MACHINE SCIENCE

ISSN: 2227-6912
E-ISSN: 2790-0479

2023, № 2 Pages 78 – 85

SOLAR PANELS CONTAMINATION DETECTION USING CNN

Javad NAJAFLI[1,a*], Yadigar IMAMVERDIYEV[1,b]

[1]Azerbaijan Technical University, Baku, Azerbaijan

E-mail: [a*]*cavad.nacafli.v@aztu.edu.az*, [b]*yadigarimam@gmail.com*

http://dx.doi.org/10.61413/GWSF2147

Abstract: Solar panels, as an important source of renewable energy, are exposed to external factors such as dust accumulation and environmental pollution, which reduces their efficiency. To ensure timely cleaning of panels from contamination, monitoring of their condition is required. Monitoring can be carried out directly by a person or using visual detectors. This paper examines the second case, namely the use of AI to assess panel contamination. The AI used to assess the condition of the panels may be trained on images captured by cameras with a different resolution than the surveillance system in which it is embedded, which may result in the AI's inability to produce results under new conditions. This work evaluates the impact of changing the quality of images processed by AI on its effectiveness. To do this, a CNN (Convolutional Neural Network) model was used and a set of images of clean and dirty solar panels was taken, followed by their separation by resolution. The model was trained on a set of images of one quality and tested on others, and this approach was applied for each resolution. As results were obtained, elements from other sets were added to each set to stabilize the training of the neural network. During the work, the importance of using images of different qualities for training a neural network was assessed, as well as the minimum set to prevent overtraining due to the lack of quality differences. Taking into account the results obtained, it is possible to prevent the creation of inflexible neural networks, as well as save on using the minimum required data to stabilize training.

Keywords: *CNN; solar panels; green energy; artificial intelligence; contamination detection*

Introduction.

Nowadays, when environmental and development issues are becoming increasingly relevant, the gradual use of green energy will inevitably attract the attention of researchers and the public [1]. In recent years of green energy, solar panels have played an important role in generating electricity from renewable sources. Research shows that the use of solar panels helps reduce greenhouse gases and create sustainable energy sources [2]. However, despite their appearance, solar panels are exposed to external factors such as pollution and dust accumulation on their surface [3]. Contamination on the surface of solar panels causes light to pass through, which ultimately reduces the amount of energy they can produce [4]. Some studies show that external influences can reduce the performance of photovoltaic panels by an astonishing 85% [5].

The purpose of this research is to develop a method to improve the efficiency of identifying dirty solar panels using artificial intelligence.

Inclusion aside, the rest of the article is organized as follows. Section 2 provides an overview of actual research in this area. Section 3 describes the proposed method based on convolutional neural networks. The experimental results are presented in Section 4. Section 5 “Conclusions” presents the main conclusions and prospects for future research.

Related works.

In the field of research devoted to optimizing energy production using solar panels, several key areas are emerging related to the problem of dust accumulation on the surface of photovoltaic modules. The paper, which analyzes image processing techniques for detecting dust on solar panels, highlights that the orientation and angle of the panels relative to the horizontal plane have an impact on performance. In this context, special attention is paid to the effects of environmental conditions, such as dust accumulation, on energy production.

The use of image processing methods has been proposed to detect dust on solar panels in order to optimize their cleaning processes [6]. The review reveals that despite the low cost and high efficiency of the proposed methods, the number of relevant studies is limited.

Another work based on computer vision examined the impact of aggressive climate conditions on the efficiency of renewable energy sources. In particular, it has been argued that the accumulation of soil and dust on the surface of photovoltaic panels reduces their energy performance. The proposed method, based on computer vision, has demonstrated high accuracy in detecting contamination and can be an effective tool for automated cleaning [7].

Another work covered the topic of color sensing to detect dirt accumulation on solar panels [8]. The proposed method uses TCS3200 and Arduino Uno to detect dirt quickly and accurately. The need for additional research was noted, including increasing the sensitivity of the color sensor and using multiple sensors to improve contamination detection efficiency. A method using a new CNN design has also been proposed [9]. According to the author's experience, it has shown a high accuracy of 98%. Another work used artificial intelligence again, but with a much lower probability of recognizing panel contamination [10].

These studies collectively highlight the importance of using advanced imaging, computer vision, and color sensing techniques to address problems associated with dust and dirt accumulation on solar panels. They also highlight the need for greater adoption of these techniques in the solar energy industry, which could potentially improve energy generation efficiency and reduce maintenance costs.

Suggested method.

Advantages of CNN

This study proposes to implement artificial intelligence based on convolutional neural networks (CNNs) to solve the problem of recognizing dirty solar panels. CNN is a type of neural network specifically designed for processing and analyzing structured mesh data such as images [11]. CNN is capable of automatically extracting important features from images. In addition, they provide automatic detection and analysis of complex patterns and textures. The CNN architecture (Figure 1) allows the model to hierarchically learn features at different levels of abstraction. For example, the first layers may study simple textures, while the deeper layers may study more complex structures. CNN has the property of being invariant to small changes in the image, such as shifts and scaling.

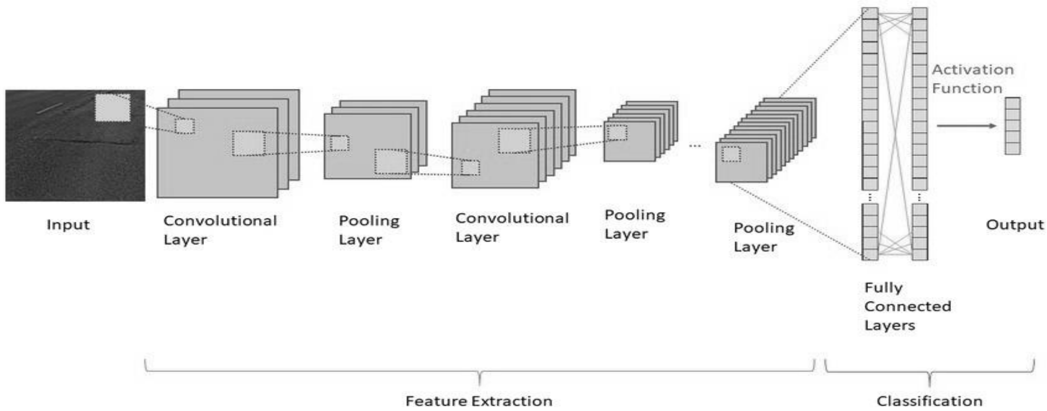


Figure 1. CNN architecture

Input layer is the entry point of our CNN, this layer represents the raw pixel values of the input image. It initiates the flow of information into the network, serving as the canvas upon which features will be extracted. Feature Extraction is pivotal phase is dedicated to capturing and highlighting relevant patterns from the input data. Convolutional layers employ learnable filters to perform convolutional operations on the input image. Filters act as feature detectors, recognizing spatial hierarchies and intricate details. Positioned after convolution, pooling layers downsample and condense the extracted features. Techniques like max or average pooling retain essential information while reducing computational load. Following feature extraction, the network transitions to the classification phase. Fully connected layer transforms high-level features into a vector, connecting every neuron and preparing for final classification. It encapsulates the consolidated knowledge acquired during feature extraction. An activation function, such as ReLU, introduces non-linearity, allowing the network to discern complex patterns and relationships. Output layer layer generates final predictions. Utilizing an appropriate activation function (e.g., softmax for classification), this layer aligns with the number of classes in the task. In this figure, the CNN seamlessly progresses from input to feature extraction and ultimately to classification, leveraging convolutional and pooling layers for nuanced feature recognition and a fully connected layer for comprehensive understanding, culminating in precise predictions at the output layer.

Mathematical background of CNN

The mathematical rationale behind CNN is based on the concepts of linear algebra and convolution.

Convolutional operation is the process of applying a filter (kernel) to the input data to extract certain features. If I is the input data, K is the filter, then the convolution operation S can be represented mathematically as follows:

$$S(i, j) = (I \cdot K)(i, j) = \sum_m \sum_n I(m,n)K(i-m, j-n) \tag{1}$$

This operation allows you to extract local features of an image, taking into account their spatial location.

The pooling operation is used to reduce data dimensionality and improve computational efficiency. Typically, the max pooling operation is used. If X is the input data, then the maximum pooling operation P can be represented as:

$$P(i, j) = \max_{m,n} X(2i+m, 2j+n) \tag{2}$$

This allows you to reduce the dimensionality of the data while preserving key features.

After feature extraction using convolutional and subsampling layers, the resulting data is fed into fully connected layers for classification or regression. If X is the output of the previous layer, W and b are the weights and biases, respectively, then the operation of a fully connected layer F can be represented as:

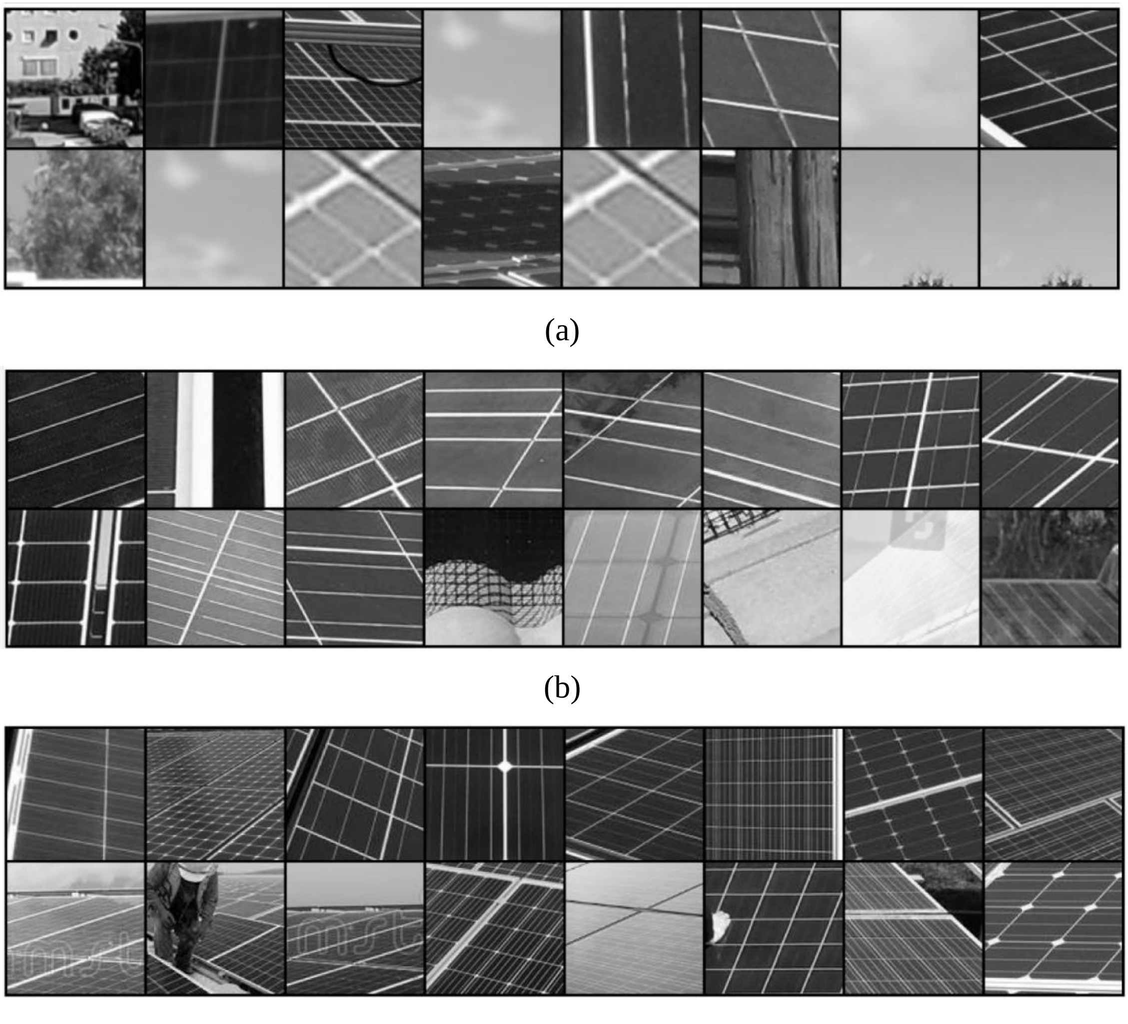
$$F(X) = \sigma(WX + b) \tag{3}$$

Where σ is the activation function (for example, ReLU or sigmoid). These operations allow CNN to learn hierarchical features from data and operate efficiently on images while preserving spatial information.

Experiments. In scientific study focusing on solar panel performance, we meticulously curated a dataset of images featuring both dusty and clean panels, categorizing them into high, medium, and low resolutions. This deliberate stratification allowed us to conduct a comprehensive analysis of the impact of image quality on our findings. For images of dusty solar panels, we prioritized capturing the granularity of dust distribution with varying resolutions. High, medium and low-resolution images were included, broadening our dataset to encompass different levels of visual detail. Similarly, for clean panels, we ensured a diverse range of resolutions to capture the pristine conditions with varying levels of clarity. This multi-resolution approach not only enriched dataset but also allowed us to explore the correlation between image quality and the accuracy of our analyses.

For the experiment, we used a standard CNN model. Two convolutional layers that take 3x32x32 input images and apply 32 filters with a 3x3 kernel, a max pooling layer (MaxPool2d), a layer (fc3) with 512 neurons and ReLU activation, a regularization layer (Dropout) after the fully connected layer fc3, a fully connected layer (fc4) with access to 3 classes. Used optimizer adam.

To begin with, images of different quality were separated into different datasets. Images have been spaced out and cut off in the middle to unify all possible resolutions (Figure 2).



(a)

(b)

(c)

Figure 2. A set of cropped images of high (a), medium (b) and low quality (c)

In the first set (Figure 2, a), you can see that padding and then cropping the center did not help in most cases, since the solar panels were not in the center of the image. In the future it is planned to correct this defect.

For initial training, we divided the sets strictly by resolution (up to 1 million pixels - low, from 1 to 4 million pixels - medium, from 4 million - high quality images). The numbers of images (clean and dirty) of solar panels for small, medium and high resolutions were: 98 and 23, 509 and 367, 886 and 679, respectively. Having trained them on different images of the same quality, we tested its capabilities on others. Below are tables that demonstrate changes in the performance indicators of the neural network depending on the part of the image sets of two other qualities added to the training set. Precision, Recall and F1-score were used for evaluation. The table also contains the number of incorrectly classified (False Negative (FN), False Positive (FP)) images for clarity. To understand the table, it is worth considering three at a time.

Table 1. Low on low

Metrics	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Precision	100%	99%	100%	99%	100%	99%
Recall	100%	100%	100%	99%	100%	100%
F1-score	100%	99%	100%	99%	100%	99%
False Negative	0	0	0	1	0	0
False Positive	0	3	0	5	0	2

Table 2. Low on medium

Metrics	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Precision	0%	100%	100%	100%	100%	100%
Recall	0%	19%	28%	25%	10%	25%
F1-score	0%	31%	44%	40%	19%	40%
False Negative	876	639	503	460	469	328
False Positive	0	0	0	0	0	0

Table 3. Low on high

Metrics	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Precision	0%	84%	88%	77%	92%	66%
Recall	0%	21%	24%	21%	16%	22%
F1-score	0%	33%	38%	33%	28%	32%
False Negative	121	83	71	63	60	43
False Positive	0	4	3	5	1	6

In the first three tables (Table 1, Table2, Table 3), the neural network trained on 0.9 of the entire set of low-quality images gradually absorbed 0.1, 0.2, 0.3... from the sets of two other qualities. The first table (Table 1) is needed to understand how adding new data to the training set affects its initial effectiveness. In it we are only interested in the loss of indicators. Tables Table 2 and Table 3 show that the best performance was achieved when the set was expanded by 0.2. F1-score rose from 31% to 44% and from 33% to 38%, respectively. At 0.2 there were also better results for Recall 28% and 24%, despite the fact that the number of FNs decreased with increasing training data. Precision began to falter when expanding to high-quality images When expanded to medium quality sets, it remained stable at 100%. The indicators in Table 1 did not change; the expansion of training data did not worsen the initial indicators.

Table 4. Medium on Low

Metrics	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Precision	0%	12%	86%	32%	50%	96%
Recall	0%	100%	100%	98%	89%	99%
F1-score	0%	21%	92%	49%	64%	98%
False Negative	1565	1238	0	5	52	1
False Positive	0	0	170	736	437	24

The following tables are shown (Table 4, Table 5, Table 6) with the performance of a neural network trained on images of average quality, followed by expanding the set to two others. Here Table 5 shows the change in the initial efficiency.

Here, the best performance was shown at an expansion of 0.5 (98% at low and 89% at high), although the F1-score changed abruptly throughout the data increase. It is also worth noting the good performance of the 0.2 extension: F1-score 92% and 81% on low and high, respectively. Precision progressed when expanded to both data sets, up to 96% on the small data sets and 86% on the high data sets. In Table 5, the maximum deterioration in F1-score did not exceed 2%.

Table 5. Medium on medium

Metrics	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Precision	100%	100%	100%	100%	100%	100%
Recall	100%	99%	99%	98%	97%	99%
F1-score	100%	99%	99%	99%	98%	99%
False Negative	0	3	1	7	11	2
False Positive	0	0	0	0	0	0

Table 6. Medium on high

Metrics	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Precision	0%	8%	68%	19%	57%	86%
Recall	0%	90%	98%	80%	95%	92%
F1-score	0%	15%	81%	31%	72%	89%
False Negative	121	1	1	4	2	4
False Positive	0	99	30	65	30	8

Lastly, a neural network was tested, trained on high-quality images and expanded to two others. Below are tables (Table 7, Table 8, Table 9) with the results, where Table 9 is an indicator of changes in initial efficiency.

Table 7. High on low

Metrics	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Precision	0%	54%	97%	47%	89%	99%
Recall	0%	45%	96%	89%	98%	99%
F1-score	0%	49%	96%	61%	93%	99%
False Negative	876	559	42	56	16	4
False Positive	0	386	31	552	99	1

Table 8. High on medium

Metrics	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Precision	0%	100%	100%	100%	100%	100%
Recall	0%	36%	97%	52%	14%	77%
F1-score	0%	53%	98%	68%	25%	87%
False Negative	1565	499	22	292	451	100
False Positive	0	0	0	0	0	0

Table 9. High on high

Metrics	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Precision	100%	95%	100%	76%	100%	100%
Recall	100%	97%	100%	98%	83%	98%
F1-score	100%	96%	100%	85%	91%	99%
False Negative	0	3	0	1	12	1
False Positive	0	5	0	20	0	0

In this case, the best indicators for the two data sets came from different extensions. When expanding the set towards low-quality images, the variable indicators stabilized and improved by expanding by 0.5 (F1-score 99%). In the case of expansion into average quality, the indicators were also variable and reached their peak at 0.2 (F1-score 98%). Precision progressed to 99% when expanding to medium data, while on small data it remained stable at 100%. Table 9 as an indicator of performance degradation varied more than 85%~99%, in contrast to the first two cases (Table1 and Table 5), but this is due to the total number of images accepted as “high quality”. Their number is several times less than the others.

Discussion.

This study examined the impact of image quality on the performance of neural network training, starting with a variety of low-quality image sets. In the first three tables (Table 1, Table 2, Table 3) we expanded the training set by adding images of medium and high quality.

An interesting aspect was that the optimal expansion of the training set differed for different image qualities. For example, the best results for a medium quality set were achieved by adding 0.2 data fractions, while for high quality, an expansion of 0.5 was found to be optimal.

The impact of adding data on performance metrics is also noticeable.

Stability and variability of metrics were also considered. In some cases, the metrics changed more smoothly, while in other cases they were more variable, which may be due to the characteristics of the training set and the quality of the added images.

It is interesting to note that the best performance for each image quality was achieved with different data addition values. This can highlight the importance of taking into account the specifics of the training set when choosing a data augmentation strategy for training a neural network.

Conclusion.

The present study examined the influence of image quality on the learning process of neural networks, highlighting this issue in the context of expanding the training set by adding data of different quality. The experimental results highlight the importance of choosing a data expansion strategy wisely to optimize the efficiency of a neural network.

In the process of training on low-quality images, optimal values of the expansion parameter were identified, at which the best metrics were achieved, such as Precision, Recall and F1-score. It was found that adding medium and high quality data improves the generalization ability of the neural network, but the optimal parameter values may vary depending on the initial quality of the training set. It is important to note that our study emphasizes not only the importance of qualitatively expanding the training set, but also the impact of this process on the final metrics of the neural network. Changes in metrics with the addition of data of varying quality were discussed, as well as the stability and variability of these metrics depending on the amount of expansion.

Based on the results, researchers and practitioners are advised to carefully analyze the characteristics of their training sets and carefully select data enhancement parameters to achieve maximum neural network performance in image classification tasks of varying quality. Additional research in this area, including the use of larger datasets (also equal in size for all three categories) and a variety of neural network architectures, can deepen the understanding of the impact of image quality on the learning processes of neural networks and contribute to the development of effective methods for learning from data of various characteristics.

REFERENCES

[1]. Oncel, S. S. *Green energy engineering: Opening a green way for the future*. Journal of cleaner production, 2017, 142, 3095-3100. https://doi.org/10.1016/j.jclepro.2016.10.158

[2]. Gillingham, K., & Stock, J. H. *The cost of reducing greenhouse gas emissions.* Journal of Economic Perspectives, 2018, 32(4), 53-72. https://doi.org/10.1257/jep.32.4.53

[3]. Baghbani, S. S. *Power losses due to dirty solar panels, Cleaning Mechanism of Solar Photovoltaic Panel: A review*. Journal of Mechanical Research and Application, 2021.

[4]. Supe, H. et al. *Google earth engine for the detection of soiling on photovoltaic solar panels in arid environments.* Remote Sensing, 2020, 12(9), 1466. https://doi.org/10.3390/rs12091466

[5]. Sulaiman, S. A., Singh, A. K., Mokhtar, M. M. M., & Bou-Rabee, M. A. *Influence of dirt accumulation on performance of PV panels.* Energy Procedia, 2014. 50, 50-56. https://doi.org/10.1016/j.egypro.2014.06.006

[6]. Dantas, G. M., Mendes, O. L. C., Maia, S. M., & de Alexandria, A. R. *Dust detection in solar panel using image processing techniques: A review.* Research, Society and Development, 2020, 9(8), e321985107-e321985107. https://doi.org/10.33448/rsd-v9i8.5107

[7]. Abuqaaud, K. A., & Ferrah, A. *A novel technique for detecting and monitoring dust and soil on solar photovoltaic panel.* Advances in Science and Engineering Technology International Conferences (ASET), 2020 (pp. 1-6). IEEE. https://doi.org/10.1109/aset48392.2020.9118377

[8]. Olorunfemi, B. O., Nwulu, N. I., & Ogbolumani, O. A. *Solar panel surface dirt detection and removal based on arduino color recognition.* MethodsX, 2023, 10, 101967. https://doi.org/10.1016/j.mex.2022.101967

[9]. Onim, M. S. H. et al. *Solnet: A convolutional neural network for detecting dust on solar panels.* Energies, 2022, 16(1), 155. https://doi.org/10.3390/en16010155

[10]. Aji, B. H., Prasetya, S., & Hidayati, N. *Smart detection of solar panel cleanliness using artificial intelligence for the remote area maintenance.* In AIP Conference Proceedings, 2023 (Vol. 2601, No. 1). AIP Publishing. https://doi.org/10.1063/5.0130130

[11]. O'Shea K. & Nash R. *An introduction to convolutional neural networks.* arXiv preprint, 2015 arXiv:1511.08458.

[12]. https://www.kaggle.com/datasets/hemanthsai7/solar-panel-dust-detection

Received: 05.03.2023
Accepted: 06.12.2023