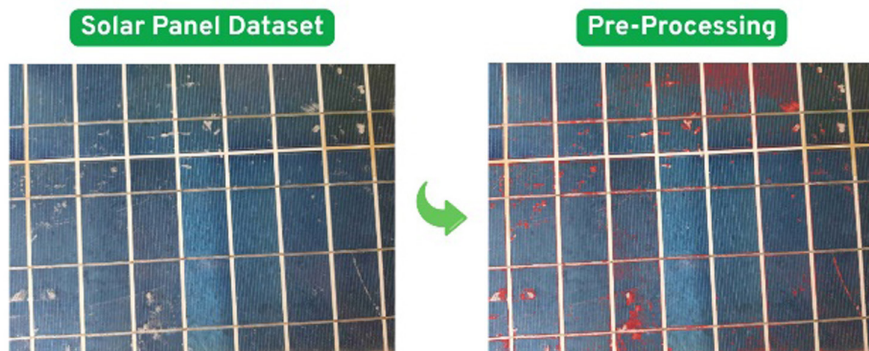


Fig. 4. Dirty solar panel and its processed state.

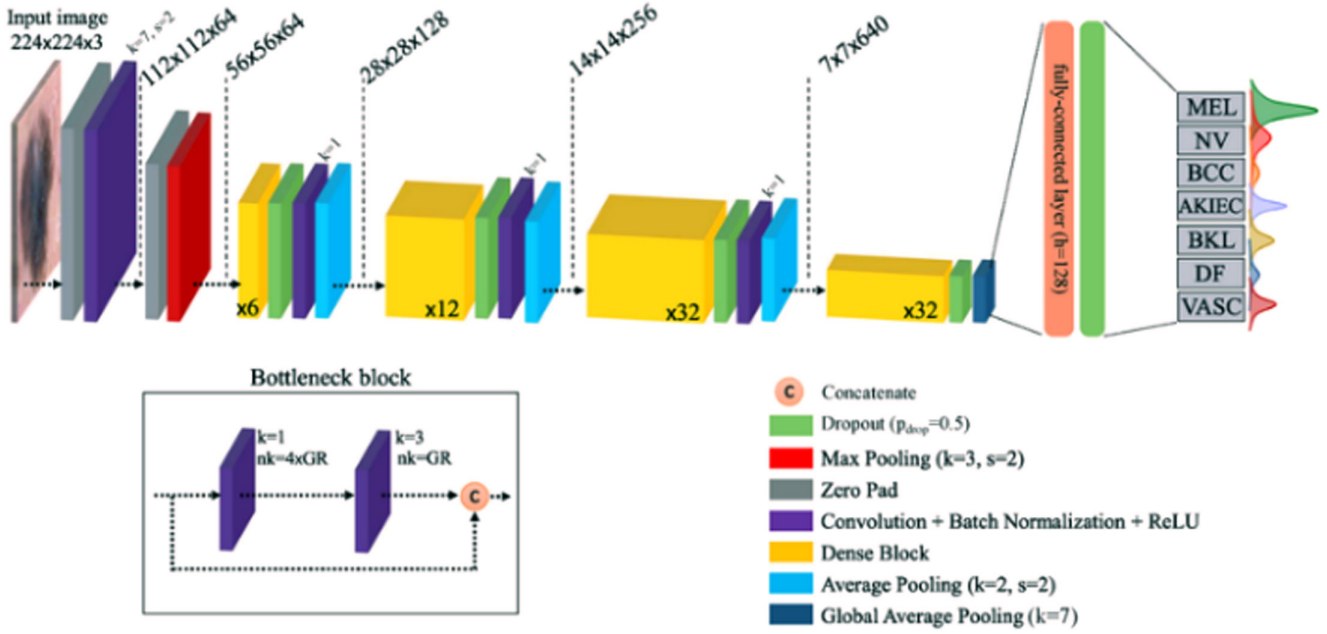


Fig. 5. Structure of DenseNet169 [12].

fine-grained and coarse features simultaneously makes the overall ensemble more versatile and robust in identifying a wide range of anomalies. InceptionV3 was chosen as the third member of the ensemble to introduce this multi-scale processing capability, offering a complementary perspective to DenseNet's feature reuse and Xception's computational efficiency [17-19].

E. Ensemble Learning

Ensemble learning is a powerful approach widely used in machine learning and deep learning. This method aims to combine the predictions of multiple models to surpass the performance of a single model. Ensemble learning is based on the idea that the errors of different models will decrease on average, and thus the model will

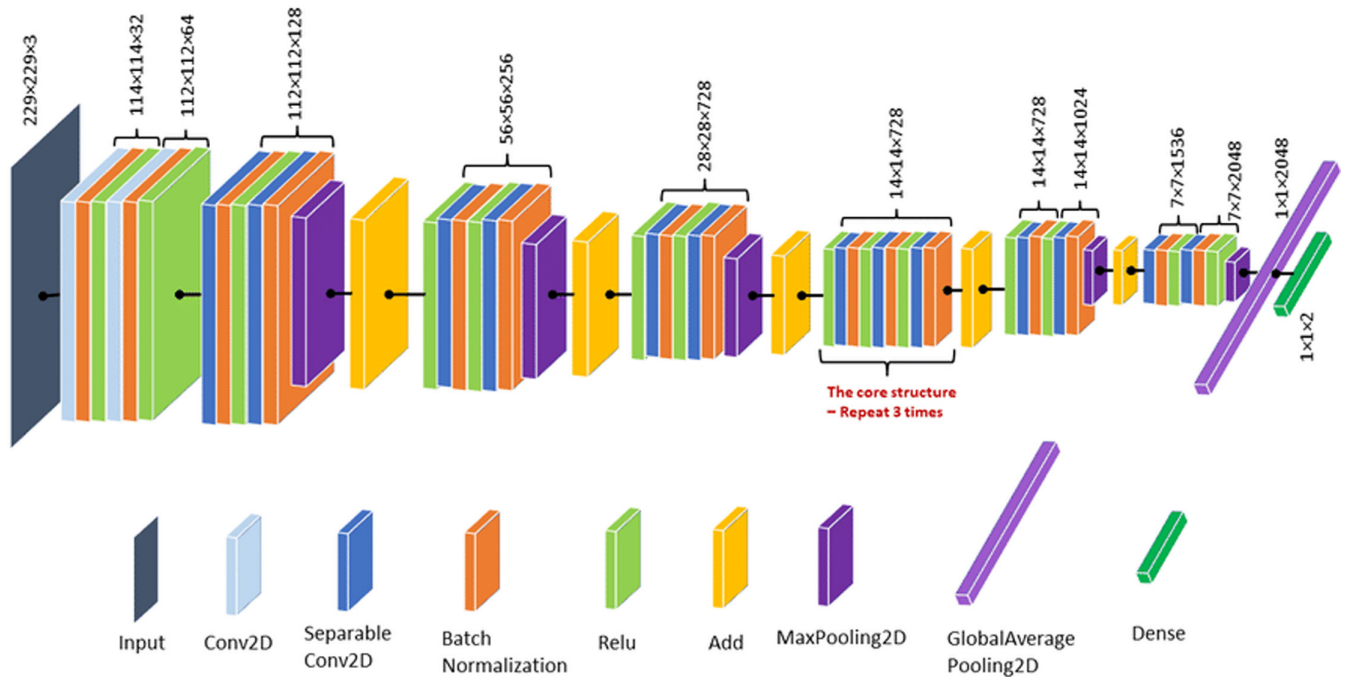


Fig. 6. Structure of Xception [14].

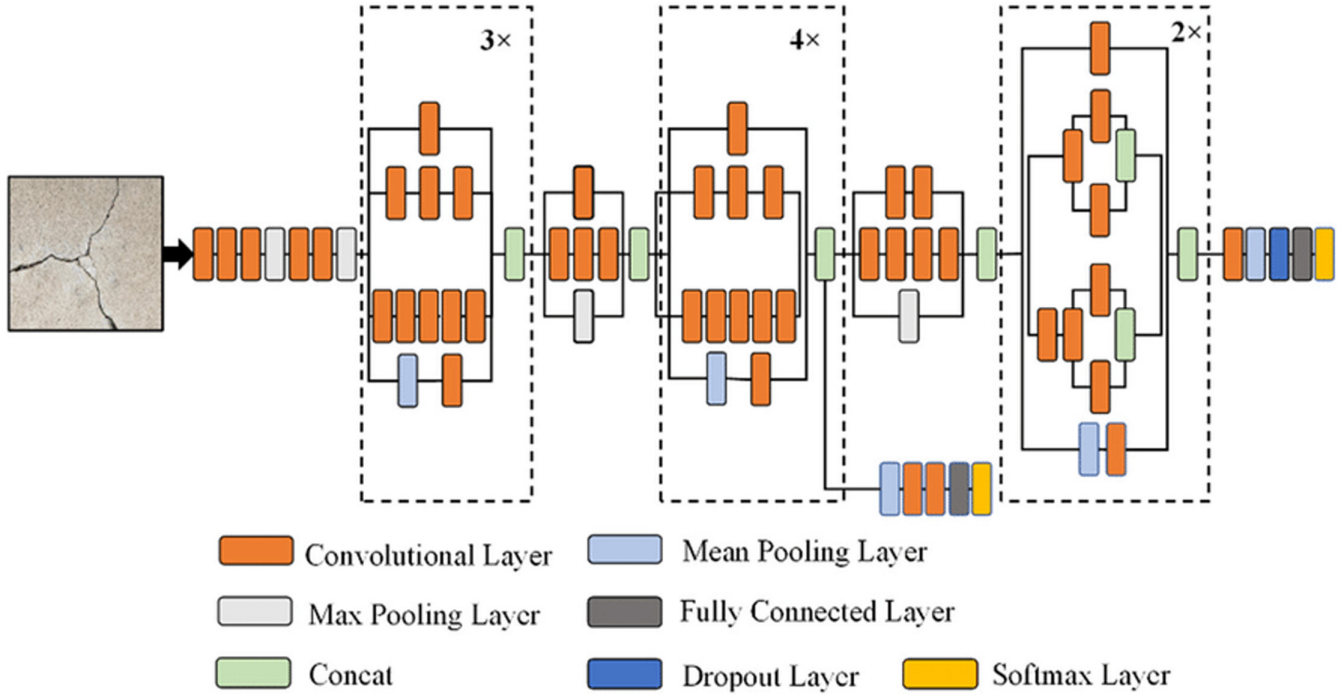


Fig. 7. Structure of InceptionV3 [17].

have higher performance, lower variance, and better generalization ability. These methods can include various machine learning and deep learning models. The main advantage of this approach is that the error of one model can be compensated for other models. In this way, problems such as overfitting and underfitting can be overcome. Ensemble learning reduces the negative impact of such errors by using more than one model to prevent artificial intelligence models from overfitting the variables in the data set and reducing the generalization ability. Ensemble learning methods also have some disadvantages. These methods can be complex, and this can make the model difficult to manage and understand. Furthermore, the computational power, cost, and time required to train multiple models and to build the final prediction model are some of the major disadvantages of ensemble learning models [20, 21].

F. Soft Voting

Soft voting, one of the ensemble learning methods, is a technique that aims to obtain a final prediction by combining the probability estimates of multiple models. Each model estimates the probability of belonging to a class. Then, the probability estimates of all models are summed and averaged on a class basis. The class with the highest average probability is selected as the final prediction. The effectiveness of soft voting depends on the diversity and quality of the models in the ensemble. The use of different algorithms and hyperparameters can improve the performance of soft voting [22-25]. The formula of the soft-voting method is given in (1).

$$y_m = \operatorname{argmax} \sum_{j=1}^{|J|} P_j(y = j | x_m) \cdot w_j \quad (1)$$

- y_m : The sequence number of the class predicted by the ensemble model.
- argmax : It is the process by which the index corresponding to the highest value of a function is determined.
- $P_j(y = j | x_m)$: It is the probability value expressing the probability that the sample x_m belongs to the j th class.
- j : Indicates the sequence number of each model or unit in the community model.
- w_j : j . represents the additive weight of the unit or model.

G. Deep Solar Ensemble Learning

The Deep Solar Ensemble Learning model was developed on the principle that a collection or ensemble of predictions from structurally diverse models leads to a more robust and precise classifier than any single one of them. The model was developed to detect dust accumulation and other surface anomalies on solar panels with high precision. The model strategically integrates the three above algorithms to benefit from their complementary strengths:

- DenseNet169 provides deep feature extraction with effective feature reuse.
- Xception provides a computationally lightweight and effective perspective with its depthwise separable convolutions.
- InceptionV3 provides the necessary multi-scale analysis for anomaly detection at varying sizes.

These three models were trained individually and then combined into an ensemble using the soft voting method. In this approach, the final prediction is made by averaging the probability scores of all the models, in a way that allows the models to 'vote' with their

confidence levels. This process prevents the risk of misclassification because of a certain weakness of any one model, leading to improved generalization. The ensemble performed wonderfully on the binary test data with 97.02% accuracy, 97.29% precision, 96.56% recall, and a 96.92% F1 score. The architecture of the Deep Solar Ensemble is illustrated in Fig. 8.

H. Performance Evaluation Metrics

It is a table used to evaluate the performance of a classification model. This matrix summarizes the number of correct and incorrect classifications by comparing the actual class labels with the model's predictions. The size of the matrix depends on the number of classes, with rows representing actual classes and columns representing predicted classes. Cells on the diagonal of the matrix indicate the number of correct classifications. Cells off-diagonal indicate the number of misclassifications. The confusion matrix allows the calculation of many metrics used to evaluate the performance of the model. These metrics include accuracy, precision, recall, and F1 score. Fig. 9 shows an example of a confusion matrix for the classification model, illustrating true positive, false negative, true negative, and false positive classifications [26-28].

- True Positive (TP): Indicates the number of instances that the model classifies as positive and are actually positive. In other words, these are the instances that the model correctly assigns to the positive class.
- False Positive (FP): Indicates the number of instances that the model classifies as positive but is actually negative. These are negative samples that the model incorrectly includes in the positive class.
- True Negative (TN): Represents the number of instances that the model classifies as negative but are actually negative. In other words, these are the samples that the model correctly assigned to the negative class.

- False Negative (FN): Represents the number of instances that the model classifies as negative but are actually positive. This is the number of positive instances that the model incorrectly includes in the negative class.

1) Accuracy:

Accuracy is a metric that measures the proportion of correct predictions of a classification model. It is calculated as the number of correctly classified samples divided by the total number of samples. A high accuracy value indicates that the model has a good classification ability. However, accuracy has some limitations. Especially in imbalanced datasets, accuracy can give misleading results. In imbalanced datasets, the model may achieve a high accuracy by correctly predicting the samples belonging to the majority class, but may not correctly classify the samples in the minority class [27, 28]. Equation (2) shows the accuracy metric formula.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

2) Precision:

Precision is a metric that measures the ratio of correct positive predictions of a classification model to all positive predictions. In other words, it shows how many of the samples that the model predicts as positive classes are actually positive. It is important when the cost of false positives is high. However, precision also has some limitations. A low precision value indicates that the model predicts many false positives. However, this does not provide information about whether the model makes a small number of true positive predictions [27, 28]. Equation (3) shows the formula for the precision metric.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3)$$

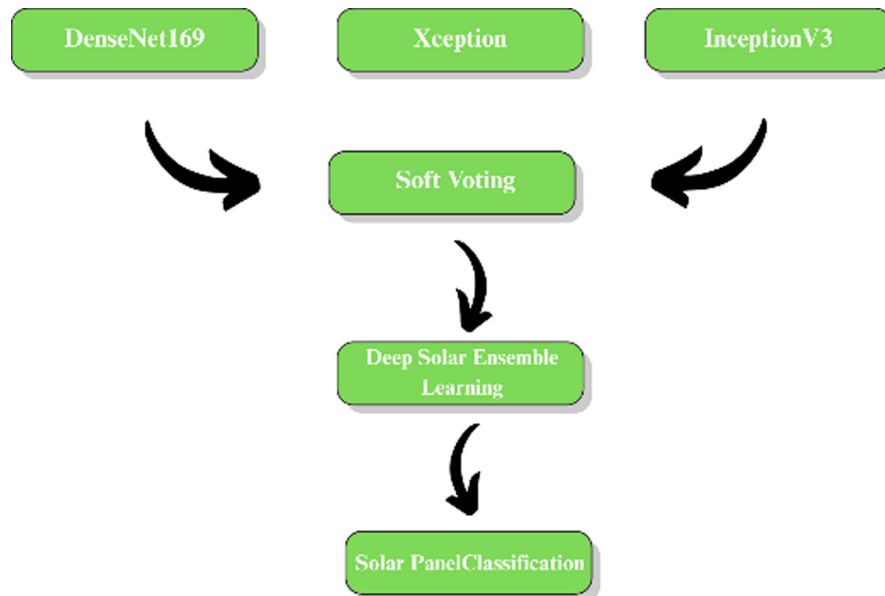


Fig. 8. Deep Solar Ensemble Learning.

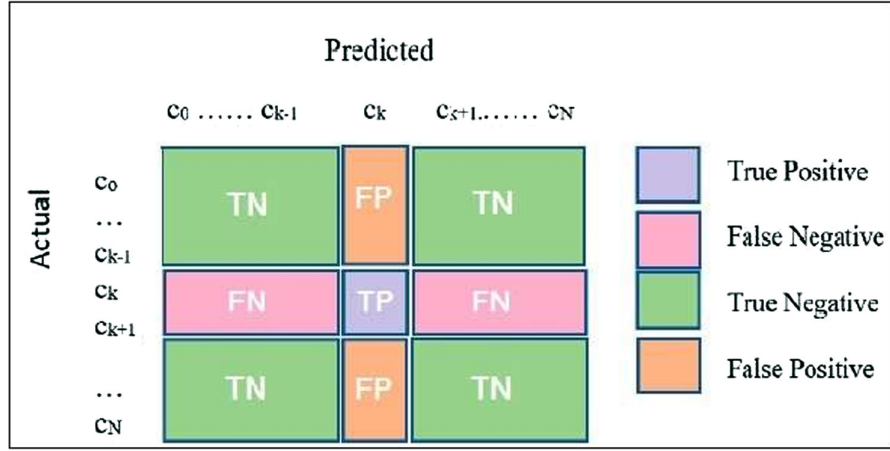


Fig. 9. Confusion matrix example [26].

3) Recall:

Recall measures the ratio of correctly classified instances of the positive class to all positive instances. In other words, it shows how well the model can recognize the positive class. Recall plays an important role, especially in scenarios where false negatives are important. However, the disadvantage of recall is that it does not take into account the false-positive predictions of the model, so it is not sufficient to evaluate the accuracy of the model alone [27, 28]. Equation (4) shows the formula for the recall metric.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4)$$

4) F1-Score:

The F1 score provides a balance by taking the harmonic mean of precision and recall values. In unbalanced data sets, misleading results can be obtained when using precision or recall alone, but the F1 score provides a more consistent performance evaluation by evaluating these two metrics together. It is particularly useful when the data set is unbalanced. By balancing both precision and recall, it provides a fairer measure of the overall success of the classification model [27, 28]. Equation (5) shows the formula for the F1 score metric.

$$\text{F1-Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

III. RESULTS

In this study, a dataset of dirty and clean solar panel images obtained from Kaggle was used to classify solar panels, and a second dataset was used for comparison. In the pre-processing stage, the data were normalized to a \$224 \times 224\$ pixel size in the range of [0,1] and converted from BGR to HSV color space. In addition, dusty areas were masked according to the color range determined in the HSV space, and these masks were improved with morphological opening/closing operations. After these steps, the weighted class method was applied to balance the classes, followed by separate training with DenseNet169, InceptionV3, and Xception algorithms. These trained models were combined using the soft voting method, an ensemble learning method, to create an ensemble learning model. For comparison with lightweight CNN models, the MobileNet and VGG19

models were also trained separately with both datasets. The training results of each model and the ensemble model were analyzed and compared one by one. The dataset was divided into 80% for training and 20% for testing, respectively, and an early stopping function was used to prevent overfitting. The prediction performance of each classifier was calculated with accuracy, precision, recall, and F1 score metrics. The outputs and findings of the models are presented below.

Table I summarizes the training hyperparameters used for the three compared deep learning models (Xception, InceptionV3, DenseNet169). To ensure a fair comparison, all models were trained with largely common settings such as 100 epochs, a batch size of 32, the Adam optimizer, a 0.2 dropout rate, and the Categorical Cross-Entropy loss function. To improve model performance, different learning rates specifically optimized for each network architecture were chosen.

TABLE I.
PARAMETERS OF DEEP LEARNING ALGORITHMS

Model	Xception	InceptionV3	DenseNet169
Epochs	100	100	100
Batch size	32	32	32
Optimizer	Adam	Adam	Adam
Learning rate	0,000025	0,000006	0,000003
Units	400	400	400
Dropout	0.2	0.2	0.2
Activation functions	ReLU (Dense), Softmax (Output)	ReLU (Dense), Softmax (Output)	ReLU (Dense), Softmax (Output)
Loss function	Categorical crossentropy	Categorical crossentropy	Categorical crossentropy

Table II shows that the best prediction performance is given by the ensemble learning model. This model is followed by the DenseNet169 algorithm, which achieves the most successful results among the individual classifiers. The Xception and InceptionV3 models, on the other hand, showed a lower prediction performance compared to DenseNet169 on this dataset. The MobileNet and VGG19 models, added for comparison, showed the lowest performance. It is clearly seen that the lightweight CNN models did not show sufficient performance in this study. The bold values in Table II indicate the highest performance values among the compared models.

According to Table III, the most accurate prediction performance on the second dataset is given by the Deep Solar Ensemble model. The Xception algorithm comes second, being the best individual classifier. The DenseNet169 and InceptionV3 models, however, showed poorer prediction performance relative to Xception on this dataset. The bold values in Table III represent the best performance values among the compared models.

The performance of the models during training was analyzed by means of graphs showing the change in accuracy and loss values on the validation dataset according to the epochs.

Fig. 10 shows the validation accuracy curves of three different models (DenseNet169, Xception, InceptionV3) over 100 epochs. As can be seen from the graph, the verification accuracy of all models showed a general increasing trend as the training progressed and reached an equilibrium (plateau) over time. Comparing these curves, DenseNet169 (blue line) performed the best among the individual models, reaching a validation accuracy of about 96%. InceptionV3 (red line) showed the second-best accuracy at around 94-95%, while Xception (black line) achieved a more modest result compared to the other two models, with an accuracy of around 91-92%.

Fig. 11 shows the change in the validation loss values of the same models according to the epochs. Consistent with the accuracy graph, it is seen that the loss values of all models generally decrease as the training progresses and reach a plateau after a certain point. The model with the lowest loss value was again DenseNet169, followed by InceptionV3, and the highest loss value was observed in the Xception model. These graphs show that the models learn on the validation set and their performance reaches saturation after certain epochs. These general trends in training and validation metrics reflect the trainability and generalization capacities of the models.

IV. DISCUSSION

The detection of dust and dirt accumulation in solar panels is critical to improve energy efficiency and optimize maintenance processes. In this field, deep learning-based approaches combined with image processing techniques have provided effective solutions. Important works in the literature and the proposed model are summarized in Table IV.

As seen in Table IV, studies in literature have generally used deep learning models to detect dust accumulation on solar panels. For example, Saquib et al. [6] achieved 88% accuracy with ANN, while Abukhait [8] achieved 92% accuracy with ResNet50. Alatwi et al. [10] achieved 86.79% accuracy with the combination of DenseNet169 and SVM, while Varikuti et al. [29] achieved 82.63% accuracy with EfficientNetB0 and 87.32% accuracy with DenseNet121. However, in most of these studies, the ensemble learning approach was not adopted, and no details about the preprocessing steps were given. This study, on the other hand, evaluated DenseNet169, Xception, and InceptionV3 models individually, and then developed the Deep Solar Ensemble model, achieving 97.02% accuracy. The bold values in Table IV indicate the highest accuracy obtained across all models. This innovation was achieved through the use of advanced preprocessing techniques (HSV color space conversion and morphological operations) and ensemble learning, providing one of the highest accuracies in the literature and an important step in optimizing solar panel maintenance processes.

V. CONCLUSION

In this study, the detection of dirt and dust particles that reduce energy efficiency in solar panels is targeted using artificial intelligence and deep learning techniques. The images in the dataset were subjected to a series of preprocessing steps before being prepared for model training; these steps include bringing the images to a standard size (224×224), normalization of pixel values (range [0, 1]), conversion to HSV color space for highlighting dusty areas, color thresholding, and morphological operations (opening and closing). To solve the problem, DenseNet169, InceptionV3, and Xception image classification models based on CNN architecture were trained using these preprocessed data. In order to increase the predictive power of these models, a new model called the Deep Solar Ensemble was created by combining it with the soft voting method from ensemble learning approaches. The proposed Deep Solar Ensemble model, which combines DenseNet169, Xception, and InceptionV3, achieved outstanding results with an accuracy of 97.02%, precision of 97.29%, recall of 96.56%, and an F1 score of 96.92%, significantly surpassing the performance of the individual models. This ensemble approach

TABLE II.
METRIC RESULTS OF DEEP LEARNING ALGORITHMS ON THE FIRST DATASET

Model	VGG19	MobileNet	DenseNet169	Xception	InceptionV3	Deep Solar Ensemble
Accuracy	0.880	0.882	0.96	0.916	0.946	0.97
Recall	0.864	0.857	0.955	0.899	0.945	0.965
Precision	0.888	0.903	0.962	0.933	0.943	0.972
F1 score	0.876	0.879	0.959	0.915	0.944	0.969

TABLE III.
METRIC RESULTS OF DEEP LEARNING ALGORITHMS ON THE SECOND DATASET

Model	VGG19	MobileNet	DenseNet169	Xception	InceptionV3	Deep Solar Ensemble
Accuracy	0.810	0.850	0.845	0.856	0.805	0.897
Recall	0.817	0.852	0.838	0.844	0.775	0.881
Precision	0.809	0.867	0.859	0.844	0.814	0.901
F1 score	0.813	0.859	0.843	0.838	0.780	0.886

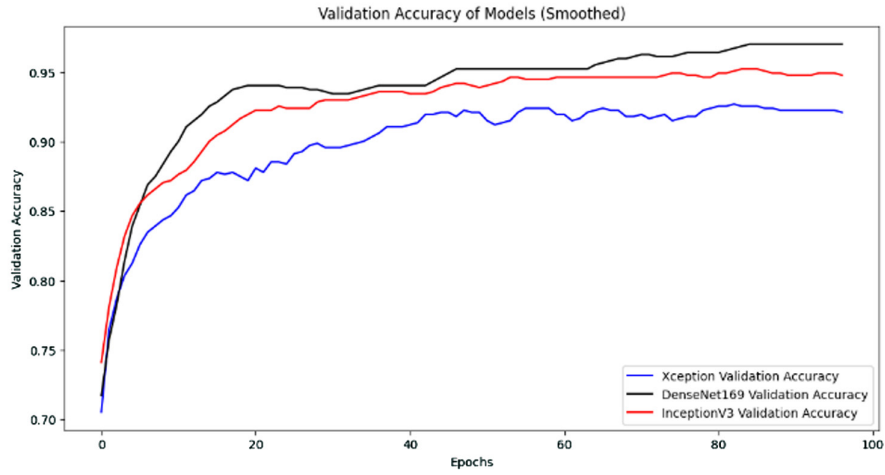


Fig. 10. Accuracy graph of the models.

addresses class imbalance and improves generalization capability, highlighting its potential for real-world deployment. Nonetheless, computational complexity and the need for real-time processing remain practical challenges to be addressed.

The findings show that the deep solar ensemble model exhibits a higher classification performance compared to the single models.

This result emphasizes the effectiveness and performance improvement of ensemble learning methods, especially in real-world datasets where class imbalances are common. Therefore, it can be predicted that this approach will be used more frequently in future classification-based studies. Ultimately, this research is expected to contribute to improving the energy efficiency of solar panels and optimizing maintenance processes such as panel cleaning.

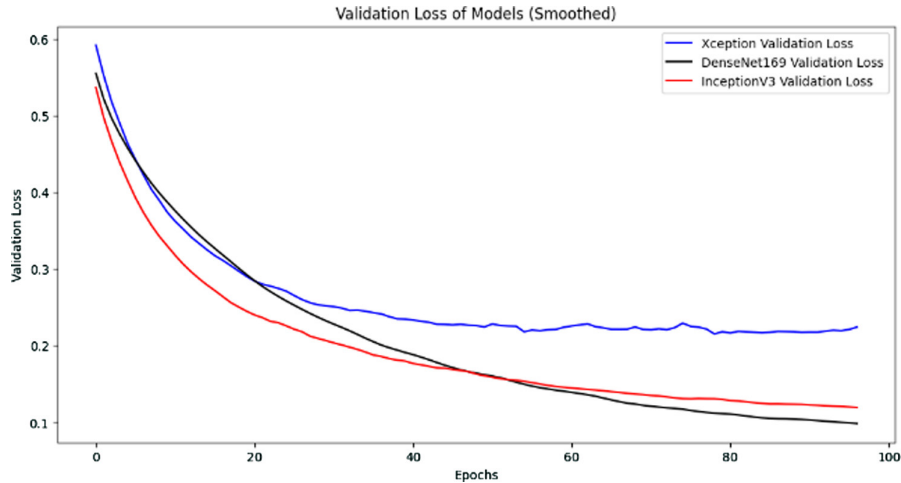


Fig. 11. Loss graph of the models.

TABLE IV.
LITERATURE COMPARISON TABLE

Study	Model	Accuracy (%)	Dataset
Saquib et al. (2021)	ANN	88	Special
Abukhait (2021)	ResNet50	92	Special
Alatwi et al. (2022)	DenseNet169 + SVM	86.79	Special
Varikuti et al. (2024)	EfficientNetB0	82.63	Kaggle
Varikuti et al. (2024)	DenseNet121	87.32	Kaggle
Deep Solar Ensemble (Proposed Model)	Deep solar ensemble	97.02	Kaggle

ANN, artificial neural network; SVM, support vector machine.

This study, while with positive results, has several limitations that should be considered in the interpretation of the results and that offer possibilities for future research. One of the principal limitations was the hardware capacity utilized for the experiments. The model training and testing were conducted on an NVIDIA GTX 1650 graphics card, which has 4 GB of VRAM. This hardware constraint prevented the authors from trying larger batch sizes, attempting more complex model architectures, and reducing training times. Future work conducted with more powerful computational resources may be able to achieve higher performance through more extensive hyperparameter tuning and the application of deeper models. Second, there were constraints related to the multi-class dataset. The model's performance on this second data was not as spectacular as it had been on the binary data for two reasons. First, the larger number of classes (six fault types) inherently makes the classification task harder. Second, the relatively small number of images in some classes, such as "Physical-Damage" and "Electrical-damage," caused a class imbalance that made it hard for the model to generalize to all classes. A larger and more balanced multi-class dataset would likely improve the model's real-world robustness and accuracy.

Data Availability Statement: The data that support the findings of this study are available upon request from the corresponding author.

Peer-review: Externally peer-reviewed.

Author Contributions: Concept – M.Y.E., H.A.; Design – M.Y.E., H.A.; Supervision – M.Y.E., H.A.; Resources – M.B., A.F.; Materials – M.B., A.F.; Data Collection and/or Processing – M.B., A.F.; Analysis and/or Interpretation – M.B., A.F.; Literature Search – M.B., A.F., M.Y.E.; Writing Manuscript – M.B., A.F., M.Y.E., H.A.; Critical Review – M.Y.E., H.A.

Declaration of Interests: The authors have no conflicts of interest to declare.

Funding: The authors declare that this study received no financial support.

REFERENCES

1. Ç. D. Gezgin, "Güneş panellerinde, güneş takip sistemlerinin ve panel kirliliğinin panel verimliliğine etkisinin incelenmesi," M.S. thesis, Dept. of Mechatronics Eng., Inst. of Sci., Trakya Univ., Edirne, Turkey, 2023

2. R. Aman, M. Rizwan, and A. Kumar, "Fault classification using deep learning based model and impact of dust accumulation on solar photovoltaic modules," *Energy Sources A*, vol. 45, no. 2, pp. 4633–4651, 2023. [\[CrossRef\]](#)
3. M. Onim et al., "SolNet: A convolutional neural network for detecting dust on solar panels," *Energies*, vol. 15, no. 22, Art. no. 8100, 2022.
4. H. Malik, M. Alsabban, and S. M. Qaisar, "Arduino based automatic solar panel dust disposition estimation and cloud based reporting," *Procedia Comput. Sci.*, vol. 194, pp. 102–113, 2021. [\[CrossRef\]](#)
5. K. A. Abuqaoud, and A. Ferrah, "A novel technique for detecting and monitoring dust and soil on solar photovoltaic panel," in Proc. 2020 Advances Sci. Eng. Technol. Int. Conf. (ASET), Dubai, UAE, Feb, pp. 1–6. [\[CrossRef\]](#)
6. D. Saquib, M. N. Nasser, and S. Ramaswamy, "Image processing based dust detection and prediction of power using ANN in PV systems," in Proc. 2020 Third Int. Conf. Smart Syst. Inventive Technol. (ICSSIT), Tirunelveli, India, Aug, pp. 1286–1292. [\[CrossRef\]](#)
7. S. Keerthana, S. Hariharasudhan, and M. Sundaram, "Image processing-based dust detection for solar panels," in Proc. 2024 Int. Conf. Smart Syst. Electr. Electron. Commun. Comput. Eng. (ICSSECC), Coimbatore, India, Jun. pp. 511–515, 2024.
8. J. Abukhait, "Dust detection on solar panels: A computer vision approach," *Ing. Syst. Inf.*, vol. 29, no. 2, pp. 533–541, 2024. [\[CrossRef\]](#)
9. V. Kavya, and R. M. R. Keshav, "Solar dust detection system," in Proc. 2018 Int. Conf. Power Energy, Environ. Intell. Control (PEEIC), Greater Noida, India, Apr. New York: IEEE, 2018, pp. 138–140. [\[CrossRef\]](#)
10. A. M. Alatwi, H. Albalawi, A. Wadood, H. Anwar, and H. M. El-Hageen, "Deep learning-based dust detection on solar panels: A low-cost sustainable solution for increased solar power generation," *Sustainability*, vol. 16, no. 19, Art. no. 8664, 2024. [\[CrossRef\]](#)
11. A. Mobiny, A. Singh, and H. V. Nguyen, "Risk-aware machine learning classifier for skin lesion diagnosis," *J. Clin. Med.*, vol. 8, no. 8, Art. no. 1241, 2019. [\[CrossRef\]](#)
12. K. Nair, A. Deshpande, R. Guntuka, and A. Patil, "Analysing X-ray images to detect lung diseases using DenseNet-169 technique," *SSRN Electron. J.*, 2022. [\[CrossRef\]](#)
13. R. T. Wahyuningrum, A. Kusumaningsih, W. Putra Rajeb, and I. K. E. Purnama, "Classification of corn leaf disease using an optimized DenseNet-169 model," in Proc. 2021 9th Int. Conf. Big Data Inf., Technol. (ICBIT), Surabaya, Indonesia, Dec, 2021, pp. 67–73.
14. A. Mehmood, "Efficient anomaly detection in crowd videos using pre-trained 2D convolutional neural networks," *IEEE Access*, vol. 9, pp. 138283–138295, 2021. [\[CrossRef\]](#)
15. S. N. Endah, and I. N. Shiddiq, "Transfer learning of Xception architecture for garbage classification," in Proc. 2020 4th Int. Conf. Informatics Comput. Sci. (ICICoS), Yogyakarta, Indonesia, Nov., 2020, pp. 1–4.
16. S. H. Kassani, P. H. Kassani, R. Khazaeinezhad, M. J. Wesolowski, K. A. Schneider, and R. Deters, "Diabetic retinopathy classification using a modified Xception architecture," in Proc. 2019 IEEE Int. Symp. Signal Process. Inf. Technol. (ISSPIT), Atlanta, GA, USA: Dec, 2019, pp. 1–6. [\[CrossRef\]](#)
17. L. Ali, F. Alnajjar, H. A. Jassmi, M. Gocho, W. Khan, and M. A. Serhani, "Performance evaluation of deep CNN-based crack detection and localization techniques for concrete structures," *Sensors (Basel)*, vol. 21, no. 5, Art. no. 1688, 2021. [\[CrossRef\]](#)
18. X. Xia, C. Xu, and B. Nan, "Inception-v3 for flower classification," in Proc. 2017 2nd Int. Conf. Image, Vision Comput. (ICIVC), Shenzhen, China, Jun, 2017, pp. 783–787.
19. S. M. Sam, K. Kamardin, N. N. A. Sjarif, and N. Mohamed, "Offline signature verification using deep learning convolutional neural network architectures GoogleNet Inception-v1 and Inception-v3," *Procedia Comput. Sci.*, vol. 161, pp. 475–483, 2019.

20. R. Polikar, "Ensemble learning," in *Ensemble Machine Learning: Methods and Applications*, C. Zhang, Ed. Cham, Switzerland: Springer, 2012, pp. 1–34. [\[CrossRef\]](#)
21. H. Lappalainen, and J. W. Miskin, "Ensemble learning," in *Advances in Independent Component Analysis*, M. Girolami, Ed. London, UK: Springer, 2000, pp. 75–92. [\[CrossRef\]](#)
22. A. Dogan, and D. Birant, "A weighted majority voting ensemble approach for classification," in Proc. 2019 4th Int. Conf. Comput. Sci. Eng. (UBMK), Elazığ, Turkey, Sep, 2019, pp. 1–6.
23. H. Wang, Y. Yang, H. Wang, and D. Chen, "Soft-voting clustering ensemble," in Multiple Classifier Systems: 11th Int. Workshop, MCS 2013, Proc., Nanjing, China, May, pp. 307–318, 2013.
24. M. A. Khan, N. Iqbal, H. Jamil, H. Jamil, and D. H. Kim, "An optimized ensemble prediction model using AutoML based on soft voting classifier for network intrusion detection," *J. Netw. Comput. Appl.*, vol. 212, Art. no. 103560, 2023. [\[CrossRef\]](#)
25. K. Tao *et al.*, "Unlocking potential of pyrochlore in energy systems via soft voting ensemble learning," *Small*, vol. 20, no. 42, Art. no. 2402756, 2024. [\[CrossRef\]](#)
26. M. Ertel, A. Sadqui, S. Amali, I. Mahmoudi, Y. Bouferma, and N. E. El Faddouli, "Predicting the severity of new SARS-CoV-2 variants in vaccinated patients using machine learning," *J. Theor. Appl. Inf. Technol.*, vol. 101, no. 10, p. 2023, 2023.
27. R. Yacoub, and D. Axman, "Probabilistic extension of precision, recall, and F1-score for a more comprehensive evaluation of classification models," in Proc. 2020 First Workshop Eval. Comparison NLP Syst. (EvalNLP Sys), Helsinki, Finland, Nov., 2020, pp. 79–91.
28. Y. Liu, Y. Zhou, S. Wen, and C. Tang, "A strategy on selecting performance metrics for classifier evaluation," *Int. J. Mob. Comput. Multimedia Commun.*, vol. 6, no. 4, pp. 20–35, 2014. [\[CrossRef\]](#)
29. V. S. S. Reddy, A. Sreejagathi, S. M. Venthana, and P. Vijayakumar, "Deep learning in solar panel maintenance: A model comparison for automated cleaning," in Proc. 2025 Fifth Int. Conf. Adv. Electr. Comput. Commun. Sustain. Technol. (ICAECT), Istanbul, Turkey, Jan. pp. 1–7, 2025, pp. 1–7. [\[CrossRef\]](#)