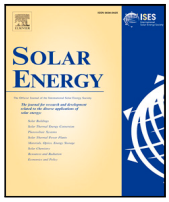
Solar Energy 265 (2023) 112089

Contents lists available at ScienceDirect

Solar Energy

journal homepage: www.elsevier.com/locate/solener



Optimized estimator of the output power of PV cells using EL images and I–V curves

Héctor Felipe Mateo Romero [a], Luis Hernández-Callejo [b,*], Miguel Ángel González Rebollo [a], Valentín Cardeñoso-Payo [c], Victor Alonso Gómez [d], Jose Ignacio Morales Aragonés [d], Ranganai Tawanda Moyo [e]

[a] Departamento Fisica de la Materia Condensada, Universidad de Valladolid, Valladolid, 47011, Spain
[b] Departamento Ingeniería Agrícola y Forestal, Universidad de Valladolid, Valladolid, 47002, Spain
[c] Departamento Informatica, Universidad de Valladolid, Valladolid, 47011, Spain
[d] Departamento de Física, Universidad de Valladolid, Valladolid, 47002, Spain
[e] Department of Mechanical Engineering, Durban University of Technology, Durban, P.O. Box 1334, South Africa

ARTICLE INFO

Keywords:
Electroluminescence image
Random Forest
Gradient Boosting
Recurrent Neural Network
Convolutional Neural Network
Regression

ABSTRACT

In this work, a method to predict the output power of Photovoltaic (PV) cells using their Electroluminescence (EL) images is presented. The data used includes Electroluminescence Images and the value of the Max Power Point computed from the Current–Voltage Curve of the cells. The method is used as follows: Firstly, the images are preprocessed to improve their quality. After that, a comparison between different Machine Learning methods from Traditional ones, such as Random Forest or Gradient Boosting, to Deep Learning methods, such as Recurrent Neural Networks or Convolutional Neural Networks is performed. Another significant contribution of this paper is that it analyzes the problem of unbalanced data, trying to solve it using Synthetic Images created by a Generative Adversarial Network. Our results show that the best model is the Gradient-Boosting based method using a pre-trained Resnet50 as a feature extraction method with a Mean Absolute Error (MAE) of 0.0341 and a Mean Squared Error (MSE) of 0.00211. The results also shows how the models trained with the unbalanced dataset are capable of obtaining results similar to the models trained with the balanced dataset.

1. Introduction

In nowadays world, the energy production faces two big problems. First, the cost of producing energy is on a steady rise due to the lack of fossil fuels and the high price of the available resources. Another important problem is that these kinds of energy have played a significant role in polluting the environment, which has resulted in climate change [1]. Due to these two problems, the use of alternative energies has been increasing in recent years. Among these energies, the use of Solar Energy is one of the most effective and most used due to its availability[2].

Photovoltaic (PV) plants are one of the most important alternatives [3] to mine solar energy [4], due to their availability, affordability, and share market [3]. PV plants gather hundreds to thousands of PV modules, each one made of a set of PV cells, depending on the module configuration. The amount of PV cells depends on the configuration of the module. The maintenance of these plants is generally highly complex. The production of PV cells is affected by different aspects such as weather, soiling, defects, etc. Another important factor is that some defects cannot be found by visual inspection; techniques such as Thermography or Electroluminescence (EL) are necessary to detect them. Thermography measures the infrared irradiation of the module and produces an image of it. However, Electroluminescence (EL) measures the amount of light a module emits when injected with electric current. More information about these techniques can be found in the bibliography [5,6].

Artificial Intelligence Techniques have been used to optimize the production of PV systems [7,8]. Different problems have been addressed, such as the Max Power Point Tracking, the estimation of the parameters of the model for simulating, the forecasting of the production, or the detection of faults in the cells.

This work focuses on using Artificial Intelligence to build a model with the capacity to estimate the performance of PV cells from its

* Corresponding author.
E-mail addresses: hectorfelipe.mateo@uva.es (H.F. Mateo Romero), luis.hernandez.callejo@uva.es (L. Hernández-Callejo), mrebollo@uva.es (M.Á. González Rebollo), valentin.cardenoso@uva.es (V. Cardeñoso-Payo), victor.alonso.gomez@uva.es (V. Alonso Gómez), joseignacio.morales@uva.es (J.I. Morales Aragonés), moyoranganai@gmail.com (R.T. Moyo).

https://doi.org/10.1016/j.solener.2023.112089
Received 22 June 2023; Received in revised form 10 September 2023; Accepted 6 October 2023
Available online 18 October 2023

Table 1
Characteristic of the solar cells used in the experiments.

Characteristic	Value
Max power	4.67 W
Open-Circuit voltage	0.645 V
Short-Circuit current	8.99 A
Fill factor	0.81
Efficiency	19%
Number of busbars	4
Size	156 mm × 156 mm
Thickness	200 μm
Technology	Polycrystalline

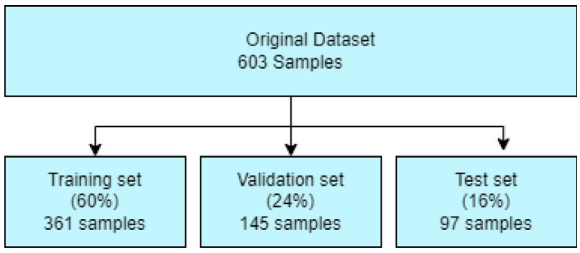


Fig. 1. Distribution of the samples of the dataset.

EL image. Different works [9–11] tackle the issue of finding defects in PV cells or modules obtaining accuracy values of near 99% [7]. Still, their limitation is that they do not directly consider the impact of these defects on their performance. This can be an issue since the final objective of these kinds of methods is to detect bad-performing modules. The main reason for this problem is that it is quite complex to measure the performance of a single cell by measuring its I–V curve since it is usually installed in the modules, and it is not possible to remove them without affecting the performance of the installations since their contacts are welded.

Using the performance of the modules is a more direct metric of their performance and state. Other works have tackled this issue by using machine learning to predict the power produced by the cell [12,13], but their training data has not been measured directly since they are not able to measure the individual I–V curve of each of these cells.

The main contribution of this paper is the creation of an Artificial Intelligence Model that is able to predict the power of the cell using EL. This model has been trained with data obtained directly from the I–V curves of the cells. The I–V curves were measured individually for each cell, using the device described in the work by J. I. Morales-Aragones in [14]. Different models were trained with the data and the images to predict the power of the cells, addressing the problem as a regression problem. This paper also tackles the issue of unbalanced data by analyzing the effects of balancing the dataset with synthetic data[15]. This is a step forward from other works since the IV curve provides direct information about the performance (The energetic production) of the PV cell.

The paper is structured as follows: First, it describes the instrumental setup for taking the EL images and the I–V curves of the cells in Section 2.1, after that, it explains the methodology that it has been followed for preprocessing the data (Section 2.2), for balancing the data (Section 2.4), choosing the metrics (Section 2.5) and designing the models (Section 2.6). It also explains the feature extraction method in Section . Finally, Section 3 analyzes the results of the experiments and explains the conclusions and the future work in Section 4.

2. Data and methodology

This section will explain the methodology of the different processes of the experiments: data gathering, data preprocessing, data labeling, balancing of the data, metrics selection, feature extraction, and hyperparameter tuning of the different models.

2.1. Data

The dataset is composed of 603 EL measurements (Fig. 1), including cells with different defects or artificial masks. The cells used in the experiments are individual and are not a part of any solar module. Table 1 presents the most important characteristics provided by the manufacturer.

The shadowing in the cells was performed using cardboard masks with a layer of aluminum. The zones covered by the mask do not contribute to the production of current since the aluminum foil is capable of blocking photons. This is clearly visible in the EL image, in the area shadowed by the mask, and in the I–V curve, since the mask reduces the current production of the cell.

These masks have been placed on the surface of the cell. Overall, there are 8 kinds of masks or defects that imitate problems found in real cells (Fig. 2). The images have been taken with different values of polarization current (Fig. 5).

The images were captured using an "InGaAs C12741-0" silicon detector camera with an 8 mm focal length lens and an "f" number of 1.4. At a PV cell level, each EL measurement was taken consecutively as the measurement of the I–V curve was carried out, in the same experimental conditions. . For these measurements, it was used the I–V measurement device created in [14] (Fig. 4) This device is capable of measuring the I–V curve of a single cell when its contacts are available, which is not the case in typical solar modules but it is possible when the cell is inserted into a module. The illumination for the measurements was provided by an LED infrared panel. The information about the individual LEDs and the panel can be found in Table 2

The temperature of the cells was also monitored. All of the measurements were taken at similar temperatures.

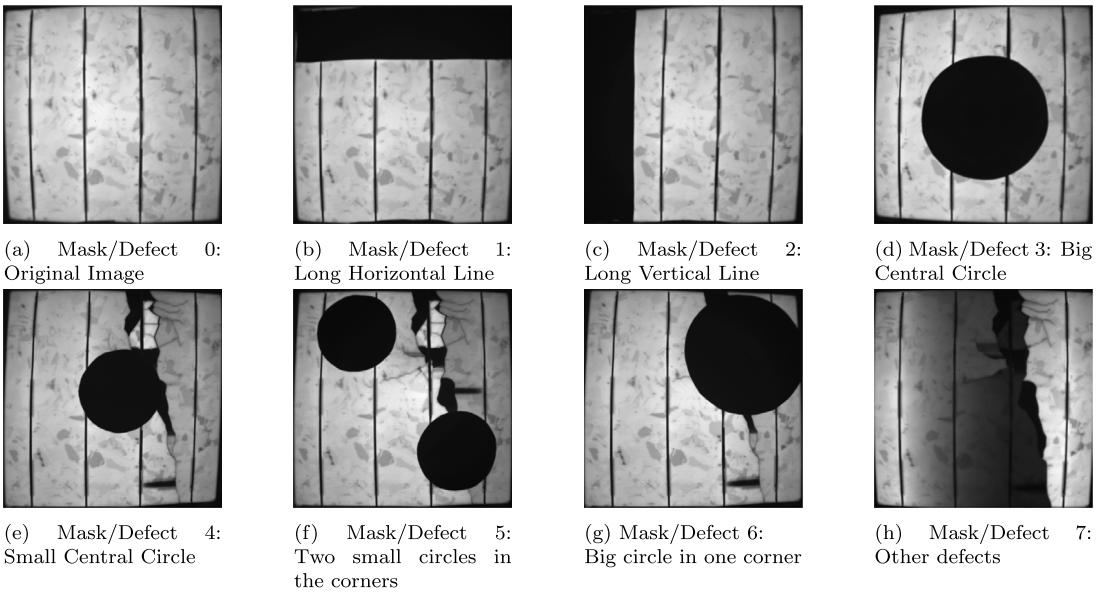
In addition to the EL measurements of the cells, some synthetic images were included. These images were created in the previous works of research group [15]. This dataset comprises 10,000 images of EL measurements of cells with a certain value of power associated with each one. was predicted using a Random Forest algorithm, which is one of the most well-known ensemble-based approaches (see reference for more details). This synthetic dataset was used to balance the original dataset (Fig. 3).

2.2. Data preprocessing

The measurements that were made required a complex process to improve their quality. This process helped the models to manage the images. The method of preprocessing included the following processes (Fig. 6):

- Removing dead pixels and image noise caused by camera deterioration and tiny amounts of external light entering the chamber when the PV cell was enclosed, which was not a perfect light-tight chamber.
- Increasing the brightness and contrast of the digital values of the pixels of the images through min–max normalization in order to improve the visibility of the image.
- Fixing the distortion produced by the lens of the camera and removing the surrounding black areas. These processes crop the image to a square closer to the area occupied by the cell. Removing the parts that do not provide information and ensuring a better correspondence of the geometry of the images with the one of the measured cells.

A more detailed explanation of these processes can be found in our previous work [15].



(a) Mask/Defect 0: Original Image
(b) Mask/Defect 1: Long Horizontal Line
(c) Mask/Defect 2: Long Vertical Line
(d) Mask/Defect 3: Big Central Circle
(e) Mask/Defect 4: Small Central Circle
(f) Mask/Defect 5: Two small circles in the corners
(g) Mask/Defect 6: Big circle in one corner
(h) Mask/Defect 7: Other defects

Fig. 2. Different Masks or Features presented in the datset.

Table 2
Information about the LEDs and the array.

Characteristic	LED	Characteristic	Panel
Dimensions	3,85 × 3,85 × 2,41 mm	Dimensions	150 × 150 mm
Operation temperature	−40 to 125 °C	Wave length	850 nm
Direct voltage	1.5 V	Excitation voltage	21 V
Direct current	1000 mA	Direct current	3 A
Radiation peak	860 nm	Electric power	43 W
Radiation center	850 nm	Branches in parallel	3
–	–	LEDs in series	14

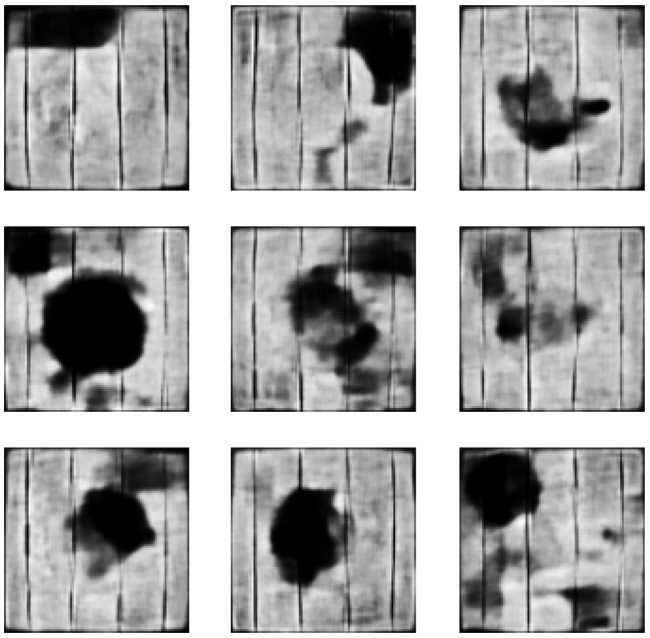


Fig. 3. Examples of generated images.

2.3. Calculation of the power

Along with the image, the I–V curve of each measurement was taken. With this data, it is possible to find the Maximum Power Point (MPP) of each measurement. This MPP depends on the amount of light emitted to the cell, the state of the cell, and its temperature. In Fig. 7, the I–V curve of a cell can be observed along with the computer Power–Voltage curve and the position of the MPP.

After calculating the MPP for each cell, it was necessary to standardize the values since these values depended on the amount of light that was used to take the measurements. The following methodology was used to standardize the value:

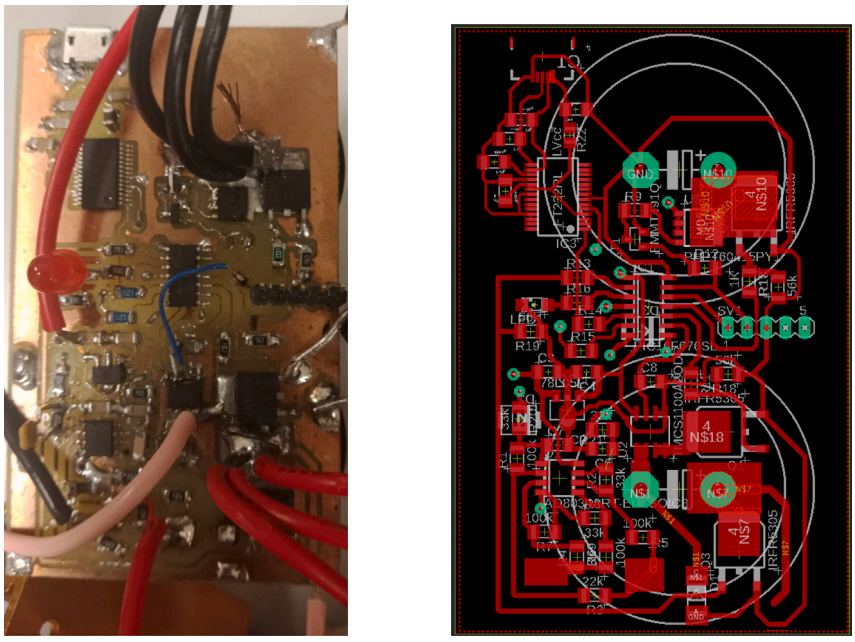
- Different groups of cells were created according to the irradiance that was used to take the measurement.
- For each one of the groups, the 5 highest values of the MPP are obtained.
- The mean of the 5 values is computed. This will reduce the problems for extreme values that can be caused by problems in the measurements.
- MPP of each of the cells is divided by the mean computed of its group.

This process results in a variable that is independent of the irradiance. Low values will correspond with cells with low output power. High values indicate that the cell operates according to its theoretical capacities. The values of power of the synthetic images were computed using a Random Forest algorithm trained with the values of the real data; more information can be found in the original article [16].

2.4. Balancing of the data

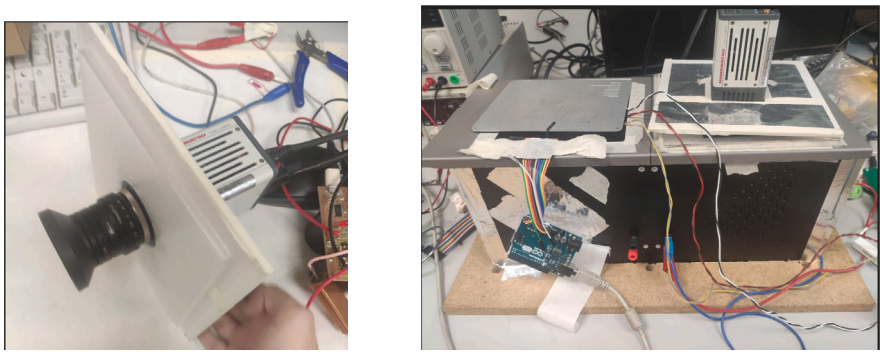
As shown in Fig. 8(a), the distribution of cells according to their maximum power is not balanced. This is caused by the nature of the problem. It is easier to find cells in a quite eroded state or with some defects than cells completely broken or in a perfect state.

As seen in other problems [17–19], the unbalance of the datasets is a common problem in machine learning that needs to be tackled



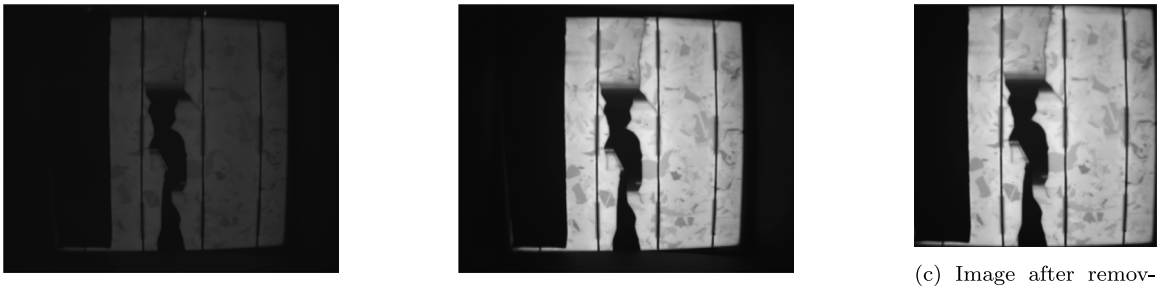
(a) Picture of the real device. (b) Schematics of the device circuitry.

Fig. 4. I–V-tracer.



(a) InGaAs Camera. (b) Capture set-up.

Fig. 5. Devices used to obtain the images.



(a) Original Image (b) Image after the standardization (c) Image after removing the black surrounding areas

Fig. 6. Image before and after applying the explained process (the process explained in the text: removing noise, adjusting contrast and brightness, and modifying image geometry to eliminate black surrounding areas and deformation.

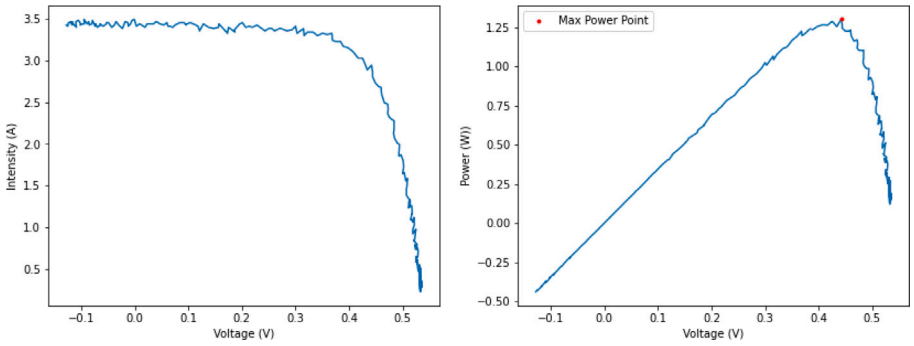
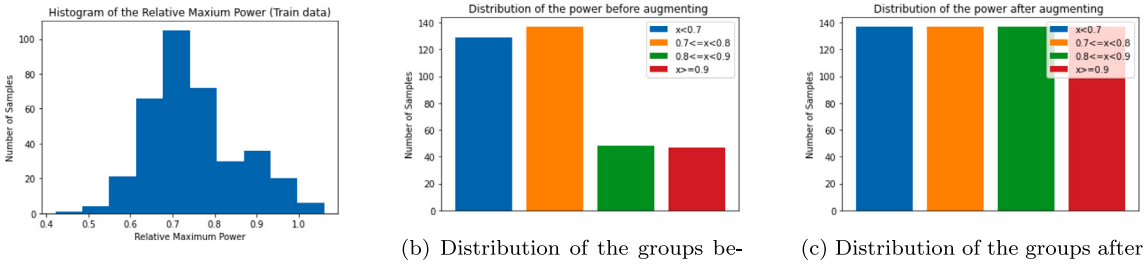


Fig. 7. Curves current–voltage and power–voltage.



(a) Distribution of the power

(b) Distribution of the groups before the augmentation

(c) Distribution of the groups after the augmentation

Fig. 8. Histogram of the distribution of the images before and after the balancing.

following a tailored strategy for each problem. Synthetic images were included to solve this problem. The process that was used can be summarized in these steps:

- The original dataset is divided into four different subgroups according to their power (Group 0: $x < 0.7$, Group 1: $0.7 \leq x < 0.8$, Group 2: $0.8 \leq x \leq 0.9$, Group 3: $x > 0.9$) (Fig. 8(b))
- For each group, synthetic images with a power value in the same range as the others of the group are added until all of the groups are of the same size as the biggest group.

This procedure helps to have a more balanced dataset (Fig. 8(c)). Different combinations of groups and sizes were tested, and it was found that 4 groups using the biggest size and maximum size gave the best results in our tests. The balancing had only been applied to the training data.

2.5. Metrics

Common metrics for regression include Mean Squared Error (MSE) and Mean Average Error (MAE).

MAE : $\sum_{i=1}^{D} |x_i - y_i|$

MSE : $\sum_{i=1}^{D} (x_i - y_i)^2$

With x being the real value and y the predicted value MSE, due to its definition, gives more importance to the outliers, performing better for underrepresented cases and worse for overrepresented cases. This also helps to deal with the unbalance of the data. More information about metrics can be found in the bibliography [20].

2.6. Experiments

Different models have been tested to solve the problem tackled in this article, which is important since, as it has been commented, there are no other works in the bibliography that tackles directly this problem. For each one of them, the issue of the extraction of features for training the model and the optimization of the architecture and hyperparameters has been addressed.

2.6.1. Traditional methods

This section proposes a baseline approach to estimate the power from a single EL image. Different traditional methods have been used for regression problems [21]. From the enormous amount of different techniques, Random Forest [16] and Gradient boosting [22] were chosen due to their good performance in our preliminary tests and their low computational cost.

- Random Forest [16]: is a widely-used algorithm that combines the output of many decision trees in order to reach a single output.
- Gradient Boosting [22]: is a popular ensemble algorithm that trains the models sequentially, each new one trying to improve the previous one.

The algorithms were trained with features extracted from the EL images. The implementation used can be found in the Sklearn [23] library for both of the methods.

Feature extraction. To apply power estimation using traditional methods, it is not possible to simply use the images. It is necessary to extract information from the images. Two different approaches have been used to perform this task:

- Manual extraction: The histogram of an EL image cell (see Fig. 9) usually presents two different peaks; the position of these peaks is related to the amount of white and dark areas in the image. Another important feature is the number of dark regions. These zones are related to the parts of the cell that are not working right [24]. The areas with pure white are usually also harmful since they are related to other kinds of defects. It also included information about the roughness of the curve and different statistical metrics such as mean, variance, and standard deviation. These features represent the most important characteristics of the EL images (Table 3).
- Resnet extraction: Pre-trained convolutional networks for feature extraction are being used in different areas with excellent results. They have also been used in topics related to EL images of solar cells. It has been shown that models trained on the ImageSet [25]

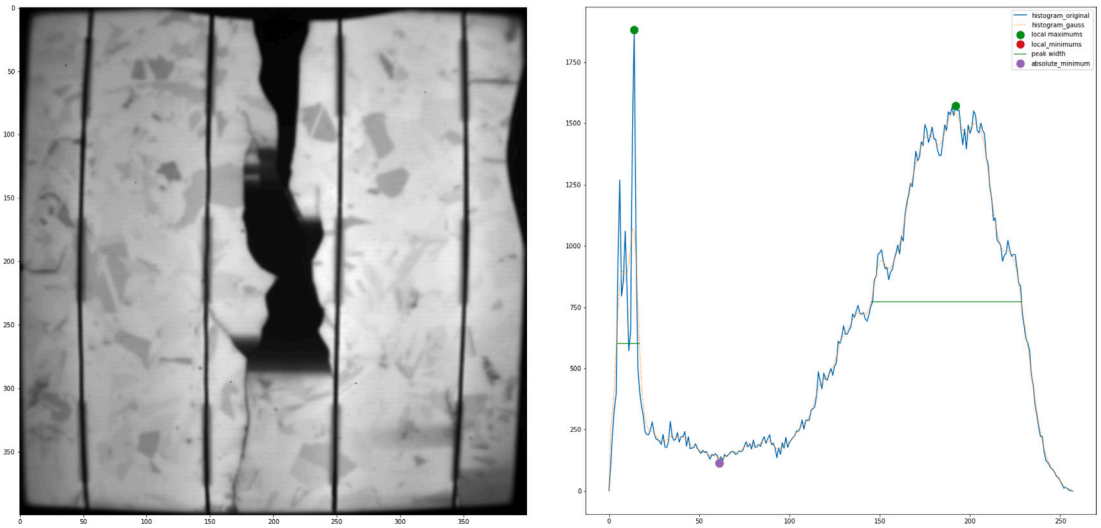


Fig. 9. Color histogram of a cell. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 3
Manually selected features. Mean, median, mode, variance, roughness and standard deviation of all the values of the histogram. Number of pixels with values of blacks, very bright whites, and other values. Number of peaks in the histogram and their height and width.

Mean	Median	Mode	Variance	std
Roughness	Blacks	Burned whites	Others	peaks_number
Peaks distance	Peak 0 height	Peak 0 width	Peak 1 height	Peak 1 width

dataset are advantageous since they have an extraordinary capacity for finding features in other images and problems [26,27]. One of the most popular architectures is Resnet [28] which presents a configuration between 18 and 152 layers. For our problem, it has been chosen the Resnet50 in its implementation found in Tensorflow Library [29]. Tensorflow is a widely used library used to create any kind of neural network, with a high variety of layers (Deep, Convolutional, Recurrent) available to use. To compute the features, It was necessary to convert the measurement to RGB images and resize them according to the specifications of the network.

Hyperparameter tuning. The tuning of the hyperparameters it is critical in Machine Learning problems since the correct adjustment of the hyperparameters can improve greatly the results of the models since it can be a complex problem in models with a high amount of parameters. Different algorithms have been created in order to automatize it. This work has been done with GridSearchCV using the Sklearn library [23]. GridSearch does an exhaustive search over the specified parameters evaluating the results according to the metrics of the model. The results of the optimization for RF can be found in Table 4 and the results for GB in Table 5.

2.6.2. Recurrent Neural Networks (RNNs)

The two methods based on Recurrent Neural Networks (RNN) have been tested. The main characteristic of these networks is that the nodes can form cycles, enabling them to exhibit temporal dynamic behavior. One of the methods is based on Long Short Term Memory (LSTM) [30], and the other is based on Gated Recurrent Unit (GRU) [31,32]. The implementation was made using Tensorflow [29].

Feature extraction. As in the traditional methods, recurrent Neural Networks are not able to work with the image itself. It is necessary to extract features from the original images. The same process as in the previous subsection was followed, using manual feature extraction and feature extraction based on Resnet50.

Optimization. Different optimization methods have been proposed in the literature. Adam [33] was used as the optimizer since it is one of the most used and it has a good performance.

The optimization of the hyperparameters is one of the most complex processes in the design of Deep Learning Models. A manual process is not viable since the number of parameters and their values to optimize is usually extremely high. To solve this problem, it has been chosen the library Keras-Tuner [34]. This library includes different optimizers, one of which is the Bayesian Optimization Tuner [35]. This algorithm takes into account the results of past evaluations when choosing the hyperparameter to evaluate next. This informed way of choosing parameters improves the search in the most promising area of the search space. It also disregards the less promising areas which reduces the iterations needed to find the optimal set of parameters

Different parameters were chosen to be optimized: the learning rate, the batch size, the number of hidden layers, and the number of nodes in each layer, adding 16 each time. The optimization process was run for 32 iterations.

Adan was used as the optimizer. The best results can be found in Table 6.

Architecture. The architecture of the best model according to the tuning process can be found in Fig. 10.

2.6.3. Convolutional Neural Networks (CNNs)

This section introduces the CNN-based approach to predict the power of a cell from its EL image. CNNs have the capacity to deal with images since they can apply Feature Extraction with their convolutional layers. Two different approaches are presented: One training a new CNN from scratch and one using transfer learning from a Resnet50 [28]. Tensorflow is used for implementing the models.

Preprocessing. Along with the processes discussed in Section 2.2, it was important to reduce the size of the images to 200 × 200, to decrease the computational cost of the algorithm.

Architecture. The architecture of the neural network was chosen using a manual process. The networks were trained during 500 epochs, and the best model found in the training of each option was considered the final one. The findings showed that the best one had 2 convolutional layers of 64 and 128 units and one dense layer of 256 units. (Fig. 11).

This architecture was chosen after doing a trial-error process with different architectures trying to minimize the validation error, and it was not possible to use an automatic optimizer due to limitations in the hardware used for the experiments.

Table 4
Best hyperparameters found for Random Forest: n_estimators ∈ [100, 1000], max_depth ∈ [2, *None*], min_samples_leaf ∈ [1, 10], min_samples_leaf ∈ [1, 10], min_weight_fraction_leaf ∈ [0, 0.8].

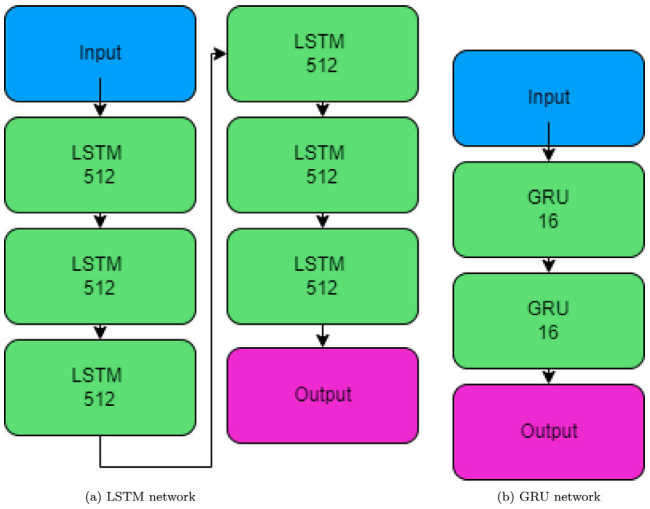
n_estimators	max_depth	min_samples_leaf	min_samples_leaf	min_weight_fraction_leaf
750	None	1	3	0

Table 5
Best hyperparameters found for Gradient Boosting: n_estimators ∈ [100, 500], max_depth ∈ [2, *None*], learning_rate ∈ [0.01, 0.001].

n_estimators	max_depth	learning_rate
275	None	0.08

Table 6
Best hyperparameters found for Recurrent Neural Networks: num_layers ∈ [2, 3, 4, 5, 6], num_units ∈ [16, 32, 64, 125, 256, 512], batch_size ∈ [16, 32, 64, 80, 96, 112, 128], learning_rate ∈ [0.001, 0.00001].

RNN	num_layers	num_units	batch_size	learning_rate
LSTM	6	512	16	0.000219
GRU	2	16	16	0.001



(a) LSTM network (b) GRU network

Fig. 10. Final architectures of the recurrent networks.

Table 7
Best hyperparameters found for Convolutional Neural Networks: batch_size ∈ [16, 32, 64, 80, 96, 112, 128], learning_rate ∈ [0.001, 0.00001].

batch_size	learning_rate
16	0.00005

Optimization. Bayesian Tuner from Keras-Tuner was also used in the experiments with RNNs, optimizing the following parameters the learning rate and the batch size. The optimization process was run for 32 iterations. Adam was used as the optimizer. The best results can be found in Table 7.

3. Results and discussion

This section assesses the performance of the methods for predicting the power of a cell by comparing different approaches in terms of their performance. Each experiment was repeated 5 times using the parameters found in the hyperparameter optimizations. This ensured the stability and quality of the results of each method. The experiments were performed with an AMD Ryzen 7 5800H, a GTX 1650 of 4 GB, and 16 GB of RAM.

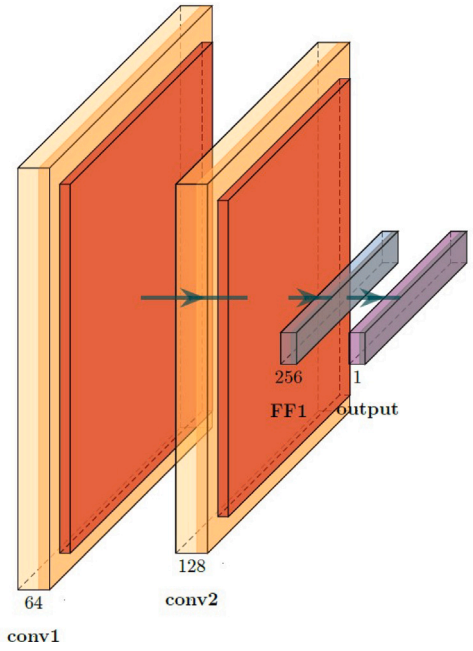


Fig. 11. Final architecture.

3.1. Training only with real images

The results of the experiments using only the original dataset can be found in Table 8.

The results show that CNN has the best performance in terms of MAE, but RF is better when the MSE is taken into account. Recurrent Networks methods perform considerably worse than the other methods. It can be seen that the results are notably good even when the problem of unbalanced data is presented.

3.2. Training with original + synthetic images

The results of the experiments balancing the original dataset with synthetic images can be found in Table 9.

The results of the experiments with the original dataset and the balanced are quite similar. In the methods that use features, there is a low increase in both errors, but in CNN, which uses the images, there is a low decrease in the errors. Two different facts can be concluded from these results: that the manual feature extraction is not good enough for this problem or that the unbalance of the original dataset is not worsening the results of the models. This first hypothesis will be verified by using the feature extraction methods presented in the methodology section.

3.3. Using resnet

For this experiment, only the two most promising feature-based methods from the previous section are considered (GB and RF). The results of both methods are compared using the original dataset and the balanced dataset.

Table 8
Results using the original dataset.

Metric	MAE			MSE		
Data	Train	Validate	Test	Train	Validate	Test
GB	0.00222	0.03674	0.03667 ± 0.00148	4.67e−05	0.00224	0.00290 ± 0.00006
RF	0.01432	0.03821	0.03721 ± 0.00179	0.00287	0.00227	**0.00285 ± 0.00002**
LSTM	0.03682	0.04602	0.04397 ± 0.00386	0.00257	0.00343	0.00354 ± 0.00027
GRU	0.03805	0.04615	0.04637 ± 0.00402	0.00258	0.00343	0.00386 ± 0.00008
CNN	0.01181	0.03328	**0.03630 ± 0.00160**	0.00039	0.00202	0.00301 ± 0.00005

Table 9
Results using the balanced dataset.

Metric	MAE			MSE		
Data	Train	Validate	Test	Train	Validate	Test
GB	0.00233	0.03647	0.03708 ± 0.00316	8.62e−05	0.00206	**0.00282 ± 0.0006**
RF	0.00980	0.03712	0.03747 ± 0.00394	0.00980	0.002155	0.002935 ± 0.0001
LSTM	0.01998	0.04497	0.04068 ± 0.00340	0.00114	0.00332	0.00349 ± 0.00008
GRU	0.02582	0.04601	0.04870 ± 0.00439	0.00149	0.00357	0.00430 ± 0.00004
CNN	0.01071	0.03255	**0.03407 ± 0.00255**	0.00034	0.00192	0.00284 ± 0.00005

Table 10
Results of using Resnet for feature extraction (R): Original Dataset, (S): Balanced Dataset.

Metric	MAE			MSE		
Data	Train	Validate	Test	Train	Validate	Test
GB (R)	0.001000	0.032786	**0.034277 ± 0.00293**	1.63e−05	0.003078	**0.002271 ± 0.0001**
RF (R)	0.009926	0.035272	0.034684 ± 0.00251	0.000202	0.003469	0.002332 ± 0.0001
GB (S)	0.001957	0.032993	**0.034143 ± 0.00307**	5.12e−05	0.003183	**0.002116 ± 0.0001**
RF (S)	0.009926	0.034884	0.034559 ± 0.00248	0.000202	0.003444	0.002319 ± 0.0001

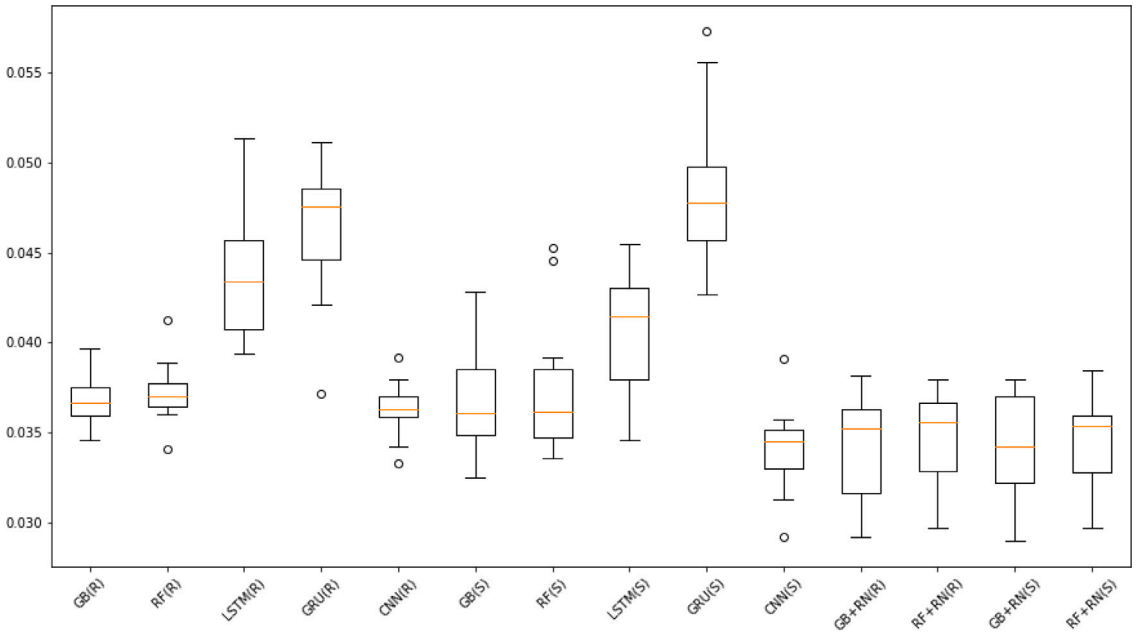


Fig. 12. Box diagram of the experiments.

It can be observed in Table 10 that GB and RF perform better with this feature extraction method. Their results are even better than CNN in terms of MSE, but they perform a bit worse in terms of MAE.

GB is considered the best model for two main reasons: MSE is a better metric than MAE in this problem since MSE gives more importance to the outliers, which helps to deal with problems that have a problem of unbalance in the data. The other reason is that the computation cost of GB [36] is considerably lower than CNN [37] even after applying Resnet to extract features.

Fig. 12 presents a summary of all of the experiments.

It can also be observed that the use of synthetic images for balancing the dataset does not make a great change in the results of the models but they do not make them worse. It is therefore concluded that the problem of the unbalance of the dataset is not damaging the performance of the models. This also reassures the quality of the synthetic images.

Fig. 13 shows the relationship between the distance real valor and predictions. Almost all the samples have a distance between $[-0,1,0.1]$ with only a few outliers in the most extreme values. This shows that the method is not biased toward certain values.

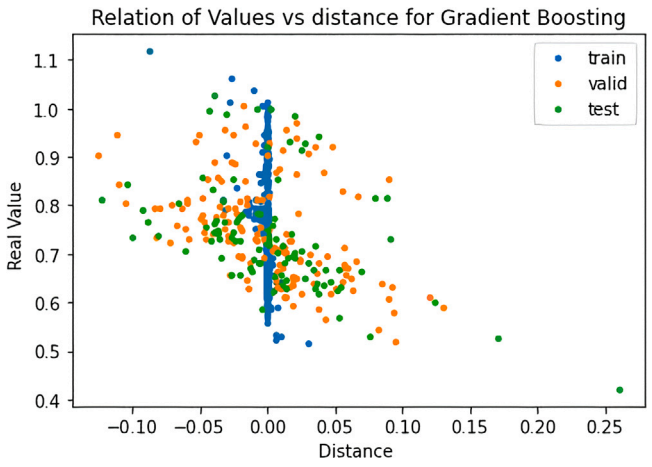


Fig. 13. Relation between the values and error.

4. Conclusions and future work

The detection of the output power of PV cells is one of the most critical research issues in the PV sector since it can analyze their production directly. Still, it is quite complex due to the difficulty of obtaining the individual I–V curves of the cells. In an attempt to design an enhanced prediction system, different models are presented and compared in the paper. The effect of an unbalanced dataset has also been analyzed. Synthetic images have been used to try and fix the problem. The results have shown two important conclusions: The Gradient Boosting model is the best-performing method when combined with a robust feature extraction method such as Resnet50, with an MAE of 0.0341 and MSE of 0.0021. It has also been observed that the unbalance in the dataset is not as critical as it can be seen. The method trained with the unbalanced dataset performs similarly to the methods with the balanced dataset.

The research objectives have been fulfilled since it has been shown that it is possible to create a model capable of predicting the output power of a PV cell using the information from the EL images. It is extremely important to remark on the importance of using the I–V curve to obtain the values of the labels since this information is an objective measurement of the performance of the cell. This is different from approaches presented in most of the works of the biography, where they do not have the possibility of measuring the I–V at the cell level. Even with the good results, the method has some flaws that should be addressed in the future works: The amount of data is quite limited, so new images with different defects would improve the generalization and performance of the model. Another interesting option would be to upgrade the preprocessing of the images to improve the amount of information that can be obtained from them. The applicability of the model is quite limited at this point of the research; for applying it to real-time applications, it would need a constant way of performing EL captures of the PV modules.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements and funding

This study was supported by the Universidad de Valladolid, Spain with the predoctoral contracts of 2020, co-funded by Santander Bank, Spain. This work has been financed also by the Spanish Ministry of Science and Innovation, Spain, under project PID2020-113533RB-C33. This study was also supported by the Universidad de Valladolid, Spain with ERASMUS+ KA-107. We also appreciate the help of other members of our departments.

References

[1] A.J. McMichael, R.E. Woodruff, S. Hales, Climate change and human health: Present and future risks, Lancet 367 (9513) (2006) 859–869, http://dx.doi.org/10.1016/S0140-6736(06)68079-3.

[2] I. – International Renewable Energy Agency, FUTURE OF SOLAR PHOTOVOLTAIC Deployment, investment, technology, grid integration and socio-economic aspects A Global Energy Transformation paper About IRENA, 2019, URL www.irena.org/publications.

[3] R. Adib, A. Zervos, M. Eckhart, M.E.-A. David, H. Kirsty, H. Peter, R. Governments, F. Bariloche, Renewables 2021 Global Status Report, REN21 Renewables Now, 2021, URL https://www.ren21.net/wp-content/uploads/2019/05/GSR2021_Full_Report.pdf.

[4] A. Herez, M. Ramadan, B. Abdulhay, M. Khaled, Short review on solar energy systems, AIP Conf. Proc. 1758 (1) (2016) 030041, http://dx.doi.org/10.1063/1.4959437, arXiv:https://pubs.aip.org/aip/acp/article-pdf/doi/10.1063/1.4959437/13080260/030041_1_online.pdf.

[5] L. Koester, S. Lindig, A. Louwen, A. Astigarraga, G. Manzolini, D. Moser, Review of photovoltaic module degradation, field inspection techniques and techno-economic assessment, Renew. Sustain. Energy Rev. 165 (2022) http://dx.doi.org/10.1016/j.rser.2022.112616.

[6] L. Koester, S. Lindig, A. Louwen, A. Astigarraga, G. Manzolini, D. Moser, Review of photovoltaic module degradation, field inspection techniques and techno-economic assessment, Renew. Sustain. Energy Rev. 165 (2022) 112616, http://dx.doi.org/10.1016/j.rser.2022.112616, URL https://www.sciencedirect.com/science/article/pii/S136403212200510X.

[7] H.F. Mateo Romero, M.A. Gonzalez Rebollo, V. Cardenoso Payo, V. Alonso Gomez, A. Redondo Plaza, R.T. Moyo, L. Hernandez Callejo, Applications of artificial intelligence to photovoltaic systems: A review, Appl. Sci. 12 (19) (2022) http://dx.doi.org/10.3390/app121910056.

[8] J.F. Gaviria, G. Narvaez, C. Guillen, L. Giraldo, M. Bressan, Machine learning in photovoltaic systems: A review, Renew. Energy 196 (2022) http://dx.doi.org/10.1016/j.renene.2022.06.105.

[9] A. Kumaradurai, Y. Teekaraman, T. Coosemans, M. Messagie, Fault detection in photovoltaic systems using machine learning algorithms: A review, in: 2020 8th International Conference on Orange Technology (ICOT), 2020, http://dx.doi.org/10.1109/ICOT51877.2020.9468768.

[10] K. AbdulMawjood, S.S. Refaat, W.G. Morsi, Detection and prediction of faults in photovoltaic arrays: A review, in: Proceedings 2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018), in: Compatibility Power Electronics and Power Engineering, Inst Elect & Elect Engineers; IEEE Ind Elect Soc; Texas A & M Univ Qatar, 2018, pp. 1–8, http://dx.doi.org/10.1109/CPE.2018.8372609, 12th IEEE International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG), Doha, QATAR, APR 10-12, 2018.

[11] A. Mellit, G.M. Tina, S.A. Kalogirou, Fault detection and diagnosis methods for photovoltaic systems: A review, Renew. Sustain. Energy Rev. 91 (2018) 1–17, http://dx.doi.org/10.1016/j.rser.2018.03.062.

[12] M. Hoffmann, J. Hepp, B. Doll, C. Buerhop-Lutz, I.M. Peters, C. Brabec, A. Maier, V. Christlein, Module-power prediction from PL measurements using deep learning, in: Conference Record of the IEEE Photovoltaic Specialists Conference, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 827–830, http://dx.doi.org/10.1109/PVSC43889.2021.9519005.

[13] M. Hoffmann, C. Buerhop-Lutz, L. Reeb, T. Pickel, T. Winkler, B. Doll, T. Würfl, I.M. Peters, C. Brabec, A. Maier, V. Christlein, Deep-learning-based pipeline for module power prediction from electroluminescense measurements, Prog. Photovolt., Res. Appl. 29 (2021) 920–935, http://dx.doi.org/10.1002/PIP.3416.

[14] J.I. Morales-Aragonés, V.A. Gómez, S. Gallardo-Saavedra, A. Redondo-Plaza, D. Fernández-Martínez, L. Hernández-Callejo, Low-cost three-quadrant single solar cell I-V tracer, Appl. Sci. 12 (13) (2022) http://dx.doi.org/10.3390/app12136623.

[15] H.F. Mateo-Romero, L. Hernandez-Callejo, M.A.G. Rebollo, V. Cardeñoso-Payo, V.A. Gomez, H.J. Bello, R.T. Moyo, J.I.M. Aragones, Synthetic dataset of electroluminescence images of photovoltaic cells by deep convolutional generative adversarial networks, Sustainability 15 (9) (2023) 7175, http://dx.doi.org/10.3390/su15097175.

[16] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5–32, http://dx.doi.org/10.1023/A:1010950718922.

[17] B. Zhou, M. Ha, C. Wang, An improved algorithm of unbalanced data SVM, in: B. Cao, G. Wang, S. Guo, S. Chen (Eds.), Fuzzy Information and Engineering 2010, Vol 1, in: Advances in Intelligent and Soft Computing, vol. 78, Liaoning Technol Univ; China Operat Res Soc, Fuzzy Informat & Engn Branch; Int Inst Gen Syst Studies, Fuzzy Informat & Engn Branch, 2010, p. 549+, 5th Annual Conference on Fuzzy Information and Engineering, Huludao, PEOPLES R CHINA, SEP 23-27, 2010.

[18] W. Gan, J. Geng, Y. Wang, J. Xu, W. Yu, H. Yuan, R. Qin, Effects of unbalanced data on radiometric transforming model fitting for relative radiometric normalization, in: International Geoscience and Remote Sensing Symposium (IGARSS), Institute of Electrical and Electronics Engineers Inc., 2020, pp. 2316–2319, http://dx.doi.org/10.1109/IGARSS39084.2020.9324679.

[19] J.H. Xue, D.M. Titterington, Do unbalanced data have a negative effect on LDA? Pattern Recognit. 41 (5) (2008) 1558–1571, http://dx.doi.org/10.1016/J.PATCOG.2007.11.008.

[20] A. Botchkarev, A new typology design of performance metrics to measure errors in machine learning regression algorithms, Interdiscip. J. Inf. Knowl. Manag. 14 (2019) 045–076, http://dx.doi.org/10.28945/4184.

[21] A. Gron, Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, first ed., O'Reilly Media, Inc., 2017.

[22] J.H. Friedman, Greedy function approximation: A gradient boosting machine, Ann. Statist. 29 (5) (2001) 1189–1232, http://dx.doi.org/10.1214/aos/1013203451.

[23] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[24] M. Köntges, S. Kurtz, C. Packard, U. Jahn, K.A. Berger, K. Kato, T. Friesen, H. Liu, M. Van Iseghem, J. Wohlgemuth, et al., Review of Failures of Photovoltaic Modules, Tech. rep., IEA International Energy Agency, 2014.

[25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255, http://dx.doi.org/10.1109/CVPR.2009.5206848.

[26] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1717–1724, http://dx.doi.org/10.1109/CVPR.2014.222.

[27] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, Q. He, A comprehensive survey on transfer learning, Proc. IEEE 109 (1) (2020) 43–76.

[28] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[29] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-scale machine learning on heterogeneous systems, 2015, Software available from tensorflow.org. URL https://www.tensorflow.org/.

[30] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (1997) 1735–1780, http://dx.doi.org/10.1162/neco.1997.9.8.1735.

[31] K. Cho, B. Van Merriënboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: Encoder-decoder approaches, 2014, arXiv preprint arXiv:1409.1259.

[32] Y. Yu, X. Si, C. Hu, J. Zhang, A review of recurrent neural networks: LSTM cells and network architectures, Neural Comput. 31 (7) (2019) 1235–1270, http://dx.doi.org/10.1162/neco_a_01199, arXiv:https://direct.mit.edu/neco/article-pdf/31/7/1235/1053200/neco_a_01199.pdf.

[33] D. Kingma, J. Ba, Adam: A method for stochastic optimization, in: International Conference on Learning Representations, 2014.

[34] T. O'Malley, E. Bursztein, J. Long, F. Chollet, H. Jin, L. Invernizzi, et al., Keras Tuner, 2019, https://github.com/keras-team/keras-tuner.

[35] R. Garnett, Bayesian Optimization, Cambridge University Press, 2022.

[36] S. Peter, F. Diego, F.A. Hamprecht, B. Nadler, Cost efficient gradient boosting, Adv. Neural Inf. Process. Syst. 30 (2017) URL http://github.com/svenpeter42/LightGBM-CEGB.

[37] J. Cong, B. Xiao, Minimizing computation in convolutional neural networks, in: International Conference on Artificial Neural Networks, in: LNCS, vol. 8681, Springer, 2014, pp. 281–290.