Bulletin of Electrical Engineering and Informatics
Vol. 15, No. 4, August 2026, pp. 3398~3407
ISSN: 2302-9285, DOI: 10.11591/eei.v15i4.11191

Comparative study of pre-trained CNN models for multiclass fault detection in solar panels

Karli Eka Setiawan[1], Marvel Martawidjaja[2], Hayyun Lisdiana[3]
[1]Department of Computer Science, School of Computer Science, Bina Nusantara University, Jakarta, Indonesia
[2]Data Science Program, Department of Computer Science, School of Computer Science, Bina Nusantara University, Jakarta, Indonesia
[3]Department of Chemistry Education, Faculty of Mathematics and Natural Science, Universitas Negeri Jakarta, Jakarta, Indonesia

Article Info

Article history:

Received Aug 12, 2025
Revised Mar 13, 2026
Accepted Jul 14, 2026

Keywords:

Clean energy
Computer vision for renewable energy
Deep learning
Pre-trained convolutional neural network
Solar panel fault detection

ABSTRACT

Solar power as renewable energy can be an alternative to fossil-based power where it is carbonless, environmentally friendly, and combats climate change. In solar panel systems, manual assessment by personnel is time-demanding and prone to human error, necessitating automated solutions. The implementation of smart systems that can automatically detect objects that hinder the solar panel from receiving solar energy can be very helpful in reducing the potential threat of decreasing performance in power generation. This study proposes a convolutional neural networks (CNN)-based image classification approach to automatically identify common solar panel conditions using visual data. The dataset used was a public dataset titled "Solar Panel Images: Clean and Faulty Images", obtained from Kaggle, containing six classes for multiclass classification. The most effective pre-trained CNN-based deep learning model for uncovering issues in solar panels was inception-V3, achieving an overall accuracy of 91% and the highest F1-score in four categories: clean, electrical damage, physical damage, and snow coverage. These outcomes confirm the potential of implementing deep learning image classification for enhancing solar panel maintenance and monitoring systems in real-world applications.

Corresponding Author:

Karli Eka Setiawan
Department of Computer Science, School of Computer Science, Bina Nusantara University
Jakarta, Indonesia
Email: karli.setiawan@binus.ac.id

1. INTRODUCTION

Renewable energy sources, including solar, wind, hydrological, tidal, wave, and geothermal energy, are deemed sustainable and ecologically benign in contrast to fossil fuels, which are limited and substantially contribute to environmental deterioration and climate change [1]. According to the data, fossil fuels account for 83% of total world energy use [2]. As people become more aware of the harmful effects of traditional carbon-emitting power production methods, the adoption of renewable energy sources is fast increasing. Solar panels have the potential to be a novel approach for combating climate change and lowering carbon emissions. Deep learning, an approach for acquiring multi-tiered representations of data and their distributions, has demonstrated exceptional performance in a variety of environmental monitoring applications, such as indoor climate prediction [3] and water quality parameter forecasting [4]. For many years, deep learning has considerably boosted machine learning (ML) research by solving high-dimensional data issues [5], making it especially ideal for complicated visual identification applications like solar panel fault detection. However, solar panel efficacy can be affected by factors such as solar direction and radiation, dust collection, and system temperature [6]. The accumulation of dust on solar panel surfaces is a significant issue that diminishes energy yield, as evidenced by a scientific paper indicating that the presence of dust, such as coal dust, fine sand, and gypsum, can result in energy yield losses of up to 64% [7]. Another cause of inefficiency of solar panels is the shading effect of bird droppings, which can reduce energy yield to as much as 38% [8]. During winters, solar panels also can reduce their performance to 90%-100% for monthly electricity generation by covering the solar panel surface [9].

In recent years, several studies have applied deep learning, especially convolutional neural networks (CNNs), to detect diverse fault types in solar photovoltaic (PV) panels, including dust accumulation, bird excrement, snow, structural/material damage, and electrical anomalies. Rudro *et al.* [10] proposed solar panel fault network (SPFNet), a hybrid U-network (UNet)/InceptionV3 model enhanced with squeeze-and-excitation and residual blocks demonstrating significantly improved segmentation-based classification performance. An explainable VGG16-based CNN model was evaluated in 2024 for detecting dust and bird-dropping contaminants, with balanced dataset augmentation and oversampling techniques [11]. A comparative analysis using several pretrained architectures—including MobileNetV3, VGG16/19, and residual network (ResNets)—on six fault categories reported up to 96.11% test accuracy with MobileNetV3, underscoring the efficacy of lightweight CNNs in real-time solar panel monitoring [12]. Additionally, a review of deep learning methods in PV defect detection highlights gaps such as dataset imbalance and generalization challenges, while identifying opportunities in physics-aware and interpretable model design [13]. Finally, the 2025 study by Kabir *et al.* [14] proposed a low-cost light-sensor–based system combined with image processing and ML to detect dust and bird/insect droppings in situ; they reported up to 55% visible light obstruction and associated energy loss quantification for proactive maintenance.

Another popular severe problem in solar panels is potential induced degradation (PID), which, as time passes, can appear, impacting the power output drop dramatically with a 27% to 39% drop in power [15]. There are two frequent varieties of solar panel damage, like physical damage and electrical damage, where physical damage refers to the existence of material cracks, fractures, and surface defect marks on the panel surface that are usually affected by weather conditions, impact from fallen objects, or manufacturing failure, and electrical damage refers to issues in electrical circuitry and connections related to the ability of solar panels to produce and deliver electrical energy with high efficiency that are usually caused by hot spots, corrosion, defective bypass diodes, or other problems [16]. Given the wide range of failure modes that reduce solar panel efficiency, automated detection and classification systems are vital to sustain peak energy performance. So, this research concentrated on identifying solar panel failures for monitoring reasons to optimize energy yield efficiency using a deep learning approach to detect the presence of some failures such as bird droppings, dust, snow, physical damage, and electrical damage with a computer vision approach. This research focuses on the implementation of pretrained CNN-based models with ResNet50, EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB7, EfficientNetV2-M, InceptionV3, InceptionResNetV2, and Xception, for image-based multiclass solar panel condition classification. As a research contribution, this study conducts a comparative evaluation of multiple pre-trained CNN architectures for multiclass solar panel fault detection on a newly available public dataset, addressing the lack of benchmark studies in this area.

2. METHOD

2.1. Dataset

This research on solar panel faults used a publicly available image dataset titled "Solar Panel Images Clean and Faulty Images" containing six class labels for multiclass classification tasks such as bird droppings, clean, dusty, electrical damage, physical damage, and snow coverage obtained from Kaggle with the link (https://www.kaggle.com/datasets/pythonafroz/solar-panel-images?resource=download). The dataset used in this study is relatively recent, having been made publicly available on Kaggle in early 2025. As a result, this work provides one of the earliest empirical evaluations of CNN-based models on this dataset, offering baseline performance results that may serve as a reference for future studies.

This research split into three parts, such as 619 images for the training set, 129 images for the validation set, and 137 images for the testing set, as detailed in Table 1. For deep learning research, this dataset is too little and not balanced, where the top three classes (bird-drop, clean, and dusty) contributed more than 66% of the overall data and the physicaldamage class, the least class, became underrepresented, containing only ~7.8% of the overall dataset. The samples of each class are depicted in Figure 1. This unbalanced data can become a problem for deep learning research, but research did not use any image generation techniques to balance the data. This is because this is our early research in computer vision for detecting faulty solar panels using a deep learning approach and investigating the capability of a pre-trained CNN-based model. We plan to apply balanced techniques in future work.

Table 1. Number of images per class in the training, validation, and testing datasets

Class	Total images of training set	Total images of validation set	Total images of testing set
Bird-drop	145 images	30 images	32 images
Clean	135 images	28 images	30 images
Dusty	133 images	28 images	29 images
Electrical damage	72 images	15 images	16 images
Physical-damage	48 images	10 images	11 images
Snow-covered	86 images	18 images	19 images
Total	619 images	129 images	137 images

Figure 1. Representative dataset samples from each of the six solar panel fault classes

2.2. Model overview

The detection of solar panel failures through image analysis presents a complex computer vision problem that requires sophisticated pattern recognition capabilities. The most powerful method for such image inspection tasks is deep learning, and in particular, CNNs because they automatically learn and develop hierarchical feature representations from the image data [17]. Deep learning demonstrates exceptional suitability for solar panel failure detection due to several key characteristics of the problem domain. Solar panel defects manifest as subtle visual anomalies including micro-cracks, hot spots, discoloration, and soiling patterns that are often difficult to detect through traditional image processing methods [18]. These defects can vary significantly in appearance, scale, and location across different panel types and environmental conditions, making rule-based detection systems inadequate. CNNs excel at this task because they can automatically learn complex feature hierarchies that capture both low-level texture patterns and high-level semantic information relevant to defect identification. The convolutional layers extract local features such as edges, corners, and texture patterns, while deeper layers combine these features to recognize more complex defect patterns [19].

Deep neural networks' hierarchical feature learning capabilities is especially useful for solar panel inspection, because errors might manifest as combinations of several visual cues rather than basic geometric patterns. Furthermore, deep learning models can account for the inherent heterogeneity in solar panel images produced by varying lighting circumstances, viewing angles, panel manufacturers, and installation environments. The capacity to build robust feature representations that generalize across these differences is critical for developing realistic failure detection algorithms that can function well in real-world deployment scenarios [20]. Due to the capabilities of deep learning especially pre-trained CNN deep architectures which is implemented in many multidisipline area, has motivated this research to implement it for energy domain problem especially for solar panel failure detection. Pre-trained models offer significant benefits compared to training from the ground up, particularly when working with constrained domain-specific datasets [21]. Transfer learning with pre-trained CNNs is a great way to identify troubles with solar panels because these models have already learned basic visual signals from big datasets like ImageNet. These learned traits, such edge detectors, texture analyzers, and shape recognizers, are great places to start when looking for problems with solar panels [22].

2.2.1. ResNet

He *et al.* [23] introduced ResNet in 2015 to address the challenge of training very deep neural networks. As deep networks add more layers, a performance contradiction known as the degradation problem arises. This paradox, as stated by ResNet, occurs in both the training and testing datasets. What creates this contradiction is difficult to pinpoint; in fact, the underlying problem derives from the difficulty to obtain an adequately optimized network. More specifically, deep networks struggle to extract meaningful gradients due to the vanishing gradient problem. Insightfully, ResNet identifies the inability to learn the identity mapping function as a critical weakness. An innovative aspect of ResNet is the residuals or skip connections that enable the transfer of information from earlier to deeper layers without any processing. Rather than learning a direct mapping H(x), ResNet learns a residual mapping F(x)=H(x)-x. This helps the network to learn identity functions whenever required. These innovations help train networks with hundreds or even thousands of layers without degrading performance [24]. ResNet-50 is a 50-layer deep residual network and is one of the most popular architectures for image classification tasks [25].

2.2.2. EfficientNet

In 2019, Tan and Le [26] presented EfficientNet, which changed CNN architectural design by proposing a principled way to scaling neural networks. The motivation behind EfficientNet was the observation that previous methods for improving CNN performance typically scaled only one dimension of the network at a time, such as depth, width, or resolution. This approach was suboptimal because these dimensions are interdependent, and scaling them in isolation often led to diminishing returns. With EfficientNet, mobile inverted bottleneck blocks (MBConv) with squeeze-and-excitation optimization was considered drastically less inaccurate for its given efficiency, which enabled balanced scaling of all three dimensions depth, width, and resolution using a simple compound coefficient, and thus, compound scaling was born. Effective scaling led to positive improvements with accuracy, and efficiency and its gains have been extensively documented in previous literature. The foundation was built with a baseline neural architecture search (NAS) which resulted in EfficientNet-B0, and its foundation unlocked the potential for positive outcomes stemming from the application of compound scaling in its larger variants [27]. EfficientNet-B0 serves as the baseline model with approximately 5.3 million parameters and uses a compound scaling coefficient of 1.0. EfficientNet-B1 scales up from B0 with a coefficient of 1.1, resulting in about 7.8 million parameters and improved accuracy. EfficientNet-B2 continues this scaling with a coefficient of 1.2, reaching approximately 9.2 million parameters. EfficientNet-B7, the largest commonly used variant, employs a compound coefficient of 2.0, resulting in about 66 million parameters and state-of-the-art performance on ImageNet [28].

2.2.3. Inception

Szegedy *et al*. [29] first proposed the inception architecture in 2014, with the goal of creating deeper and larger networks while keeping computational costs under control. The essential idea behind Inception was that different sized convolution filters may be best for recognizing characteristics at different scales within the same image. Traditional methodologies necessitated selecting a single filter size, whereas Inception proposed utilizing many filter sizes in parallel inside the same layer, allowing the network to capture multi-scale characteristics. The original Inception module used 1×1, 3×3, and 5×5 convolutions, as well as 3×3 max pooling, in simultaneously, concatenating the results to generate the input to the next layer. However, this simplistic method was computationally expensive. The novel idea of employing 1×1 convolutions as dimensionality reduction modules before expensive operations dramatically lowered computational complexity while retaining representational power [30]. Inception V3, which was released in 2015, improved the original design by dividing larger convolutions into smaller ones, replacing 5×5 convolutions with two 3×3 convolutions, and factoring 3×3 convolutions into asymmetric 1×3 and 3×1 convolutions. This factorization decreased parameter and computational costs while boosting network depth. Inception V3 has roughly 24 million parameters and adds new regularization techniques such batch normalizing and label smoothing [31]. Inception-ResNet V2 represents a hybrid architecture that combines the multi-scale feature extraction capabilities of Inception modules with the training advantages of residual connections from ResNet [32].

2.2.4. Xception

Basing his claims on the inception architecture, Francois Chollet suggested a new design, extreme inception (Xception) in 2016 [33]. He sought reasoning on if channel interactions on CNNs could be decoupled, cross-channel correlations and spatial correlations could be treated separately. The Inception module as depthwise separable convolution is the main idea behind Xception. A depthwise separable convolution spatially convolves and then applies pointwise convolution on the merged outputs. The motivation for Xception stemmed from the observation that regular convolutions perform channel mixing and spatial correlation modeling simultaneously, which may not be optimal. Depthwise separable convolutions, the core building block of Xception, first apply a spatial convolution to each input channel separately, then use a 1×1 convolution to combine the results. This approach significantly reduces the number of parameters and computational cost while often maintaining or improving performance [34]. Xception outperformed many traditional networks while having a lower parameter count with 22.9 million parameters on ImageNet. Because of the architecture's efficiency in parameters and computation, it is well suited for transfer learning, particularly in the case of automated solar panel inspection systems where computational resources are limited [35].

2.3. Experiment and parameter training

This research faced the limitation of computation problems due to the research budget, which caused hyperparameter searches using some techniques like grid search or random search to be unable to be done. This research used a guessing or trail-and-error approach to figure out how many neurons to use in each dense layer of the classifier part of the deep learning models, as shown in Figure 2. Firstly, this research used 128 neurons for the first dense layer and 64 neurons for the second dense layer in classifier parts of each CNN. Pretrained-based model, then we increased the number of each neuron and decreased the dropout if the result was underfit, and we decreased the number of neurons and increased the dropout if the result was overfit. This research performs three to five experiments of each deep learning model, and then we choose the best model achieving the best testing result.

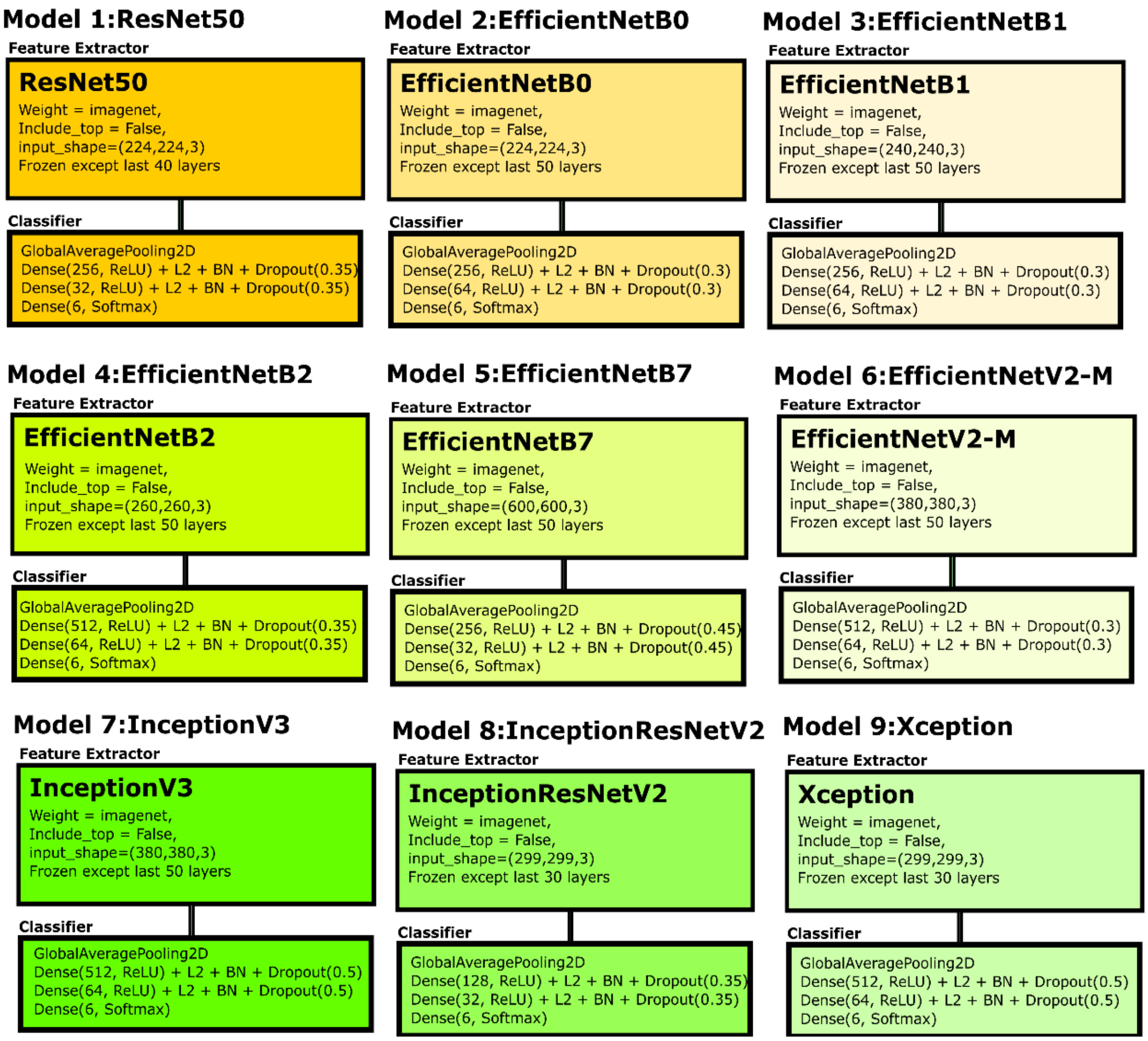

Figure 2. The architectural designs of the nine pre-trained CNN models used in this research, illustrating the feature extractor and classifier components

The data augmentation mechanism performed in this research used the ImageDataGenerator class, including the preprocessing function, filling mode, rotating, shifting, shearing, zooming, and flipping. This research set filling mode, rotating, width shifting, height shifting, shearing, zooming, and horizontal flipping with values "nearest," 20, 0.2, 0.2, 0.2, 0.2, and "True," respectively. Meanwhile, for the preprocessing function, every model used a different preprocessing function setup as described in Table 2 and a different input size as depicted in Figure 2. All pre-trained CNN-based deep learning architecture used in this research used ImageNet weights as previous knowledge and used a different number of unfrozen layers for every architecture model based on trial-and-error experiments. The preprocessing function for each CNN, as in Table 2, was selected according to the normalization scheme used during its original ImageNet pretraining, ensuring consistency in input distribution and preserving pretrained feature representations.

Table 2. Model-specific input preprocessing functions applied for each pretrained CNN architecture

Pretrained CNN-based model	Preprocessing function
ResNet50	from tensorflow.keras.applications.resnet50 import preprocess_input
EfficientNetB0	from tensorflow.keras.applications.efficientnet import preprocess_input
EfficientNetB1	from tensorflow.keras.applications.efficientnet import preprocess_input
EfficientNetB2	from tensorflow.keras.applications.efficientnet import preprocess_input
EfficientNetB7	from tensorflow.keras.applications.efficientnet import preprocess_input
EfficientNetV2-M	from tensorflow.keras.applications.efficientnet import preprocess_input
InceptionV3	Rescale=1./255
InceptionResNetV2	from tensorflow.keras.applications.inception_resnet_v2 import preprocess_input
Xception	from tensorflow.keras.applications.xception import preprocess_input

3. RESULTS AND DISCUSSION

This research sets 50 epochs for each pre-trained deep learning model, where the learning process is illustrated in Figure 3 for accuracy history. Looking at the accuracy history in Figure 3, ResNet50, EfficientNetB2, and Xception maintained a steady accuracy of around 90% or higher and did not show signs of overfitting. EfficientNetV2-M has the lowest accuracy and is unstable, which shows inadequate performance in this dataset. The InceptionResNetV2 tends to show the overfitting indication with training accuracy near and almost 100% but having a wider gap with validation accuracy. Overall, these Figure 3 highlight the importance of evaluating both accuracy jointly to assess model robustness, generalization, and suitability for real-world solar panel fault detection applications.

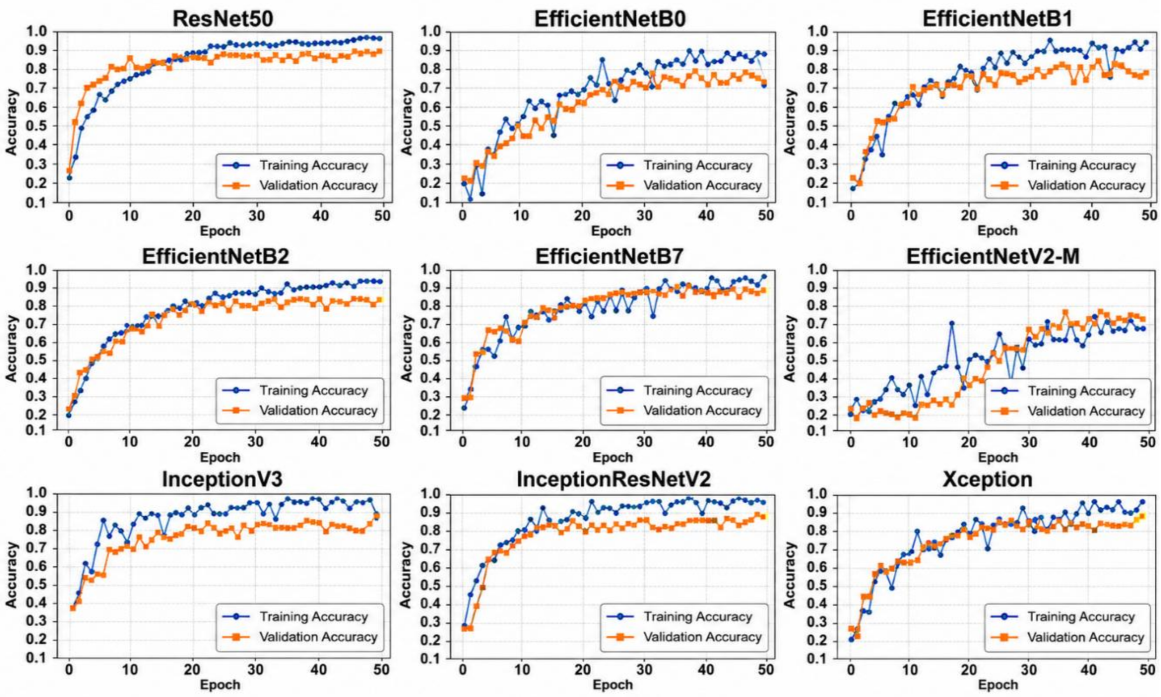

Figure 3. Training and validation accuracy curves across epochs for all pretrained deep learning models

The evaluation testing results, which used 20% of the dataset to classify six categories, such as bird excrement, clean, covered in dust, electrical malfunction, material/structural damage, and snow-covered, are displayed in Tables 3 and 4. The Table 1 dataset distribution noted the number of images used in evaluation testing for Tables 3 and 4. Overall, the best models were InceptionV3 and InceptionResNetV2, both reaching 91% accuracy, while the lowest accuracy was 78% from EfficientNetV2-M. Overall, InceptionV3 and InceptionResNetV2 achieved 91% accuracy, while EfficientNetV2-M achieved the lowest accuracy at 78%. The physical-damage class is becoming the most challenging class to predict because almost all models achieved low recall scores under 75%, such as ResNet50 with 73%, EfficientNetB0 with 55%, EfficientNetB1 with 64%, EfficientNetB2 with 64%, EfficientNetV2-M with 27%, and Xception with 73%. This can be a sign of the visual or physical damage class, maybe similar to other classes. It seems that all models were very good at predicting snow-coverage class, where all models were above 88% in F1-score. If we look at the F1-score for the harmonic mean of precision and recall, the InceptionV3 can achieve the best F1-score between all models in four different classes, such as clean, electrical malfunction, material/structural damage, and snow obstruction, so that InceptionV3 deserves to be crowned the best model overall.

For future work, this research may apply for any grant with the aim of financing the computing process to perform hyperparameter tuning by conducting grid search or random search to search optimal settings for all deep learning models. The multi-label classification approach will be explored further because, in a real case, the problem in the solar panel may occur together with other faults. Image augmentation may be explored further in future work for better evaluation testing results.

Table 3. Testing performance metrics (accuracy, precision, recall, and F1-score) for the bird droppings, clean, and dusty classes across all pretrained CNN models

Index	Model	Overall	Bird-drop			Clean			Dusty		
		Overall accuracy	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
1	ResNet50	0.88	0.85	**0.97**	0.88	0.84	0.87	0.85	0.89	0.83	0.86
2	Efficient-NetB0	0.85	0.74	0.88	0.80	0.82	0.90	0.86	0.88	0.76	0.81
3	Efficient-NetB1	0.83	0.82	0.84	0.83	0.74	0.97	0.84	0.86	0.66	0.75
4	Efficient-NetB2	0.81	0.84	0.81	0.83	0.72	0.97	0.83	0.84	0.72	0.78
5	Efficient-NetB7	0.87	0.88	0.88	0.88	**0.89**	0.80	0.84	0.85	0.79	0.82
6	Efficient-NetV2-M	0.78	0.80	0.88	0.84	0.76	0.53	0.63	0.57	**0.83**	0.68
7	Inception-V3	**0.91**	**0.88**	0.88	0.88	0.88	**1.00**	**0.94**	0.92	0.79	0.85
8	Inception-ResNetV2	**0.91**	**0.88**	0.94	**0.91**	0.88	0.97	0.92	**0.96**	0.79	0.87
9	Xception	0.88	0.85	0.91	0.88	0.85	0.93	0.89	**0.96**	0.86	**0.91**

Table 4. Testing performance metrics (precision, recall, and F1-score) for the electrical damage, physical damage, and snow coverage classes across all pretrained CNN models

Index	Model	Electrical-damage			Physical-damage			Snow-coverage		
		Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
1	ResNet-50	0.89	0.83	0.86	0.80	0.73	0.76	**1.00**	0.95	0.97
2	Efficient-NetB0	0.89	**1.00**	0.94	**1.00**	0.55	0.71	**1.00**	0.89	0.94
3	Efficient-NetB1	0.82	0.88	0.85	**1.00**	0.64	0.78	0.95	0.95	0.95
4	Efficient-NetB2	0.72	0.81	0.76	0.88	0.64	0.74	**1.00**	0.79	0.88
5	Efficient-NetB7	0.80	**1.00**	0.89	0.82	0.82	0.82	0.95	**1.00**	0.97
6	Efficient-NetV2-M	0.86	0.75	0.80	0.50	0.27	0.35	0.95	0.95	0.95
7	Inception-V3	**1.00**	0.94	**0.97**	0.83	**0.91**	**0.87**	**1.00**	**1.00**	**1.00**
8	Inception-ResNetV2	0.88	0.88	0.88	0.90	0.82	0.86	0.95	**1.00**	0.97
9	Xception	0.83	0.94	0.88	0.80	0.73	0.76	**1.00**	0.84	0.91

4. CONCLUSION

This work offers a comparative analysis of various pre-trained CNN architectures for the identification of faults in image-based solar panels. Inception-V3 was the best of the models tested, with an accuracy of 91% and excellent generalization across fault categories. The results demonstrate that transfer learning utilizing CNN-based architectures is efficacious for facilitating automated monitoring and maintenance of solar systems, thereby diminishing reliance on manual inspection and enhancing the reliability of renewable energy generation. This study is one of the first to use this publicly available dataset. It sets baseline performance standards for future research on how to find problems with solar panels. Future research may concentrate on more extensive and balanced datasets, sophisticated data augmentation techniques, and the incorporation of physics-informed or hybrid models to further improve detection precision.

FUNDING INFORMATION

The article processing charge (APC) for this publication was funded by Bina Nusantara University.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Karli Eka Setiawan	✓	✓	✓	✓		✓		✓	✓	✓		✓	✓	✓
Marvel Martawidjaja		✓	✓						✓	✓				
Hayyun Lisdiana	✓		✓	✓	✓		✓			✓	✓			

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

This study on solar panel defects utilized a publicly accessible picture dataset named “Solar Panel Images Clean and Faulty Images,” available at the link https://www.kaggle.com/datasets/pythonafroz/solar-panel-images?resource=download.

REFERENCES

[1] F. Martinez-Gil, C. Sansom, A. Fernández-García, A. Alcayde-García, and F. Manzano-Agugliaro, “Maintenance techniques to increase solar energy production: A review,” *Energy Nexus*, vol. 17, pp. 1-16, Mar. 2025, doi: 10.1016/j.nexus.2025.100384.

[2] J. L. Holechek, H. M. E. Geli, M. N. Sawalhah, and R. Valdez, “A Global Assessment: Can Renewable Energy Replace Fossil Fuels by 2050?,” *Sustainability*, vol. 14, no. 8, pp. 1-22, Apr. 2022, doi: 10.3390/su14084792.

[3] Y. Zhao *et al*., “Prediction of daily home indoor temperature and relative humidity using a deep ensemble machine learning approach,” *Building and Environment*, vol. 295, May 2026, doi: 10.1016/j.buildenv.2026.114392.

[4] K. P. Wai, C. H. Koo, Y. F. Huang, and W. C. Chong, “Enhancing water quality index forecasting: a novel approach integrating signal decomposition and deep learning models for multi-step ahead predictions,” *Ain Shams Engineering Journal*, vol. 17, no. 6, Jun. 2026, doi: 10.1016/j.asej.2026.104133.

[5] K. E. Setiawan, G. N. Elwirehardja, and B. Pardamean, “Comparison of Deep Learning Sequence-To-Sequence Models in Predicting Indoor Temperature and Humidity in Solar Dryer Dome,” *Communications in Mathematical Biology and Neuroscience*, 2022, doi: 10.28919/cmbn/7655.

[6] K. R. C. Lakshmi and G. Ramadas, “Dust Deposition’s Effect on Solar Photovoltaic Module Performance: An Experimental Study in India’s Tropical Region,” *Journal of Renewable Materials*, vol. 10, no. 8, pp. 2133–2153, 2022, doi: 10.32604/jrm.2022.019649.

[7] S. Yakubu *et al.*, “A holistic review of the effects of dust buildup on solar photovoltaic panel efficiency,” *Solar Compass*, vol. 13, pp. 1-13, Mar. 2025, doi: 10.1016/j.solcom.2024.100101.

[8] K. A. M. Ali, Y. K. O. T. Osman, G. G. A. El-wahhab, T. A. M. Abdelwahab, and A. E. M. Fodah, “Improving solar PV performance under bird-dropping conditions with a dual-cooling approach,” *Scientific Reports*, vol. 15, no. 1, pp. 1-12, Mar. 2025, doi: 10.1038/s41598-024-84932-w.

[9] R. E. Pawluk, M. Rezvanpour, Y. Chen, and Y. She, “A sensitivity analysis on effective parameters for sliding/melting prediction of snow cover on solar photovoltaic panels,” *Cold Regions Science and Technology*, vol. 185, May 2021, doi: 10.1016/j.coldregions.2021.103262.

[10] R. A. M. Rudro *et al.*, “SPF-Net: Solar panel fault detection using U-Net based deep learning image classification,” *Energy Reports*, vol. 12, pp. 1580–1594, Dec. 2024, doi: 10.1016/j.egyr.2024.07.044.

[11] Y. Ledmaoui, A. El Maghraoui, M. El Aroussi, and R. Saadane, “Enhanced Fault Detection in Photovoltaic Panels Using CNN-Based Classification with PyQt5 Implementation,” *Sensors*, vol. 24, no. 22, pp. 1-20, Nov. 2024, doi: 10.3390/s24227407.

[12] S. R. Mohanty, M. U. Maruf, V. Singh, and Z. Ahmad, “Machine learning approaches for automatic defect detection in photovoltaic systems,” *Solar Energy*, vol. 298, Sep. 2025, doi: 10.1016/j.solener.2025.113672.

[13] M. Abdelsattar, A. AbdelMoety, and A. Emad-Eldeen, "Advanced machine learning techniques for predicting power generation and fault detection in solar photovoltaic systems," *Neural Computing and Applications*, vol. 37, no. 15, pp. 8825–8844, May 2025, doi: 10.1007/s00521-025-11035-6.

[14] M. S. Kabir *et al.*, "A Novel Method for Detecting Dust Accumulation in Photovoltaic Systems: Evaluating Visible Sunlight Obstruction in Different Dust Levels and AI-based Bird Droppings Detection," *arXiv preprint*, 2025, doi: 10.48550/arXiv.2501.08304.

[15] G. Badran and M. Dhimish, "Field study on the severity of photovoltaic potential induced degradation," *Scientific Reports*, vol. 12, no. 1, pp. 1-11, Dec. 2022, doi: 10.1038/s41598-022-26310-y.

[16] H. N. Noura, K. Chahine, J. Bassil, J. A. Chaaya, and O. Salman, "Efficient combination of deep learning models for solar panel damage and soiling detection," *Measurement*, vol. 251, Jun. 2025, doi: 10.1016/j.measurement.2025.117185.

[17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[18] A. Mellit, G. M. Tina, and S. A. Kalogirou, "Fault detection and diagnosis methods for photovoltaic systems: A review," *Renewable and Sustainable Energy Reviews*, vol. 91, pp. 1–17, Aug. 2018, doi: 10.1016/j.rser.2018.03.062.

[19] L. Alzubaidi *et al.*, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, pp. 1-74, Mar. 2021, doi: 10.1186/s40537-021-00444-8.

[20] M. Dhimish, V. Holmes, B. Mehrdadi, and M. Dales, "Comparing Mamdani Sugeno fuzzy logic and RBF ANN network for PV fault detection," *Renewable Energy*, vol. 117, pp. 257–274, Mar. 2018, doi: 10.1016/j.renene.2017.10.066.

[21] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A Survey on Deep Transfer Learning," in *27th International Conference on Artificial Neural Networks (ICANN*), Rhodes, Greece, Spinger, vol. 11141, 2018, pp. 270–279, doi: 10.1007/978-3-030-01424-7_27.

[22] J. A. F. Thompson, M. Schönwiesner, Y. Bengio, and D. Willett, "How Transferable Are Features in Convolutional Neural Network Acoustic Models across Languages?," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, May 2019, pp. 2827–2831, doi: 10.1109/ICASSP.2019.8683043.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[24] M. Shafiq and Z. Gu, "Deep Residual Learning for Image Recognition: A Survey," *Applied Sciences*, vol. 12, no. 18, pp. 1-43, Sep. 2022, doi: 10.3390/app12188972.

[25] A. Du, Q. Zhou, and Y. Dai, "Methodology for Evaluating the Generalization of ResNet," *Applied Sciences*, vol. 14, no. 9, pp. 1-18, May 2024, doi: 10.3390/app14093951.

[26] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in *Proceedings of Machine Learning Research (ICML)*, Long Beach, California: PMLR, 2019, pp. 6105–6114.

[27] M. Tan *et al.*, "MnasNet: Platform-Aware Neural Architecture Search for Mobile," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2019, pp. 2815–2823, doi: 10.1109/CVPR.2019.00293.

[28] M. Tan and Q. V. Le, "EfficientNetV2: Smaller Models and Faster Training," *arXiv preprint*, 2021, doi: 10.48550/arXiv.2104.00298.

[29] C. Szegedy *et al.*, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 2015, pp. 1-9, doi: 10.1109/CVPR.2015.7298594.

[30] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A review of convolutional neural networks in computer vision," *Artificial Intelligence Review*, vol. 57, no. 4, pp. 1-43, Mar. 2024, doi: 10.1007/s10462-024-10721-6.

[31] M. A. S. Al Husaini, M. H. Habaebi, T. S. Gunawan, M. R. Islam, E. A. A. Elsheikh, and F. M. Suliman, "Thermal-based early breast cancer detection using inception V3, inception V4 and modified inception MV4," *Neural Computing and Applications*, vol. 34, no. 1, pp. 333–348, Jan. 2022, doi: 10.1007/s00521-021-06372-1.

[32] A. C. S, "Advancements in CNN Architectures for Computer Vision: A Comprehensive Review," in *2023 Annual International Conference on Emerging Research Areas: International Conference on Intelligent Systems (AICERA/ICIS)*, Kanjirapally, India: IEEE, Nov. 2023, pp. 1–7, doi: 10.1109/AICERA/ICIS59538.2023.10420413.

[33] F. Salim, F. Saeed, S. Basurra, S. N. Qasem, and T. Al-Hadhrami, "DenseNet-201 and Xception Pre-Trained Deep Learning Models for Fruit Recognition," *Electronics*, vol. 12, no. 14, pp. 1-23, Jul. 2023, doi: 10.3390/electronics12143132.

[34] M. Rybczak and K. Kozakiewicz, "Deep Machine Learning of MobileNet, Efficient, and Inception Models," *Algorithms*, vol. 17, no. 3, pp. 1-17, Feb. 2024, doi: 10.3390/a17030096.

[35] R. Indraswari, R. Rokhana, and W. Herulambang, "Melanoma image classification based on MobileNetV2 network," *Procedia Computer Science*, vol. 197, pp. 198–207, 2022, doi: 10.1016/j.procs.2021.12.132.

BIOGRAPHIES OF AUTHORS

Karli Eka Setiawan received an S.Si. degree in mathematics from Padjadjaran University in 2015 and an M.Kom. degree in computer science from Bina Nusantara University in 2023. He achieved outstanding grades of excellence, summa cum laude, and best graduate in his master's study. His research interests cover artificial intelligence, computer vision, data science, blockchain, and mathematical modeling. He is an associate lecturer specialist S2 in the School of Computer Science at Bina Nusantara University. He can be contacted at email: karli.setiawan@binus.ac.id.

Marvel Martawidjaja is pursuing a bachelor's degree in data science at Bina Nusantara University while working as an AI researcher and Data Scientist intern at PT Bank Central Asia Tbk. He has co-authored multiple research papers including studies on climate change sentiment analysis, water quality prediction using neural networks, and mental health prediction with explainable AI techniques. At BCA, he has developed NLP models and machine learning pipelines for banking applications. His research focuses on explainable AI, natural language processing, and machine learning applications across various domains. He can be contacted at email: marvel.martawidjaja@binus.ac.id.

Hayyun Lisdiana is an assistant expert in the Chemistry Education Program at the Department of Chemistry Education, Universitas Negeri Jakarta, Jakarta 13220, Indonesia. She is an active member of the Indonesian Chemistry Society in the education division. She completed her bachelor's and master's degrees at Chemistry Education Program Universitas Negeri Jakarta. Her research interests include education, STEM education, learning environments, chemistry for children with special needs, and chemistry learning within STEM frameworks. She can be contacted at email: hayyunlisdiana@unj.ac.id.