sustainability

Article

Synthetic Dataset of Electroluminescence Images of Photovoltaic Cells by Deep Convolutional Generative Adversarial Networks

Héctor Felipe Mateo Romero [1,*], Luis Hernández-Callejo [2,*], Miguel Ángel González Rebollo [1], Valentín Cardeñoso-Payo [3], Victor Alonso Gómez [4], Hugo Jose Bello [5] and Ranganai Tawanda Moyo [6] and Jose Ignacio Morales Aragonés [4]

[1] Departamento Fisica de la Materia Condensada, Universidad de Valladolid, 47011 Valladolid, Spain
[2] Departamento Ingeniería Agrícola y Forestal, Universidad de Valladolid, 47002 Valladolid, Spain
[3] Departamento Informatica, Universidad de Valladolid, 47011 Valladolid, Spain
[4] Departamento de Física, Universidad de Valladolid, 47002 Valladolid, Spain
[5] Departamento de Matemática Aplicada, Universidad de Valladolid, 47011 Valladolid, Spain
[6] Department of Mechanical Engineering, Durban University of Technology, P.O. Box 1334, Durban 4000, South Africa
* Correspondence: hectorfelipe.mateo@uva.es (H.F.M.R.); luis.hernandez.callejo@uva.es (L.H.-C.); Tel.: +34-975129418 (L.H.-C.)

Abstract: Affordable and clean energy is one of the Sustainable Development Goals (SDG). SDG compliance and economic crises have boosted investment in solar energy as an important source of renewable generation. Nevertheless, the complex maintenance of solar plants is behind the increasing trend to use advanced artificial intelligence techniques, which critically depend on big amounts of data. In this work, a model based on Deep Convolutional Generative Adversarial Neural Networks (DCGANs) was trained in order to generate a synthetic dataset made of 10,000 electroluminescence images of photovoltaic cells, which extends a smaller dataset of experimentally acquired images. The energy output of the virtual cells associated with the synthetic dataset is predicted using a Random Forest regression model trained from real IV curves measured on real cells during the image acquisition process. The assessment of the resulting synthetic dataset gives an Inception Score of 2.3 and a Fréchet Inception Distance of 15.8 to the real original images, which ensures the excellent quality of the generated images. The final dataset can thus be later used to improve machine learning algorithms or to analyze patterns of solar cell defects.

Keywords: generative adversarial neural networks; photovoltaics; artificial intelligence; synthetic data; electroluminescence

check for updates

Citation: Romero, H.F.M.; Hernández-Callejo, L.; Rebollo, M.Á.G.; Cardeñoso-Payo, V.; Gómez V.A.; Bello, H.J.; Moyo, R.T.; Aragonés, J.I.M. Synthetic Dataset of Electroluminescence Images of Photovoltaic Cells by Deep Convolutional Generative Adversarial Networks. *Sustainability* **2023**, *15*, 7175. https://doi.org/10.3390/su15097175

Academic Editors: Jack Barkenbus and Firoz Alam

Received: 24 February 2023
Revised: 23 April 2023
Accepted: 24 April 2023
Published: 25 April 2023

1. Introduction

A number of factors (energy crisis, wars, climate change, etc.) are causing a rise in renewable energies use. Solar energy can be easily and affordably converted either into thermal energy by means of thermal panels or into electrical energy, using photovoltaic panels (PV) [1]. Industrial plants generating electricity from solar energy, commonly known as solar farms, are generally composed of a high number of photovoltaic (PV) panels made of PV cells. As the number of installations increases, the maintenance of solar farms becomes a nontrivial problem [2]. The energy produced depends on different conditions, such as the state of the panels, the climate, or the time of year. Solar panels are also vulnerable to phenomena that can reduce or nullify their performance. These issues make necessary a system to control and optimize production, since manual human labor is not enough as the number of panels gets higher.

Artificial intelligence is usually applied to solve difficult control and optimization problems. When applied to PV systems, different AI methods such as fuzzy logic, metaheuristics, or neural networks have been used to solve problems [3]. The most important problems include [4] max power point tracking, output power forecasting, parameter estimation, and defect detection.

Defect detection is one of the most interesting problems and researched topics in PV systems [5,6]. It is usually tackled with different kinds of neural networks, since they have great performance, although, as an important drawback, they need high amounts of data to perform better than other machine learning methods [7]. This could be a major impossibility for problems where it is difficult to harvest new data. Most of the approaches to detect the state of PV panels use electroluminescence images of the cells as an input, which is an invasive method that makes it difficult to carry out measurements and gather data.

Data augmentation is the most common method to deal with image data scarcity by means of the introduction of slight modifications (rotations, flips, and minor deformations) to the original images in order to create new images [8–10]. More recent papers promote the use of more complex AI techniques, such as Generative Adversarial Networks (GANs), to generate synthetic images [11,12]. GANs are state-of-the-art algorithms for data generation [13]. They have also been applied to PV systems for solving different problems [14].

The generation of synthetic EL images of PV cells using GANs has also been proposed in other works [11,12]. These works present synthetic datasets created with different GAN architectures trained with EL images of cells with different kinds of defects. Although the datasets in these works have not been made public, from their work, it can be deduced that the synthetic images are just labeled from visual inspection in order to train standard defect/normal classifiers, ignoring the output energy output performance of synthetic cells, since they cannot be measured.

In this paper, we present a new approach to deal with the image data scarcity problem. Starting from a small set of electroluminescence images of PV cells obtained under experimental conditions to be described, a synthetic dataset of images was created using Generative Adversarial Networks (GANs). For each synthetic image, a scalar value that represents the performance of their energy production is associated using machine learning techniques trained with the energy production of the original PV cells. It is also a continuation of the paper presented in [15], where we augmented and improved the dataset and applied new metrics to ensure the appropriate quality of the generated data.

To ensure reproducibility, the resulting dataset was made publicly available, so it can be used to improve the performance of AI models, to analyze the characteristics, properties, and defects of the cells, or to compare with other methods of generation of synthetic images.

This paper is structured as follows: Section 2 briefly reviews the Generative Adversarial Networks, Section 3 presents the methodology that was followed to generate the dataset, in Section 4, the synthetic dataset is described, and finally, the conclusions of the paper are presented in Section 5.

2. Generative Adversarial Networks

Generative Adversarial Networks (GANs) [16] are one of the most important and popular technologies nowadays and have been applied to different fields [17]. They can be applied to semisupervised and unsupervised learning. A GAN is usually defined as a pair of neural networks that are competing against each other. The network known as the Generator is the one designed with the job of trying realistic new data to try to deceive the other network, known as the Discriminator. This network has to decide if the data that it receives were forged.

The Generator does not have access to the real data, which is an important feature of these algorithms. The Generator has to learn how to create the data based on the feedback from the Discriminator. The Discriminator has access to both kinds of data, but it does not know which kind of data is an image before carrying out the prediction. The networks change their weights depending on the results of the deception; the Generator uses them to improve the forgeries, while the Discriminator tries to improve its recognition of forgeries. Figure 1 shows a diagram of the behavior of the algorithm.

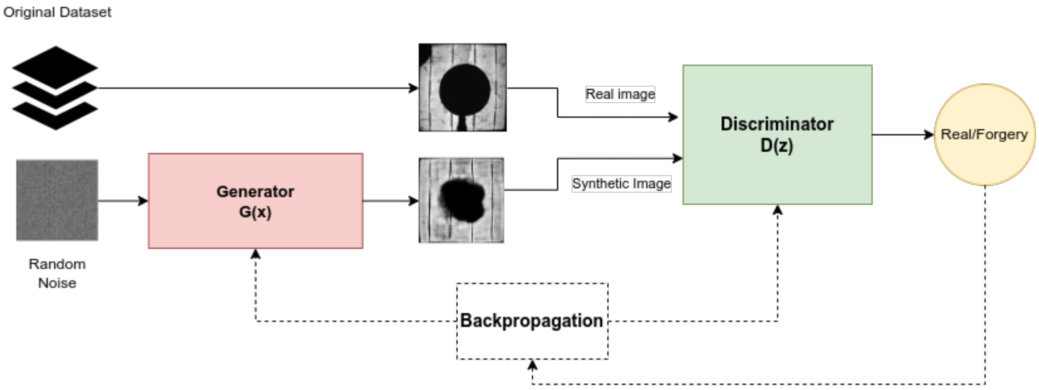


Figure 1. Diagram of a GAN.

The basic principle of operation of a GAN can be expressed as a two-player minimax game played between D and G, with a value function $V_{GAN}(G, D)$ given by the following mathematical expression [16], a binary cross-entropy function, commonly used in binary classification problems:

$$\min_G \max_D V_{GAN}(G, D) = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[\log D(x)] + \mathbb{E}_{\mathbf{z}\sim p_z(\mathbf{z})}[\log(1 - D(G(z)))]. \tag{1}$$

The first term in (1) represents the expected value of entropy given by the Discriminator over the real data and the second is the entropy given by the Generator over the fake data obtained.

GANs are usually composed of Deep Feed-forward Networks, but more complex architectures can be used in order to improve the generative capacities of the algorithm and the quality of the forged data. One of the most used architectures are Convolutional Neural Networks.

Deep Convolutional GAN

Research on GANs has led to new interesting architectures which substantially improve the performance of the networks and the quality of the forged data [18]. For this paper, we implemented the architecture known as Deep Convolutional GAN [19]. This architecture is based on convolutional layers, but it also provides a set of constraints in order to provide more stabilized training and better quality in the output. The most important guidelines are the following:

- Use of batch normalization in the Generator and the Discriminator.
- Removal of fully connected hidden layers in both networks.
- Usage of convolutional transposed layers instead of the stridden convolutional layers. This is only applied in the Generator network.

3. Methodology

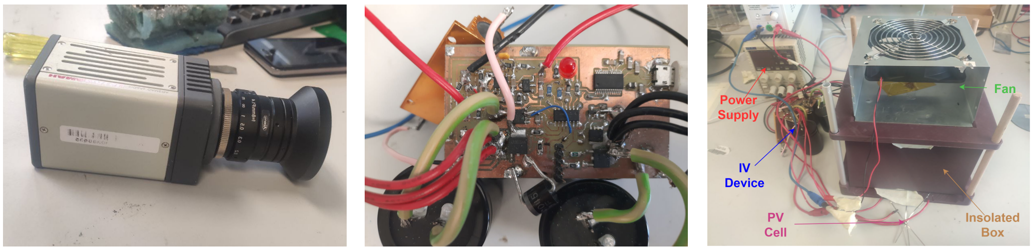
The correct preparation of the real and synthetic image datasets is a complex process which requires several steps to complete. In this section, we present the methodology that was followed in this research, which implies four relevant stages: manual acquisition of real EL images of PV cells beside their electrical characteristics (IV curves); data preprocessing to prepare for synthetic image generation using GAN; maximum power output assignment to synthetic cells from regression models trained with real images; and model and result validation.

3.1. Real Images Acquisition

Data availability is one of the most critical issues when using deep learning techniques. For precise PV cell characterization, we need two different kinds of data: the electroluminescence (EL) images of the photovoltaic cells and their corresponding IV curves. For the first, there are some public datasets available in the bibliography [20], but they do not include information about the IV curve, since it is not easy to measure the individual curve of a single PV cell. To solve this problem, we had to obtain the data using a specific technique developed previously [21] and following a manual process in the laboratory.

The capturing of the EL images was performed using a Hamamatsu InGaAs Camera C12741-03. The cell was isolated from external light in other to avoid the light produced by other kinds of radiations. Figure 2a presents the camera for taking the pictures. EL was chosen as the image capture technique, since it is widely used for detecting defects on PV modules [22] and it is the most used technique in other related works with PV systems. In order to test the different levels of luminous emission of the cell, various values of current were used to power the LED array when obtaining the IV curve and to the PV cell when capturing the EL image.

To measure the IV curve, we used a device specifically designed to measure the values of current and voltage of a single cell [21]. This device (Figure 2b) provided voltage and current values to build the IV curve of the cells. The IV curves can be used to calculate the max power point and the performance of the cells. Figure 2c shows the setting used for taking the IV curves. The original paper for this device validates the accuracy of the measurements.



(**a**) (**b**) (**c**)

Figure 2. Devices used to obtain the data: (**a**) InGaAs Camera; (**b**) IV curve tracer; and (**c**) setting to measure the IV curves.

The number of damaged cells was extremely limited in quantity and variety. The solution for this issue was to create artificial shadows in order to improve the amount and variety of images. The different shadows were created taking into account the most important defects found in solar farms. Figure 3 presents a representative image of each kind of artificial shadow that was used.

The final acquired dataset was composed of 602 different EL images with their corresponding IV curves. In order to allow the repeatability of the experiments, the dataset was made publicly available (The dataset can be downloaded from https://github.com/hectorfelipe98/Synthetic-PV-cell-dataset. The repository includes the original images, the synthetic dataset, and a CSV file that maps each image with its relative power and its class).

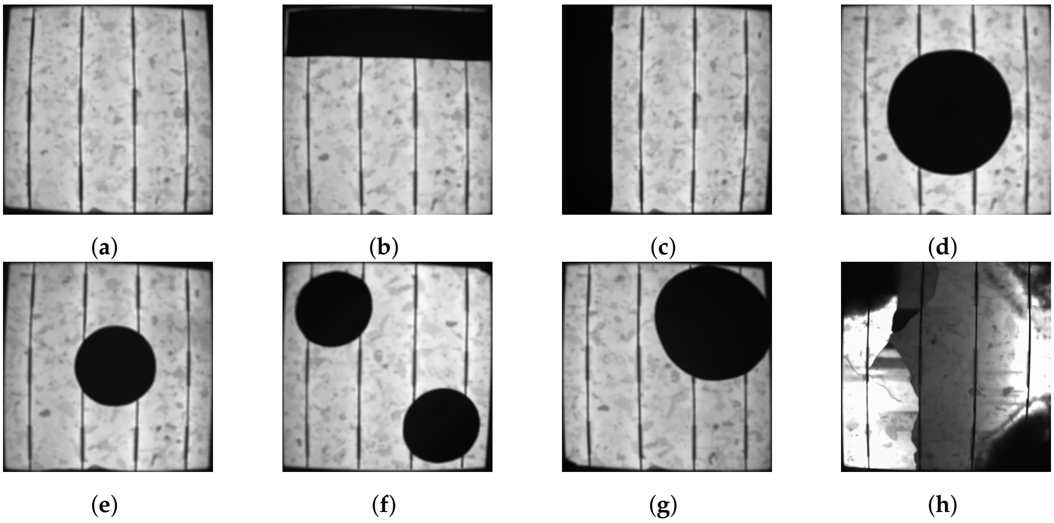
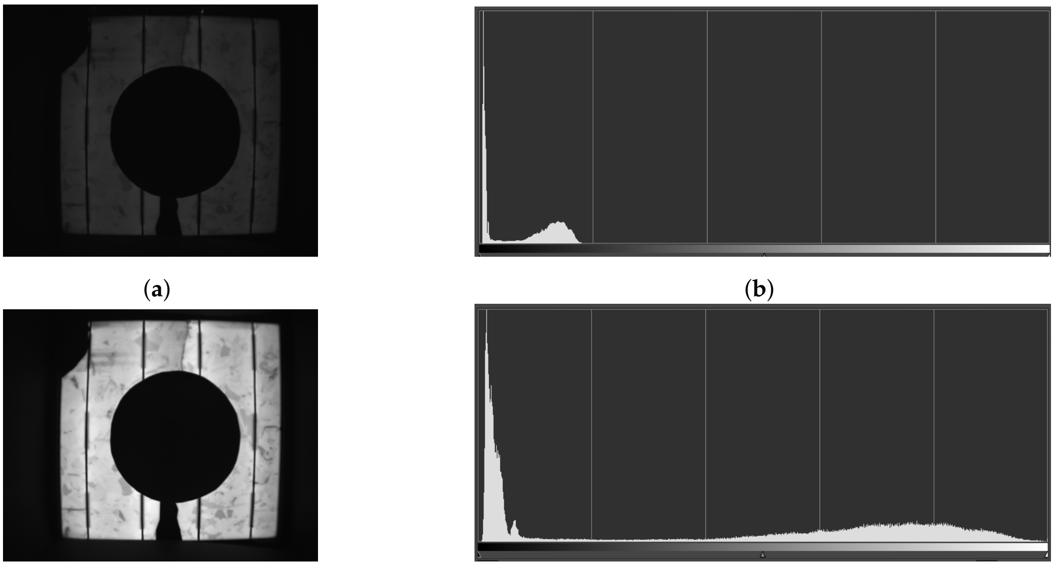


Figure 3. Shadows and defects used to create the dataset: (**a**) shadow/defect 0: original image; (**b**) shadow/defect 1: long horizontal line; (**c**) shadow/defect 2: long vertical line; (**d**) shadow/defect 3: big central circle; (**e**) shadow/defect 4: small central circle; (**f**) shadow/defect 5: two small circles in the corners; (**g**) shadow/defect 6: big circle in one corner; and (**h**) shadow/defect 7: other defects.

3.2. Image Preprocessing

The captured EL images were not suitable to be directly used with machine learning algorithms, since there were factors affecting image quality to be addressed first. For this reason, some algorithms were used to improve both the quality and normalization of the images. This section explains how the different problems found in image quality were solved.

- Dead pixels and luminous noise: The insulation of the system when capturing the images was not perfect. Some leaks could be found in some cases. This produced luminous noise when the image was taken. Another problem could be found in the camera. Due to its extended use, the camera had some dead pixels. Even though these phenomena are not easily visible to the human eye, they degenerate the quality of the obtained images. In order to solve these two problems, an image was taken before the real measure and without turning on the lights from the LED array. This provided information about the luminous noise and the dead pixel of the camera. Having this information, it is possible to perform a subtraction of the noise from the original image. This process was performed for each of the EL images.
- Images with poor lighting: The scale of the histograms of the images can produce problems in some libraries or programs which cannot recognize which values correspond with the white color. Figure 4a,b shows how the values are distributed in only a small fraction of the possible values, which makes the image look extremely dark. The solution to this problem corresponds to performing a min–max normalization, subtracting the minimum values of the image, and dividing it by the maximum value. The resulting image can be found in Figure 4c,d.
- Black surrounding contours: Figure 4c shows how images are surrounded by a black area. The reason for this phenomenon is caused by a limited focusing of the camera, which is capturing part of the walls of the insulated area. The solution to this issue consisted of performing a change in perspective. The first step consisted of applying different filters in order to delete the details, obtaining the maximum contour polygon of the cell. The next step consisted of performing a Hough transform to find the corners of the cells, using the polygon calculated previously as the base. In some cells, it was necessary to make some tweaks manually, as they presented strange defects that notably changed the aspect of the cell. The results of the transformation can be found in Figure 5.



(**a**) (**b**)

(**c**) (**d**)

Figure 4. Image before and after applying color normalization: (**a**) original image; (**b**) original histogram; (**c**) image after the min–max normalization; and (**d**) histogram after the normalization.

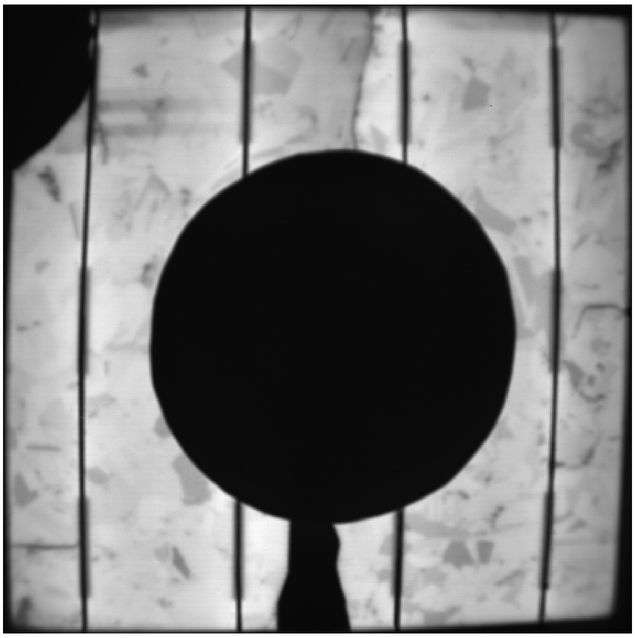


Figure 5. Cell after removing the surrounding image contour corresponding to wall portions.

3.3. Generation of the Synthetic Images

As announced, a Deep Convolutional Generative Adversarial Network was chosen to generate extra images from the original recorded set. The model is composed of two different networks: the Generator and the Discriminator. This section explains the architecture and parameters of both networks and the process of training.

3.3.1. Generative Network

The generative network was created following the principles of the DCGN. It has 3 different Convolutional Transpose Layers (Deconvolutional) in order to generate patterns. The use of these layers in combination with batch normalization [23] improves the generative capacities of the network and improves the stability of the training, The network uses Leaky Relu [24] as an activation function, since it usually outperforms the standard Relu. The architecture of this networks is represented in Figure 6. The input to the network

(Figure 7a) is a random noise array from a normal distribution. The output is a 200 × 200 image (Figure 7b. This size was chosen in order to reduce the computation cost of the algorithm while still obtaining images with a large amount of information. Other important hyperparameters can be found in Table 1.

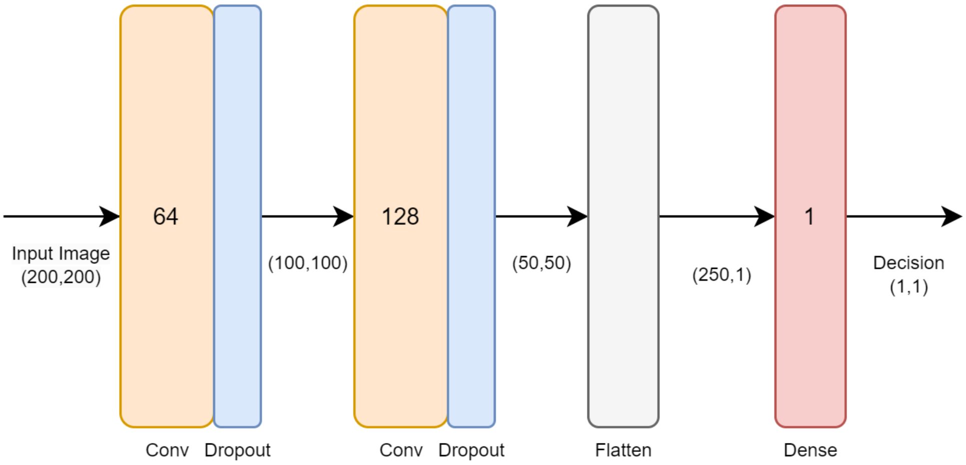
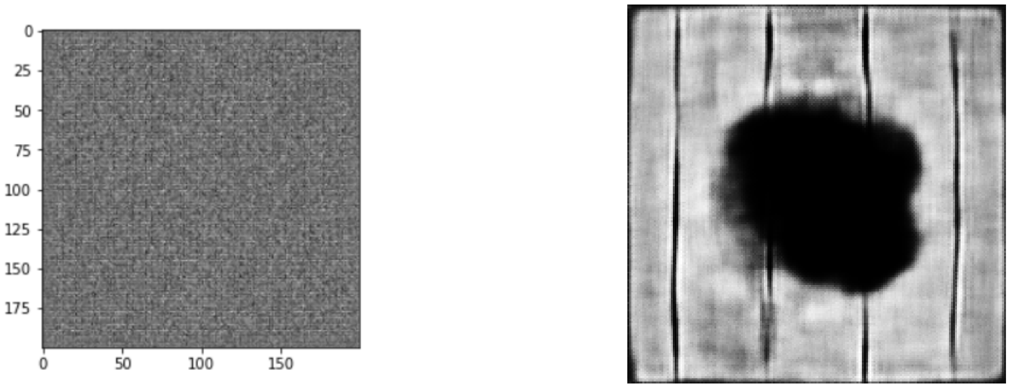


Figure 6. Architecture of the generative network.



(**a**) (**b**)

Figure 7. GAN's output images before and after training. (**a**) Output image before training. (**b**) Output image after training.

Table 1. Hyperparameters for both networks

Activation Function	Loss Function	Learning Rate	Epochs	Batch Size	Output Size
Leaky Relu	Cross Entropy	5×10^{-5}	800	4	200 × 200

3.3.2. Discriminator Network

The Discriminator network also followed the principles of DCGAN. The network uses different Convolutional Neural Networks to find patterns from the images. The typical feed-forward part of the network was also removed, with only the output layer remaining. The use of dropout layers and batch normalization improves the generalization capacity of the networks and stabilizes the training. The architecture is represented in Figure 8. The input corresponds to a 200 × 200 image and the output to a binary value that determines if the image is a real cell or a forgery. The other important hyperparameters were the same as the Generator, so they can be found in Table 1.

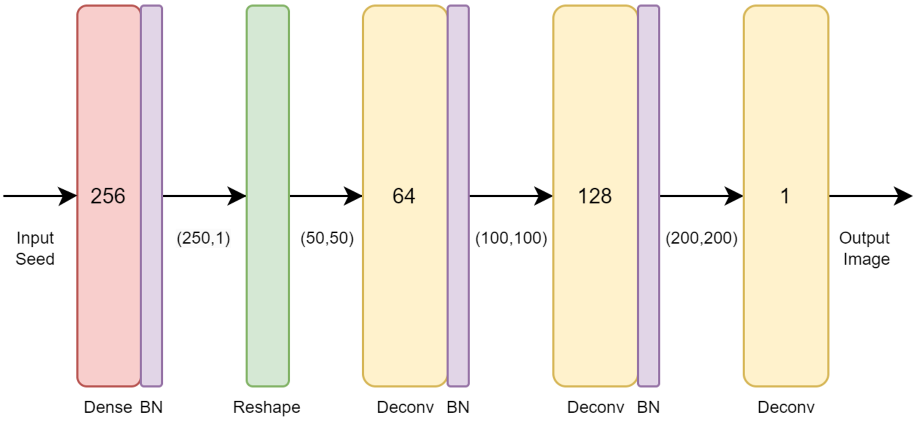


Figure 8. Architecture of the Discriminator network.

3.3.3. Training the GAN

The training was performed simultaneously in both networks with all the available samples (602). The training loop starts with the creation of synthetic images by the Generator using random seeds. After that, the Discriminator is provided with a mix of real and synthetic images. That input data are used to train the network, the loss is computed for each of the networks, based on the results of the Discriminator. Figure 9 shows the evolution of the loss for both networks. In the first epochs, the loss of the Discriminator network is quite high. It has not learned the patterns of the original images, so it cannot identify which images are real or forgered, even if the forged images are extremely similar to noise. After a while, the Discriminator reduces its loss since it can differentiate which images are real; this provokes an increase in the loss of the Generator. After that, the loss of the Generator steadily decreases its value since it starts to learn the patterns to create images similar to the original ones. Until epoch 400, it continues to learn when it obtains its lowest value. At the same time, the Discriminator is unable to identify which images are real, making its loss its maximum value. After epoch 400, the values have almost no change, so it can be concluded that the training should end at this point.

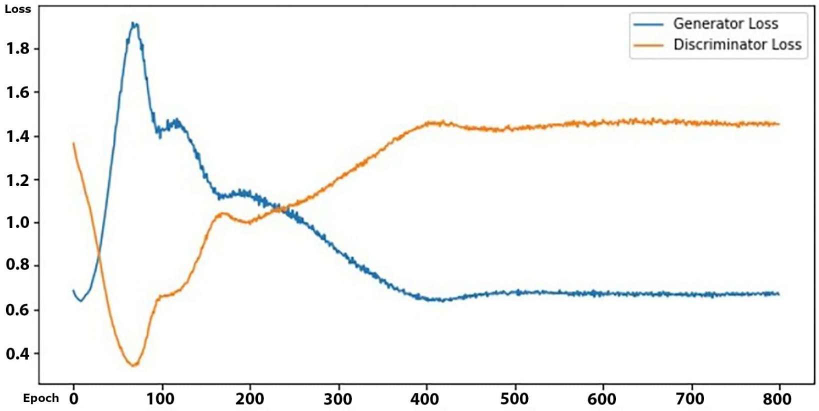


Figure 9. Evolution of the Generator and Discriminator loss.

The training was performed with a CPU AMD Ryzen 7 5800H, 16 GB of RAM, and a GPU Nvidia Geforce GTX 1650. It took 2 h and 41 min to complete the training.

After completing the training, the Generator network was used to generate the synthetic dataset. A total of 10,000 different images were created using 10,000 different random seeds. Figure 10 presents a selection of some generated images.

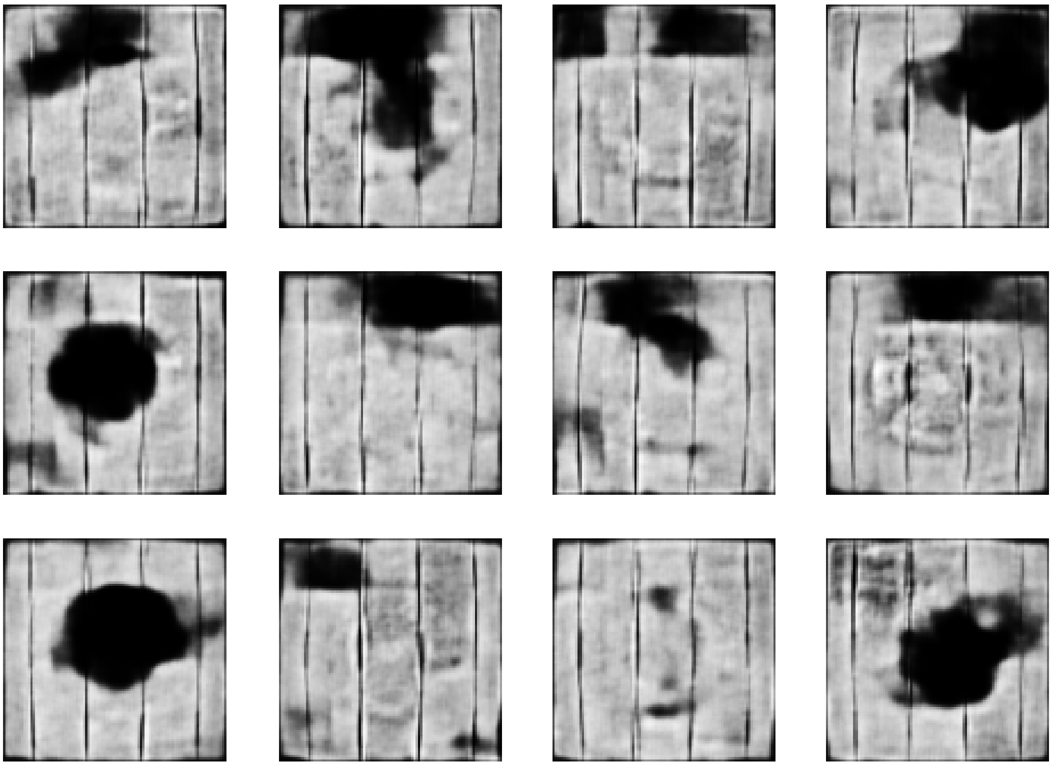


Figure 10. Samples of the synthetic images generated by our GAN-based method.

3.4. Assigning Maximum Power to Cells

The labeling of the images was not a trivial process, since it required associating each image (real or synthetic) with an output maximum power value, which represents the performance of the cell. This section explains how this association was made both for original and GAN-generated images.

3.4.1. Original Images

As explained before, each EL image has an associated IV curve. The IV curve provides information about the performance of the cell, which can only be obtained after taking the following steps:

- Compute the power–voltage curve of each cell from its IV curve.
- The maximum value of the curve is saved as the max power point (MPP).
- We gather the cells into 6 groups, in terms of the irradiance used to excite the cell before emission, since the MPP depends on the amount of light projected onto the cell.
- For each group, the 5 highest values of MPP are considered, and their mean is calculated.
- This average of the maximum values is understood as the expected MPP of a cell without any defects or shadows in that group.
- The MPP of each cell is divided by the averge MPP of the cell group it belongs to.

The resulting value will measure the relative performance of the cell, normalizing its MPP with the maximum value of the cells of the same group. This maximum is the mean of the 5 highest values, since it reduces the effects of potential incorrect measurement. Values of relative power near 1 will correspond with cells in good condition, with only a few or no defects. Low values correspond with underperforming cells, mostly due to their defects.

3.4.2. Synthetic Images

The labeling of the synthetic dataset was a completely different problem. As explained, the values of the original images were calculated based on their IV curve. This process cannot be replicated in the synthetic images, since they are not real cells; so, they cannot be measured.

To solve this problem, we formulated it as a regression problem that can be solved using a machine learning model. The model is trained with the full dataset of original images, together with their normalized power (MPP) calculated from IV curves, as explained in the previous subsection (602 samples). The chosen model is Random Forest [25], since it provided a low error in the original dataset and showed excellent generalization power to associate the MPP to the synthetic images. The implementation of the algorithm in the Sklearn library [26] was used for this work. The tuning of the hyperparameters of the RF model was carried out using the Grid Search method in the Sklearn library (GridSearchCV), obtaining the optimal values shown in Table 2.

Table 2. Estimation of Random Forest hyperparameters using GridSearchCV.

Parameter	Range	Optimum Value
n_estimators	$[20, 500]$	200
max_depth	$[0, 10]$	10
min_samples_split	$[1, 10]$	1
min_samples_leaf	$[1, 10]$	1
min_weight_fraction_leaf	$[0, 0.8]$	0

Since Random Forest is not suitable to work directly on raw images, some features were extracted from the images. The features are based on typical statistics (mean, standard deviation, etc.) and other characteristics directly extracted from the histogram (amounts peaks, peaks width, peaks height, amount of colors, etc.) A complete list can be found in Table 3. Feature selection (FS) is an important step in the preparation of machine learning models. We used correlation-based FS. As depicted in Figure 11, the cross-correlation between all the original sets of features shows that almost no feature is highly correlated with the others, except for the standard deviation and the variance, which are completely dependent on each other, meaning one of them can thus be safely removed from the final set of features.

The dataset was split into two sets: training (67%) and validation (33%). We decided to only use two due to our data limitation. The target variable was the relative power of each cell, standardized between 0 and 1.

Table 3. Features for Random Forest Regressor.

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

The model obtained a Mean Absolute Error (MAE) of 0.041 and a Mean Squared Error (MSE) of 0.0038 in the validation dataset. The distribution of the predictions of the model can be found in Figure 12. The low error and the similarity in the distribution confirm the validity of the model. The distribution of the prediction for the synthetic dataset can also be observed. Finally, the images were divided into two groups according to their predicted power (class 0 $>$ 0.8 and class 1 $<=$ 0.8). In total, 6963 images were classified as class 0 and 3037 as class 1.

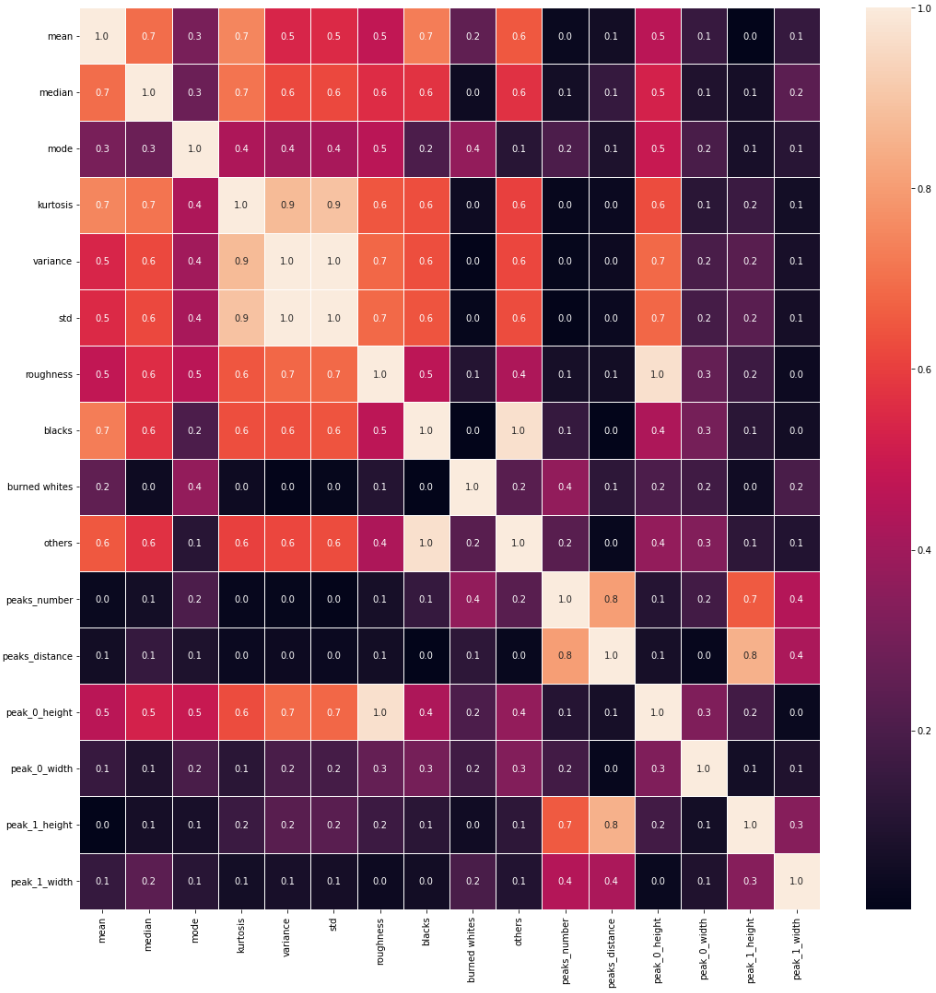


Figure 11. Correlation heatmap of the initial set of features.

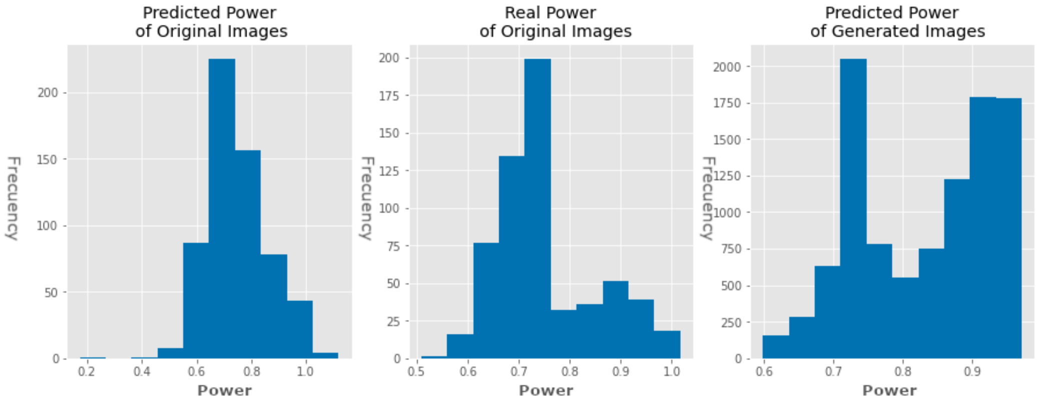
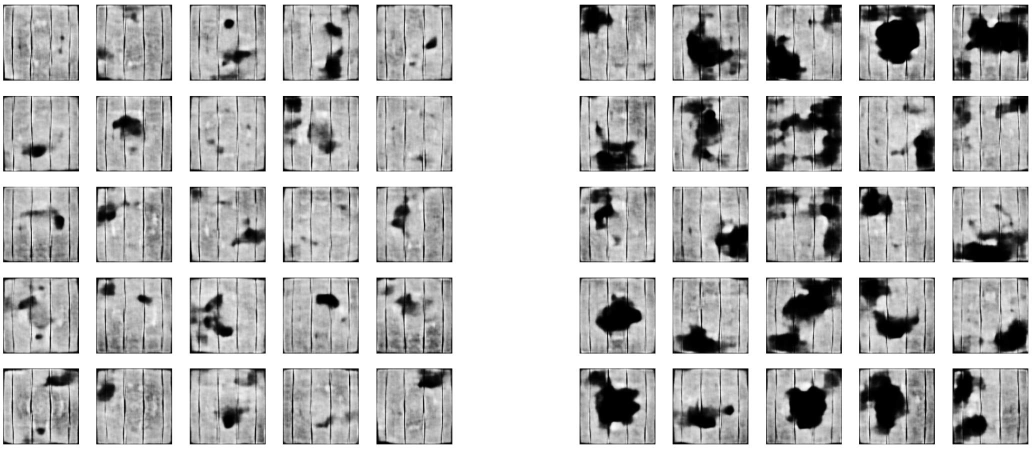


Figure 12. Histograms of real and predicted normalized power of the original and generated dataset.

4. Results

The resulting dataset was divided into two different folders, one for each class: Class 0 (6963 samples, Figure 13a) represented the images whose relative power is at least 0.8, and the images in that class can be considered as functional PV cells. Class 1 (3037 samples, Figure 13b) represented the images with a power of less than 0.8, and the images in that class can be considered as underperforming PV cells.



(**a**) (**b**)

Figure 13. Sample of images of both classes. (**a**) Sample of images of class 0. (**b**) Sample of images of class 1.

4.1. Visual Analysis

For ensuring the quality and similarity of the images, we propose two different methods: in this section, an analysis based on visual characteristics and histogram, and in the next section, an analysis based on different metrics.

As can be observed in Figure 13, the generated images present a similar structure while presenting new patterns of shadows different from the original ones (Figure 3). This is an interesting feature produced by the generative capacity of the GAN, since it can combine the different kinds of shadows presented in the original images in order to create new kinds of patterns. This improves the variety of shadows presented in the dataset.

Figure 14 presents the distribution of the features selected previously for labeling (Table 3). For each feature, the relationship between the values of the feature and the relative power of the cell is presented. Synthetic images are represented with orange dots, and the original images are represented with blue dots. In most features, the original dataset images appear as a subset of the synthetic dataset, with some exceptions caused mostly by underrepresented cases. This means that synthetic images not only present the characteristics of the original images, but they also present new cases of defects or shadows while maintaining the most important characteristics. This is mostly produced by the generative properties of the GAN, which can create new patterns combining the patterns of the input data; these new patterns improve the diversity of the dataset. This can lead to an improvement in performance in the machine learning methods that use this dataset. Another interesting finding is that the most underrepresented cases in the original data do not appear in the synthetic data; this is also provoked by the properties of the GAN, since it needs a considerable amount of samples to find patterns.

4.2. Histogram Analysis

The histogram of the images gives a lot of information about them. Figure 15a presents the mean histogram of all images of class 0 of the original dataset and the mean for all of the pictures of class 0 of the synthetic one. It can be seen that the images in this class present a large amount of light gray–white pixels (Values near 200), but they also can present some minor defects or shadows, as can be seen by the number of black pixels (values near 0). A difference between both datasets can be seen: the synthetic dataset images have higher but narrower peaks and are sometimes a bit moved to the left.

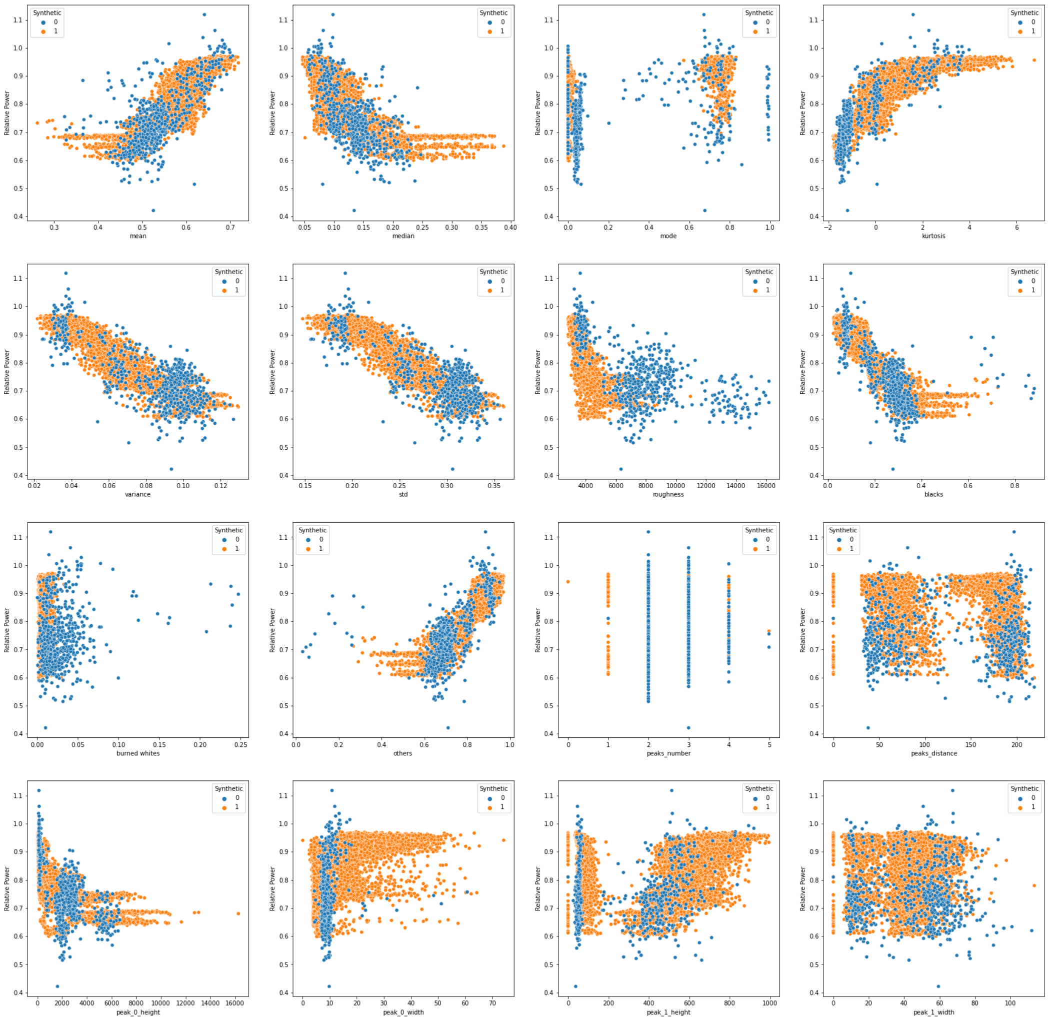
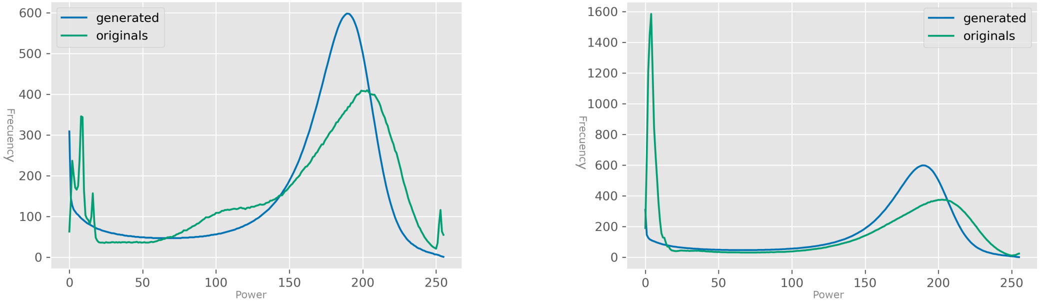


Figure 14. Distribution of the relative power generated by a cell as a function of the value of each of the sixteen features used to characterize the images: orange dots—synthetic images; blue dots—original images.

Figure 15a presents the same information for the images in class 1. The images in this class present a large number of dark pixels due to their defects and shadows. The amount of lighter pixels is considerably lower. In the synthetic images, the peak of black pixels is higher, but its width is narrower. The light pixels are extremely similar to the original ones.

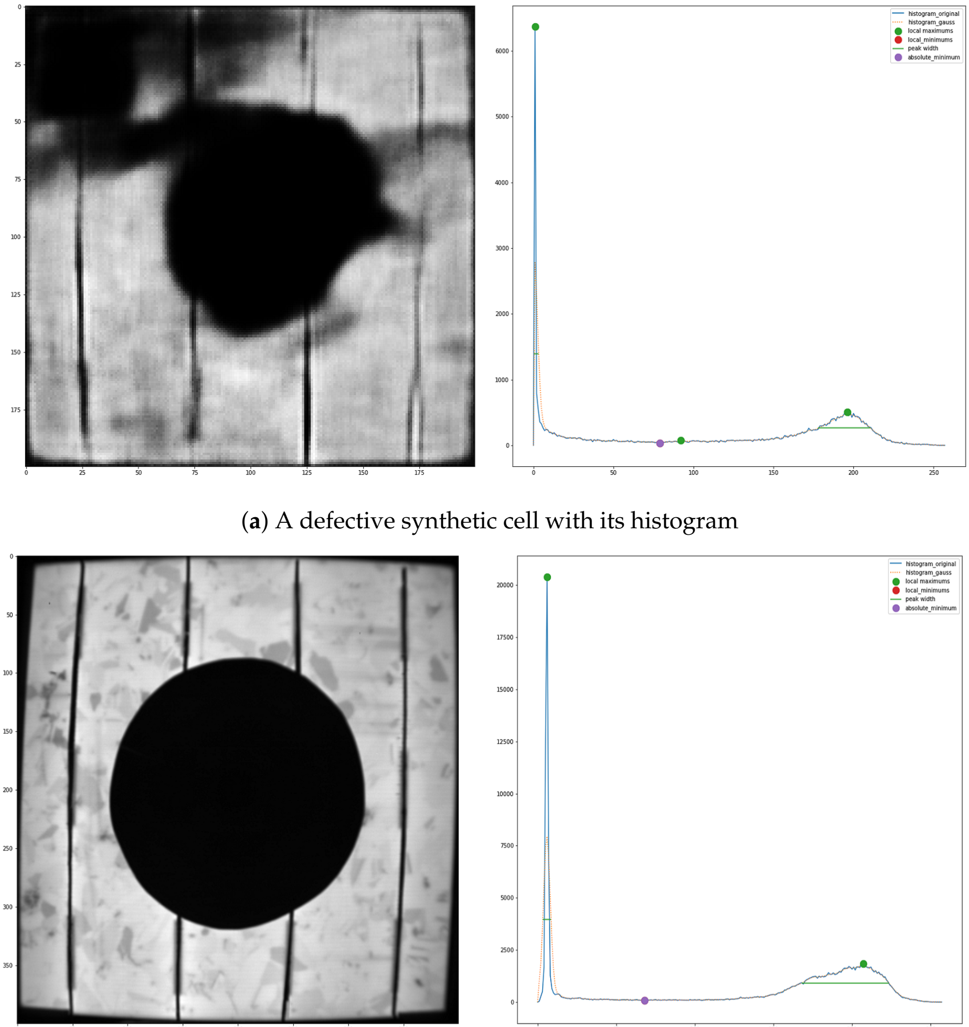


(**a**) Histogram of the images of class 0 (**b**) Histogram of the images of class 1

Figure 15. Different generated images of both classes.

As shown, the aspect of the histograms of both datasets is quite similar. The minor differences are mostly produced due to the augmented variety of patterns of defects and shadows.

Figure 16 presents two different cells of class 1: one original and one synthetic with similar aspects. Visual inspection of both histograms shows that they have a similar structure, presenting the same amount of peaks and even placed in a similar position. Nevertheless, synthetic images tend to show a more symmetrical histogram and a shift from maximum to lower intensities, since the extreme intensity values are less frequent than in real images.

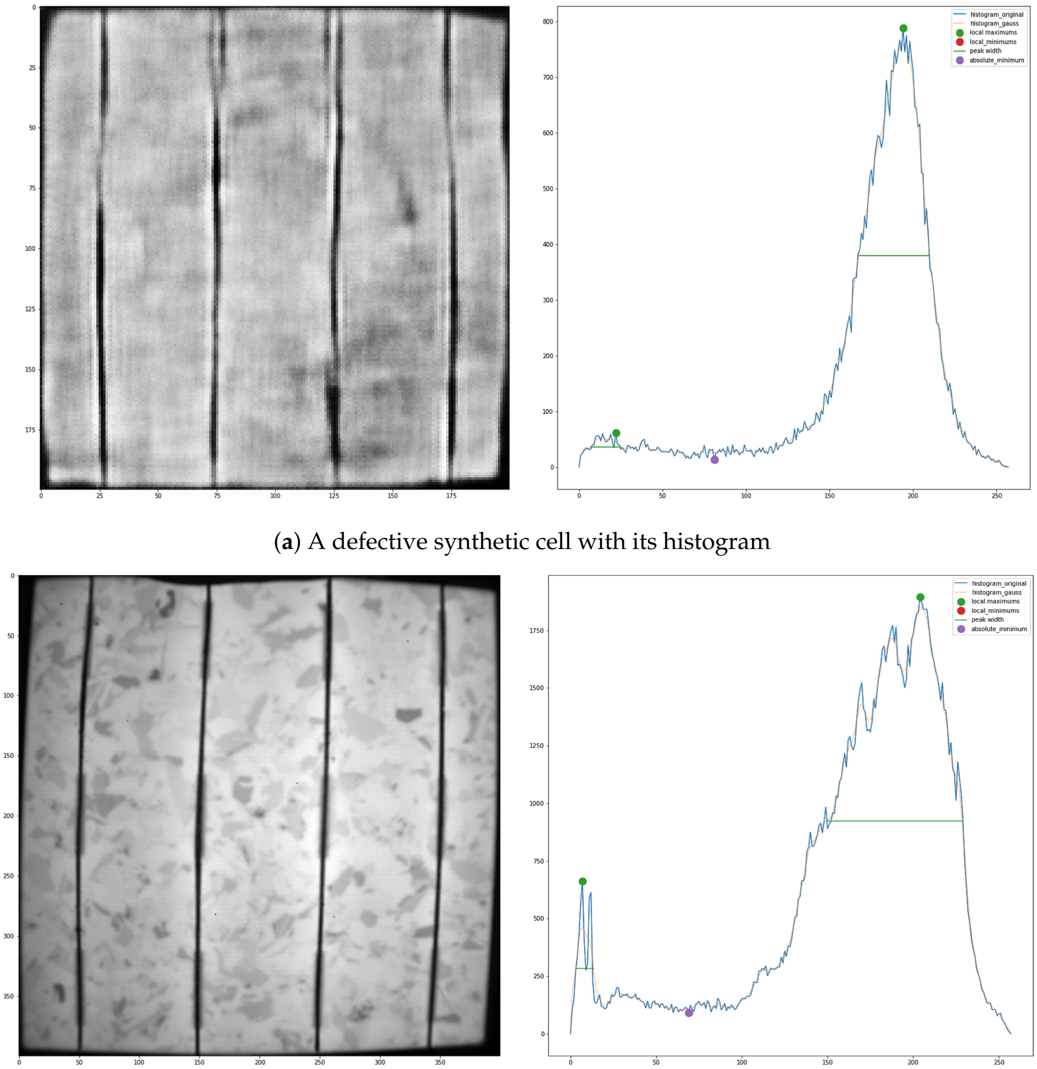


(**a**) A defective synthetic cell with its histogram

(**b**) A defective original cell with its histogram

Figure 16. Comparison of a defective synthetic cell and a defective original cell.

Figure 17 presents the same for two images of class 0. Both images have almost no apparent defects. Their histograms have similar shapes, as can be seen by the number of peaks and their placement.

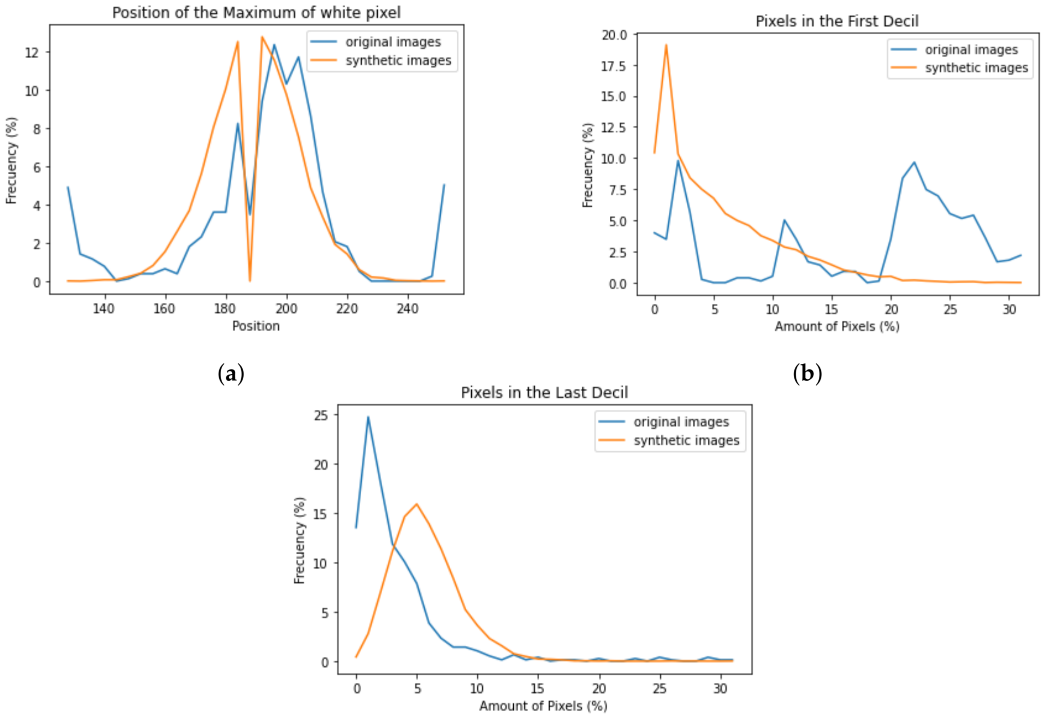


(**a**) A defective synthetic cell with its histogram

(**b**) A defective original cell with its histogram

Figure 17. Comparison of a defective synthetic cell and a defective original cell.

Figure 18a presents a comparison of the histograms of the position of the maximum, found in the right half of the histograms (gray/white colors); it can be seen that both histograms have a similar shape. Figure 18b presents the histograms of the number of pixels that have low values (up to 10% of the maximum values); it can be seen how the synthetic images do not completely imitate the original ones. A similar case can be seen in Figure 18c, which represents the histograms of the number of pixels in the last decile; it shows how even if the shapes are pretty similar, there is a shift to the left. It seems this GAN method has some flaws when finding the patterns around the most extreme values. This issue is not critical but shows that this method still has some room to improve.



(**a**) (**b**)

(**c**)

Figure 18. Comparison of the different aspects of the histograms of both original and synthetic images: (**a**) histogram of the position of the peak of gray/white colors for both original and synthetic images; (**b**) histogram of the number of dark pixels (first decile) for both original and synthetic images; and (**c**) histogram of the number of white pixels (last decile) for both original and synthetic images.

4.3. Image Quality Metrics

Previous works related to the synthetic generation of EL images of PV cells have not addressed the issue of ensuring the quality of their data by providing objective metrics. The Inception Score (IS) and the Fréchet Inception Distance (FID) are the most important metrics to ensure the quality of synthetic images. In the next paragraphs, both metrics are explained and used. A summary of the results of these metrics can be found in Table 4.

Table 4. Metrics for ensuring the quality of the synthetic dataset (Original: O, Synthetic: S, and Noise: N).

Metric	Mean Score	Std.
IS (O)	2.144	0.055
IS (S)	2.341	0.407
IS (N)	1.050	0.002
FID (O-O)	0.431	0.095
FID (S-S)	0.150	0.025
FID (N-N)	2.511	0.016
FID (O-S)	15.80	0.075
FID (O-N)	293.8	0.366
FID (S-N)	296.7	0.735

4.3.1. Inception Score

This metric was first proposed in 2016 [27] for evaluating the quality of generated artificial images. The score is computed based on the results of a pretrained InceptionV3 model [28] applied to the generated images. This score is maximized when two conditions are met: The value of the labels is the same for each image, in other words, the entropy of the distribution is minimized. The other condition is that the images are diverse, meaning that the labels are evenly distributed across all possible labels.

For our problem, we used a custom implementation of the metric based on Python and Tensorflow. We compared the results of three different datasets: the original dataset, the synthetic dataset, and a dataset composed only of noise. For each experiment, we split the evaluated dataset into 10 subsets and computed the IS of each one; after that, we computed the mean and the standard deviation. This process reduces the memory cost of the algorithm and the effects of randomness.

The original dataset obtained a mean score of 2.1440 with a standard deviation of 0.0559, the synthetic dataset obtained a mean score of 2.3418 with a standard deviation of 0.4079, and the noise dataset obtained a mean score of 1.0506 with a standard deviation of 0.0026. The results show that both datasets do not obtain a great result with this metric, but their results are better than the results of the noise. This is provoked mostly by the fact that Inceptionv3 was not trained to deal with EL images. It can also be observed that the results of both datasets are similar, with an error of 9.2%. The fact that [26] produced similar IS values for both shows that they have a high similarity. This proves the quality of the synthetic dataset but also shows that it has some room to improve.

4.3.2. Fréchet Inception Distance

This metric was first proposed in 2017 [29] to evaluate the quality of synthetic images. In contrast to the IS, this metric compares the distribution of the synthetic data with the distribution of the original data, measuring the similarity between the two datasets.

As in the other case, we used a custom implementation based on Python and Tensorflow. We compared the same three datasets as in the other metric. To compare each dataset with itself, we divided the datasets into two halves after shuffling them. We repeated the process five times in order to reduce the effects of the randomization. The results show that the values of the datasets with themselves were near 0 (0.43118 originals, 0.15095 synthetics, and 2.5117 noise).

We also compared the distance between the three datasets: 15.808 between originals and synthetics, 293.82 between originals and noise, and 296.77 between synthetics and noise. It can be seen that both originals and synthetics are considerably far from noise. It is true that the distance between them is not a value near 0, but it is still a low value, which shows that the difference between both datasets is not high. This is reinforced when this difference is compared with the difference between the noise and both datasets. The difference from 0 is mostly provoked due to the new patterns of shadows and defects that are generated due to the combinations of the different shadows of the original images, thanks to the generative capacity of the DCGAN models.

5. Conclusions and Future Work

The creation of synthetic electroluminescence images of photovoltaic cells is not a trivial problem. Different factors need to be taken into account in order to create high-quality images. The gathering of data is by its nature a manual process, requiring the EL image and its IV curve. The IV curve was measured individually for each cell, which is an innovation from other papers in the bibliography. The obtained images need proper preprocessing in order to perform well in the models. The labeling of the generated data is also a complex problem, since it is not possible to measure the output power or the IV curve of a not-real cell; so, we designed a model to assign a value of performance. This model was trained with the values of the real images, and its low error values show its good performance.

The algorithm used for creating the new images, DCGAN, has shown great performance, not only creating images that are very similar to the originals but also creating new patterns of defects and shadows. This similarity between the original images and the synthetics images was proved using different methods: A visual inspection of the cells shows that they present the characteristics of a real PV cell, and the histogram also proves their similarity, since they share the most important aspects of its shape. Image quality metrics such as Inspection Score (2.3) and French Inception Distance (15.8) also prove their similarity.

The results of this paper prove the quality of the synthetic images, but the dataset can be improved. The most direct way of improving the dataset would be increasing the amount of data. The inclusion of new kinds of defects and shadows would improve the generative capacities of the GAN. Another interesting option would be trying to use different kinds of PV cells, such as monocrystalline PV cells, since our dataset only consists of polycrystalline PV cells. This would improve the usefulness of the dataset in machine learning problems that use other kinds of PV cells.

Another way of improving the quality of the dataset would be increasing the size of the networks. The size of the current networks are based on the limitations in hardware and budget. Bigger networks could reduce the amount of epochs and improve the quality of the generated images. The use of the most innovative architectures could also improve the dataset.

Author Contributions: Conceptualization, H.F.M.R., L.H.-C. and V.C.-P.; methodology, H.F.M.R., V.C.-P. and M.Á.G.R.; validation, H.F.M.R. and V.A.G.; writing—original draft preparation, H.F.M.R. and H.J.B.; writing—review and editing, H.F.M.R., L.H.-C., V.C.-P., J.I.M.A. and R.T.M.; project administration, L.H.-C. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by the University of Valladolid with the predoctoral contracts of 2020, co-funded by Santander Bank.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The synthetic dataset can be found in https://github.com/hectorfelipe98/Synthetic-PV-cell-dataset.

Acknowledgments: This study was supported by the Universidad de Valladolid with ERASMUS+ KA-107. We also appreciate the help of other members of our departments.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

EL	Electroluminescence
PV	Photovoltaic
GAN	Generative Adversarial Network
DCGAN	Deep Convolutional Generative Adversarial Network
IV	Intensity–Voltage
LED	Light-Emitting Diode
MPP	Max Power Point
MAE	Mean Absolute Error
MSE	Mean Squared Error
IS	Inception Score
FID	Fréchet Inception Distance

References

1. Rathore, N.; Panwar, N.; Yettou, F.; Gama, A. A Comprehensive review on different types of solar photovoltaic cells and their applications. *Int. J. Ambient Energy* **2019**, *42*, 1–48. [CrossRef]
2. Hernández-Callejo, L.; Gallardo-Saavedra, S.; Alonso-Gómez, V. A review of photovoltaic systems: Design, operation and maintenance. *Sol. Energy* **2019**, *188*, 426–440. [CrossRef]

3. Mellit, A.; Kalogirou, S.A. Artificial intelligence techniques for photovoltaic applications: A review. *Prog. Energy Combust. Sci.* **2008**, *34*, 574–632. [CrossRef]
4. Mateo Romero, H.F.; Gonzalez Rebollo, M.A.; Cardenoso-Payo, V.; Alonso Gomez, V.; Redondo Plaza, A.; Moyo, R.T.; Hernandez-Callejo, L. Applications of Artificial Intelligence to Photovoltaic Systems: A Review. *Appl. Sci.* **2022**, *12*, 10056. [CrossRef]
5. Pillai, D.S.; Blaabjerg, F.; Rajasekar, N. A Comparative Evaluation of Advanced Fault Detection Approaches for PV Systems. *IEEE J. Photovoltaics* **2019**, *9*, 513–527. [CrossRef]
6. Hong, Y.Y.; Pula, R.A. Methods of photovoltaic fault detection and classification: A review. *Energy Rep.* **2022**, *8*, 5898–5929. [CrossRef]
7. Sarker, I.H. Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN Comput. Sci.* **2021**, *2*, 420. [CrossRef]
8. Demirci, M.Y.; Beşli, N.; Gümüşçü, A. Efficient deep feature extraction and classification for identifying defective photovoltaic module cells in Electroluminescence images. *Expert Syst. Appl.* **2021**, *175*, 114810. [CrossRef]
9. Su, B.; Chen, H.; Chen, P.; Bian, G.; Liu, K.; Liu, W. Deep Learning-Based Solar-Cell Manufacturing Defect Detection with Complementary Attention Network. *IEEE Trans. Ind. Inform.* **2021**, *17*, 4084–4095. [CrossRef]
10. Balzategui, J.; Eciolaza, L.; Arana-Arexolaleiba, N. Defect detection on Polycrystalline solar cells using Electroluminescence and Fully Convolutional Neural Networks. In Proceedings of the 2020 IEEE/SICE International Symposium on System Integration, SII 2020, Honolulu, HI, USA, 12–15 January 2020; pp. 949–953. [CrossRef]
11. Shou, C.; Hong, L.; Ding, W.; Shen, Q.; Zhou, W.; Jiang, Y.; Zhao, C. Defect Detection with Generative Adversarial Networks for Electroluminescence Images of Solar Cells. In *Proceedings of the 2020 35th Youth Academic Annual Conference of Chinese Association of Automation (YAC)*; IEEE: Piscataway, NJ, USA, 2020; pp. 312–317. [CrossRef]
12. Luo, Z.; Cheng, S.Y.; Zheng, Q.Y. GAN-Based Augmentation for Improving CNN Performance of Classification of Defective Photovoltaic Module Cells in Electroluminescence Images. *IOP Conf. Ser. Earth Environ. Sci.* **2019**, *354*, 012106. [CrossRef]
13. Creswell, A.; White, T.; Dumoulin, V.; Arulkumaran, K.; Sengupta, B.; Bharath, A.A. Generative Adversarial Networks: An Overview. *IEEE Signal Process. Mag.* **2017**, *35*, 53–65. [CrossRef]
14. Schreiber, J.; Jessulat, M.; Sick, B. Generative Adversarial Networks for Operational Scenario Planning of Renewable Energy Farms: A Study on Wind and Photovoltaic. In Proceedings of the Artificial Neural Networks and Machine Learning—ICANN 2019: Image Processing: 28th International Conference on Artificial Neural Networks, Munich, Germany, 17–19 September 2019; pp. 550–564. [CrossRef]
15. Mateo Romero, H.; González Rebollo, M.; Cardeñoso-Payo, V.; Gomez, V.A.; Moyo, R.; Hernandez Callejo, L.; Bello, H. Synthetic Dataset of Electroluminescence images of Photovoltaic cells by Deep Convolutional Generative Adversarial Networks. In Proceedings of the V Ibero-American Congress of Smart Cities, Cuernavaca, Mexico, 13–17 November 2022; pp. 855–869.
16. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. *arXiv* **2014**, arXiv.1406.2661.
17. Xue, J.H.; Titterington, D.M. Do unbalanced data have a negative effect on LDA? *Pattern Recognit.* **2008**, *41*, 1558–1571. [CrossRef]
18. Pan, Z.; Yu, W.; Yi, X.; Khan, A.; Yuan, F.; Zheng, Y. Recent Progress on Generative Adversarial Networks (GANs): A Survey. *IEEE Access* **2019**, *7*, 36322–36333. [CrossRef]
19. Radford, A.; Metz, L.; Chintala, S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arxiv* **2015**, arXiv:1511.06434.
20. Deitsch, S.; Christlein, V.; Berger, S.; Buerhop-Lutz, C.; Maier, A.; Gallwitz, F.; Riess, C. Automatic classification of defective photovoltaic module cells in electroluminescence images. *Sol. Energy* **2019**, *185*, 455–468. [CrossRef]
21. Morales-Aragonés, J.I.; Gómez, V.A.; Gallardo-Saavedra, S.; Redondo-Plaza, A.; Fernández-Martínez, D.; Hernández-Callejo, L. Low-Cost Three-Quadrant Single Solar Cell I-V Tracer. *Appl. Sci.* **2022**, *12*, 6623. [CrossRef]
22. Gallardo-Saavedra, S.; Hernández-Callejo, L.; Alonso-García, M.d.C.; Santos, J.D.; Morales-Aragonés, J.I.; Alonso-Gómez, V.; Moretón-Fernández, Á.; González-Rebollo, M.Á.; Martínez-Sacristán, O. Nondestructive characterization of solar PV cells defects by means of electroluminescence, infrared thermography, I–V curves and visual tests: Experimental study and comparison. *Energy* **2020**, *205*, 117930. [CrossRef]
23. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv* **2015**, arXiv:1502.03167.
24. Xu, B.; Wang, N.; Chen, T.; Li, M. Empirical Evaluation of Rectified Activations in Convolutional Network. *arxiv* **2015**, arXiv:1505.00853.
25. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
26. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
27. Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X.; Chen, X. Improved Techniques for Training GANs. In *Proceedings of the Advances in Neural Information Processing Systems*; Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2016; Volume 29.

28. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–June 30 2016; pp. 2818–2826. [CrossRef]
29. Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; Hochreiter, S. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. *Adv. Neural Inf. Process. Syst.* **2017**, *30*. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.