E-ELPV: Extended ELPV Dataset for Accurate Solar Cells Defect Classification

Marco Grisanti, Maria Ausilia Napoli Spatafora, Alessandro Ortis, and Sebastiano Battiato

Department of Mathematics and Computer Science, University of Catania, Catania, Italy

Abstract. Solar modules are subjected to every atmospheric event such as rain, wind, and snow and for this reason, they are usually built with protection frames. Nevertheless, these measures are insufficient to prevent some damages, especially for what regards the mechanical ones (e.g., the fall of tree branches) decreasing the power efficiency of solar modules. Then it is necessary to monitor their healthy conditions and replace or repair defective units. Electroluminescence, a useful inspection modality of solar modules, makes it possible to detect even the finest defects on the surface of solar modules. However, the analysis of these images is usually carried out by human operators, making this inspection practice expensive, time-consuming and it requires very specific knowledge. In state-of-the-art there are several works that distinguish between a healthy cell and defective cell, but a public dataset of possible defects in solar cells has never been published. For this reason, we propose a new dataset and a preliminary benchmark to make an automatic and accurate classification of defects in solar cells. The dataset includes five classes of defects and the pre-trained ResNext50 network reaches 0.07 Hamming Distance.

Keywords: Solar Energy, Solar Modules, Electroluminescence Imaging, Convolutional Neural Networks, Deep Learning, Visual Inspection, Defect Detection, Defect Classification

1 Introduction

Luminescence is a physical phenomenon that consists in the emission of photons of light by materials excited by causes other than the increase in temperature [1]. It arises from the property of some materials to absorb certain quantities of energy, which are subsequently returned in the form of light photons. Electroluminescence is a particular type of luminescence that characterizes some materials capable of emitting light under the action of an electric field, or rather when crossed by an electric current [2].

The electroluminescence (EL) test applied to photovoltaic panels is based on the reverse process of photovoltaics: a voltage is applied to the modules to check the current flows, while a camera with special sensors makes the infrared light emitted by the cells visible to the naked eye. Working cells will appear bright, while damaged ones will look dark [3]. The test is to be considered positive only for modules that demonstrate uniform current distribution. It is possible to run electroluminescence tests both in the laboratory and in the field. In the latter case, since the infrared radiation emitted by the sun is much higher than that of a solar cell, this type of test can only be performed at night. Generally, solar cell defects can be divided into two broad defect categories: intrinsic and extrinsic defects. Figure 1 shows an example of a cell extracted from an EL image of a photovoltaic module.

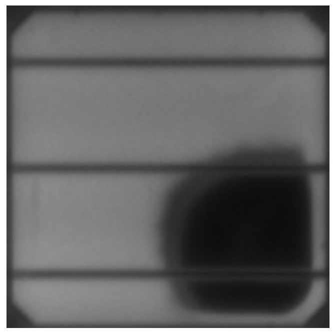


Fig. 1: The electroluminescence test applied to a photovoltaic panel cell. Note as the cell presents a Dark Area in the bottom-right part.

There is an increasing interest towards the deep detection of defects in several industrial products (e.g. Sarpietro et al. [4] developed a deep pipeline for classification of defect patterns applied in Silicon technology). This interest motivated us to propose a new dataset and its benchmark for the classification of defects in solar cells. The rest of the paper is structured as follows: the Section 3 describes the proposed dataset and method; in Section 4 our results are reported. Finally, there are conclusions and future works in Section 5.

2 Related Works

Starting from 44 EL images of photovoltaic (PV) modules, which consisted in 18 monocrystallyne modules and 26 polycrystalline modules, the work in [5] proposed a segmentation strategy in order to extract the various cells from the modules. By this process, the authors were able to extract 2624 cells. Subsequently, the authors of [6] dealt with the automatic classification of the various cells by a classifier reaching an average accuracy of 88.42% by considering a CNN based solution. This classification task aimed at indicating how likely a defect is present without specifying any details. The dataset (ELPV Dataset) used for the classification of the cells with the associated labeling has been publicly released. Using the same dataset, but with a little different labelling, the work in [7] implemented an isolated CNN, that is not pre-trained, for the classification of the cells, which achieves an average accuracy of 93.02%. The authors of [8] with a Deep Feature-Based Support Vector Machine (DFB-SVM) technique outperformed the works in [6] and [7] obtaining an average accuracy of 89.63% and 94.52%, respectively. The authors of [9] classified with an average accuracy of 83% two kind of defects: micro-cracks and finger-interruption. However, it used both the public ELPV Dataset and its own dataset, which makes the experiments not replicable.

The aim of this research is to develop a cell classifier able to classify the specific defect detected on the input image, if any, on the basis of a set of predetermined defects. Although previous research works in this field shown high quantitative performances, they only focused on the task of defect detection. This motivated us to extend the ELPV Dataset with a new labeling, concerning 5 specific defect classes and healthy classes (for the samples without defects). The labeling of an already existing large-scale dataset will be useful for the community, as ELPV represents a standard in the field. In this sense, the main contribution of the paper is the release of the new and extended labeling of the ELPV dataset, named E-ELPV (i.e., Extended-ELPV), and a benchmark evaluation for the task of defect classification. To our knowledge, no prior studies addressed the task of defects classification at this level of detail, nor similar public datasets have been released.

3 Methods

3.1 Dataset

The ELPV Dataset[1] [6] consists of 2624 EL cells of monocrystalline and polycrystalline photovoltaic modules labeled by an expert, to whom for each cell, in addition to answering the question "Is the cell defective?", also took care of answering the question "Are you sure?". By doing so, if the evaluator indicated with certainty that a cell was defective, a probability of defectiveness equal to 100% was assigned; if the evaluator indicated without certainty that a cell was defective, a probability of defectiveness equal to 67% was assigned; if the evaluator indicated with certainty that a cell was healthy, a probability of defectiveness equal to 0% was assigned; if the evaluator indicated without certainty that a cell was healthy, a probability of defectiveness equal to 33% was assigned. Table 1 resumes the original labelling strategy of the ELPV Dataset with the cardinality of each class. Figure 2 shows a monocrystalline cell and a polycrystalline cell, whereas Figure 3 shows a comparison between a cell with a probability of defectiveness equal to 0% and a cell with a probability of defectiveness equal to 100%

Cells that have been assigned a defect probability of 0% are marked as *Healthy* on our labeling. The remaining cells have been labeled as follows: *Crack*

[1] https://github.com/zae-bayern/elpv-dataset

Condition	Confident?	Probability of Defectiveness	Cardinality
Healthy	Yes	0	1508
Healthy	No	0.33	295
Defective	No	0.67	106
Defective	Yes	1	715
		Total	2624

Table 1: The original labelling strategy of the ELPV Dataset with the cardinality of each class.

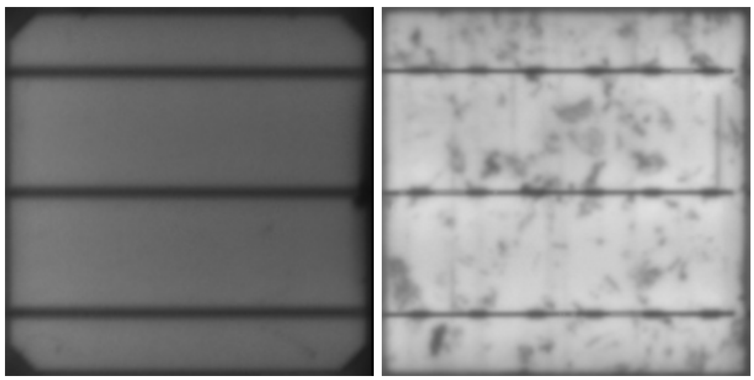


Fig. 2: On the left is shown a monocrystalline cell. On the right is shown polycrystalline cell. Both images have a probability of defectiveness of 0%. The monocrystalline cell appears cleaner than the polycrystalline cell, which appears to be dirty.

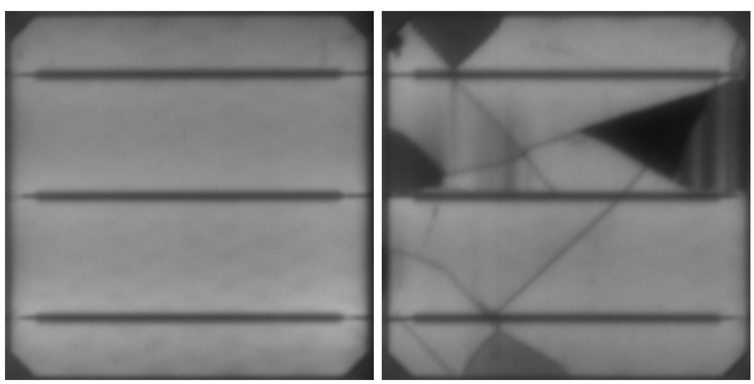
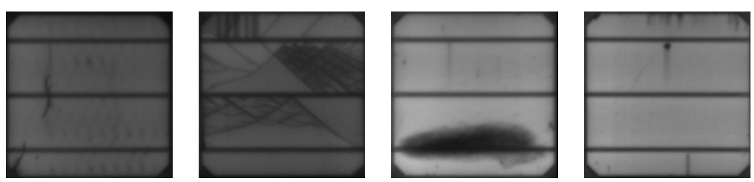


Fig. 3: On the left is shown a cell with a probability of defect equal to 0. On the right is shown a cell with a probability of defect equal to 1.

if the cell presents one or more cracks (not too much otherwise the cell is considered as totally broken); *CellBreakage* if the image presents cracks covering more than 50% of the cell's surface; *DarkArea* if the cell has one dark area; *HotSpot* if the cell has one or more hot spots; *OtherDefect* if the cell has a defect different from the previous ones. Then, the associations in Table 2 were obtained. The number of associations is greater than the number of images because to each defective cell could contain one or more defects. In particular, 2551 images are associated to one label and 73 images are associated to two labels. Figure 4 shows the four types of defects studied in this work.

Label	Associations
Healthy	1508
Crack	430
CellBreakage	133
DarkArea	62
HotSpot	171
OtherDefect	393
Total	2697

Table 2: Our custom labelling of the ELPV Dataset.



(a) (b) (c) (d)

Fig. 4: **(a)** cell with Crack defect; **(b)** cell with a CellBreakage defect; **(c)** cell with DarkArea defect; **(d)** cell with a HotSpot defect.

3.2 Proposed Method

Our methodology consists in the use of two classifiers: the first deals with classifying the health of a cell, indicating whether it is healthy or defective. In the latter case, the cell is given to the second classifier who is responsible for establishing which defects is present. Furthermore, we have developed an additional classifier that treats the healthy images together with the defective images, the latter labeled with the respective defects. Our purpose is to answer to the following questions: 1) Is training a Healthy VS Not-Healthy binary classifier better than training a multiclass classifier with the same architecture? 2) Does adding the Healthy class to the other classes in the same training improve the classification rate of the other classes?

All developed classifiers use the same image processing and image augmentation. They are developed with PyTorch library, using Python as Programming Language. Furthermore, they are tested and compared with the following networks: ResNext50[10], VGG-11[11], Inception-V3[12] and DenseNet-121[13]. These neural networks have different characteristics and depth.

Images are normalized using the following formula:

$$\overline{image} = \frac{image - mean}{std} \tag{1}$$

where $\overline{image}$ is the normalized image; *image* is the original image; *mean* and *std* are respectively the mean and the standard deviation of ImageNet [14] challenge database. This is due to the fine-tuning of a pretained model on this dataset. Data augmentation is used. Offline image augmentation simply consists of rotating each image 90°, 180° and 270°. Online image augmentation consists of the following transformations: Gaussian Blur (Kernel $5 * 5$); Color Jitter; Random Horizontal Flip ($p = 0.5$); Random Vertical Flip ($p = 0.5$); Random Rotation $(-3, +3)$; Random Translation $(0.02, 0.02)$.

3.3 First Classifier (Healthy Classifier)

Given an image of a cell, the first classifier has the purpose of indicating whether the cell is healthy or has defects, without indicating in the latter case which defects are present. The metric used to measure the performance was Overall Accuracy. This classifier was trained with the following hyperparameters: learning rate is 0.0001 and batch size is 32.

3.4 Second Classifier (Defects Classifier)

This task deals with a multilabelling problem. It means that given an image of a cell marked by the first classifier as unhealthy, the second classifier is intended to indicate which defects are present between Crack, CellBreakage, DarkArea, HotSpot and OtherDefect. For this reason we have chosen a properly metric for the multilabelling problem such as the Hamming Distance calculated as follows for each *batch*:

$$HD = 1 - \frac{|predictions \cap groundtruths|}{|predictions|} \tag{2}$$

The loss function used for this classifier is *BCELoss* for ResNext50 and *BCEWithLogitsLoss* for the other networks. For this classifier the training setting was learning rate 0.0001 and batch size 32.

3.5 Third Classifier

To assess the need of two different classifers, we also trained a third classifier that performs the classification including all the classes. Given an image of a cell, the third classifier is intended to indicate if the cell is Healthy or if there are defects between Crack, CellBreakage, DarkArea, HotSpot and OtherDefect. The metric and the loss function are the same of the second classifier. The setting employed for the third classifier was learning rate 0.001 and batch size 32.

4 Results

We have chosen the mentioned neural networks (ResNext50, VGG-11, Inception-V3 and Densenet-121) to benchamark our dataset. First we computed a baseline for each neural newtork and then we fine-tuned them. We have chosen this strategy to measure the advantage of training the network instead of using its pre-trained weights on ImageNet. The results of the first classifier, the second classifier and the third classifier are shown in Table 3, Table 4 and Table 5, respectively. The dataset described in 3.1 is clearly unbalanced but we did not balance it because balancement strategies cause overfitting. We can note that with the ResNext50 network, it is possible to pass from a Hamming Distance of 0.41 (Baseline) to 0.08 in only two epochs for the second classifier; moreover, the Baselines of the VGG11, InceptionV3 and DenseNet121 networks have quite low Hamming Distances of 0.08, 0.10 and 0.18 respectively, making the training of the various networks unable to improve performances. Instead, the third classifier reaches convincing Hamming Distances after a significant number of epochs. The best result is obtained with the ResNext50 network passing from a Hamming Distance of 0.50 (Baseline) to 0.07 only after 100 epochs. It is important to underline that the other networks also seem to work well, reaching a Hamming Distance of 0.08. For each class and for each classifier Table 6 shows information about True Positives, True Negatives, False Positives and False Negatives. Instead, Table 7, Table 8 and Table 9 illustrate example of matching and mismatching. Additional materials reporting the classification results for each involved class, the overall dataset (and related labeling) is available at the companion website [2]. A machine with an Nvidia Quadro RTX 6000 was used to carry out the various experiments.

[2] https://iplab.dmi.unict.it/EELPV/

Network	Best Epoch	Overall Accuracy
ResNext50 Baseline	/	0.41
ResNext50	**86**	**0.79**
VGG11 Baseline	/	0.55
VGG11	98	0.77
InceptionV3 Baseline	/	0.47
<u>InceptionV3</u>	<u>88</u>	<u>0.78</u>
DenseNet121 Baseline	/	0.38
DenseNet121	91	0.77

Table 3: First Classifier Results. The best and the running up results are highlighted in bold and underline, respectively.

Network	Best Epoch	Hamming Distance
ResNext50 Baseline	/	0.41
ResNext50	**2**	**0.08**
VGG11 Baseline	/	0.08
VGG11	**97**	**0.08**
InceptionV3 Baseline	/	0.10
InceptionV3	**16**	**0.08**
DenseNet121 Baseline	/	0.18
DenseNet121	**54**	**0.08**

Table 4: Second Classifier Results. The best and the running up results are highlighted in bold and underline, respectively.

Network	Best Epoch	Hamming Distance
ResNext50 Baseline	/	0.50
ResNext50	**100**	**0.07**
VGG11 Baseline	/	0.17
<u>VGG11</u>	<u>92</u>	<u>0.08</u>
InceptionV3 Baseline	/	0.17
<u>InceptionV3</u>	<u>99</u>	<u>0.08</u>
DenseNet121 Baseline	/	0.22
<u>DenseNet121</u>	<u>99</u>	<u>0.08</u>

Table 5: Third Classifier Results. The best and the running up results are highlighted in bold and underline, respectively.

Classifier	Class	TP	TN	FP	FN	Total
First Classifier	Healthy	258	156	45	65	524
Second Classifier	Crack	13	88	40	60	201
	Cell Breakage	2	122	58	19	201
	Dark Area	2	164	26	9	201
	Hot Spot	0	167	3	31	201
	Other Defect	1	110	13	77	201
Third Classifier	Healthy	297	165	36	26	524
	Crack	52	443	8	21	524
	Cell Breakage	21	499	4	0	524
	Dark Area	5	506	7	6	524
	Hot Spot	15	481	12	16	524
	Other Defect	55	389	57	23	524

Table 6: Information about True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) for each classifier and for each class.

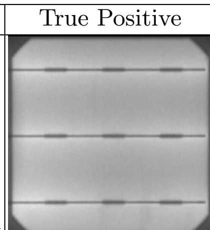
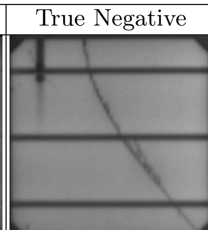
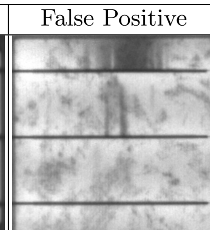
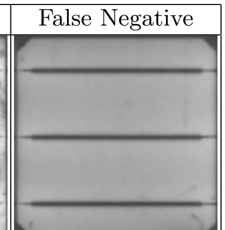
Class	True Positive	True Negative	False Positive	False Negative
Healthy				

Table 7: Some examples about True Positives, True Negatives, False Positives and False Negatives for each class of the First Classifier.

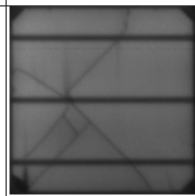
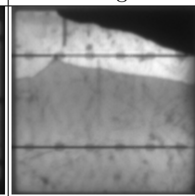
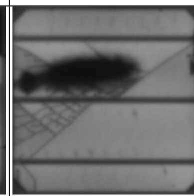
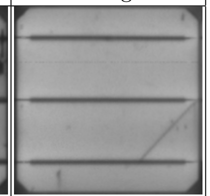
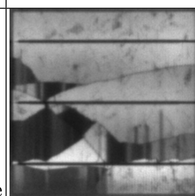
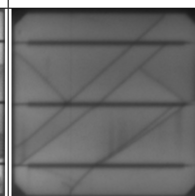
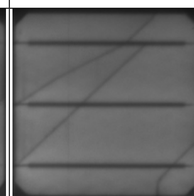
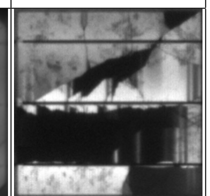
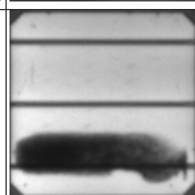
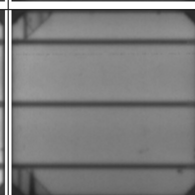
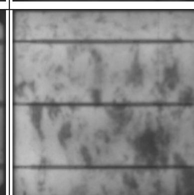
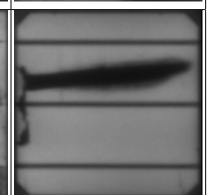
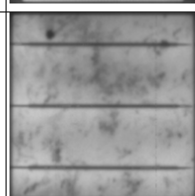
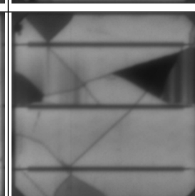
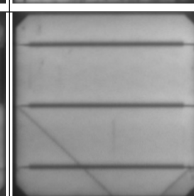
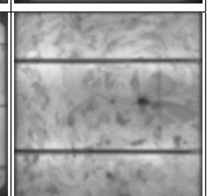
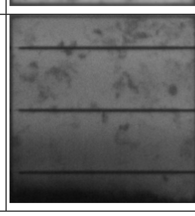
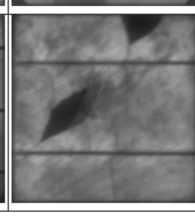
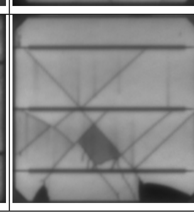
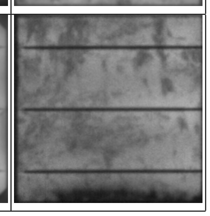
Class	True Positive	True Negative	False Positive	False Negative
Crack				
CellBreakage				
DarkArea				
HotSpot				
OtherDefect				

Table 8: Some examples about True Positives, True Negatives, False Positives and False Negatives for each class of the Second Classifier.

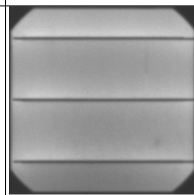
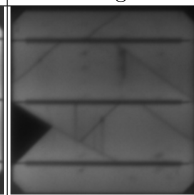
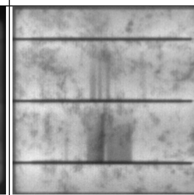
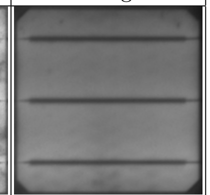
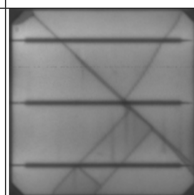
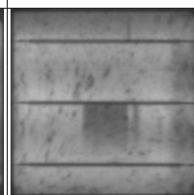
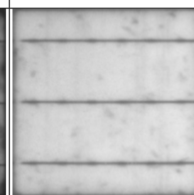
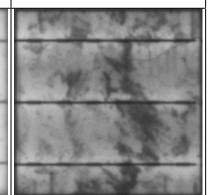
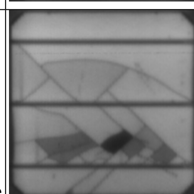
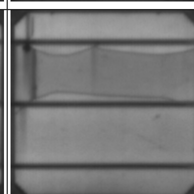
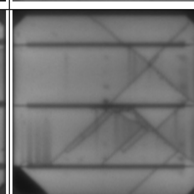
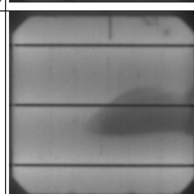
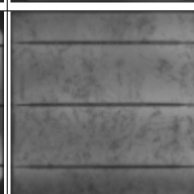
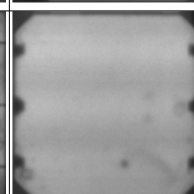
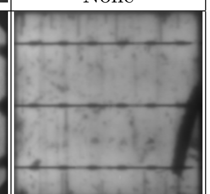
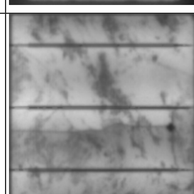
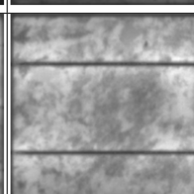
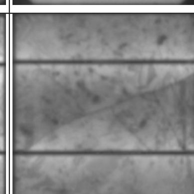
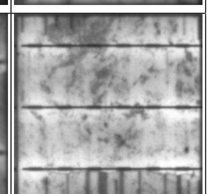
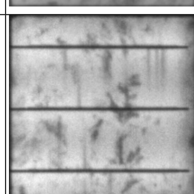
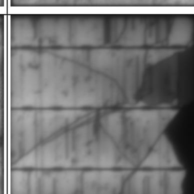
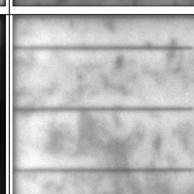
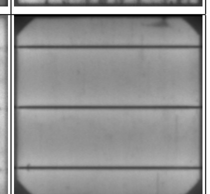
Class	True Positive	True Negative	False Positive	False Negative
Healthy				
Crack				
CellBreakage				None
DarkArea				
HotSpot				
OtherDefect				

Table 9: Some examples about True Positives, True Negatives, False Positives and False Negatives for each class of the Third Classifier.

5 Conclusions

Until now, the scientific community has concentrated its energies on finding techniques that will allow to automatically classify the health or the defect of a photovoltaic cell, without specifying the type of defect or, if it had been searched to classify the types of defects it would have been necessary to resort to private datasets, making the work not very transparent and replicable to researchers interested in this field of study. In fact the ELPV Dataset, that is the most famous public dataset of photovoltaic panel cells on which the most well-known works in the state-of-the-art are based, does not have specific labels regarding defects, indicating only a probability of defect. In our work we have extended this dataset by adding for each cell at least one type of defect among four common and well-known types, giving the scientific community the possibility of dedicating future energy to classify the various types of defects efficiently. Just to give a starting point, we performed benchmarks using pre-trained neural networks. Initially, we tried to use two classifiers, the first to distinguish healthy cells from defective cells and the second to identify various defects in the cells marked by the first classifier as defective. Subsequently, with the aim of making the discussion more complete and with more food for thought, we decided to implement another classifier that would automatically classify healthy cells and defective cells, indicating for the latter also the various defects present. By doing so, perhaps for the greater quantity of images taken together under examination, we have achieved concrete results, reaching a Hamming Distance of 0.07 with the RexNext50 network. Therefore, we can affirm: 1) Training a Healthy VS Not-Healthy binary classifier is better than training a multiclass classifier with the same architecture. 2) Adding the Healthy class to the other classes in the same training setting further improves the classification rate of the other classes. Due to the lack of a common dataset and benchmark in the field of the classification of defects in solar cell, we could not compare our results with oher works. Although this is a limitation for us, we have made a contribution to the scientific community with a new dataset and a benchmark on it that is a good starting point for future comparisons. As future works, we planned to apply on our task methods for defects detection in other industrial sectors such as Silicon technology.

References

1. Ilya Obodovskiy, *Radiation - Fundamentals, Applications, Risks, and Safety*, chapter Luminescence, Elsevier, 2019.
2. Huisheng Peng, Xuemei Sun, Wei Weng, and Xin Fang, *Polymer Materials for Energy and Electronic Applications*, Academic Press, 2016.
3. Sofiane Boulhidja, Sebastian Mellit Adel and, Voswinckel, Vanni Lughi, Alessandro Ciocia, Filippo Spertino, and Alessandro Massi Pavan, "Experimental Evidence of PID Effect on CIGS Photovoltaic Modules," *Energies*, vol. 13, pp. 537, 2020.
4. Riccardo Emanuele Sarpietro, Carmelo Pino, Salvatore Coffa, Angelo Messina, Simone Palazzo, Sebastiano Battiato, Concetto Spampinato, and Francesco Rundo, "Explainable Deep Learning System for Advanced Silicon and Silicon Carbide Electrical Wafer Defect Map Assessment," *IEEE Access*, vol. 10, pp. 99102–99128, 2022.
5. Sergiu Deitsch, Claudia Buerhop-Lutz, Evgenii Sovetkin, Ansgar Steland, Andreas Maier, Florian Gallwitz, and Christian Riess, "Segmentation of Photovoltaic Module Cells in Uncalibrated Electroluminescence Images," *Machine Vision and Applications*, vol. 32, pp. 1–23, 2021.
6. Sergiu Deitsch, Vincent Christlein, Stephan Berger, Claudia Buerhop-Lutz, Andreas Maier, Florian Gallwitz, and Christian Riess, "Automatic Classification of Defective Photovoltaic Module Cells in Electroluminescence Images," *Solar Energy*, vol. 185, pp. 455–468, 2019.
7. M. Waqar Akram, Guiqiang Li, Yi Jin, Xiao Chen, Changan Zhu, Xudong Zhao, Abdul Khaliq, M. Faheemm, and Ashfaq Ahmad, "CNN based automatic detection of photovoltaic cell defects in electroluminescence images," *Energy*, vol. 189, pp. 116319, 2019.
8. Mustafa Yusuf Demirci, Nurettin Beşli, and Abdülkadir Gümüşçü, "Efficient deep feature extraction and classification for identifying defective photovoltaic module cells in Electroluminescence images," *Expert Systems with Applications*, vol. 175, pp. 114810, 2021.
9. Wuqin Tang, Qiang Yang, Kuixiang Xionga, and Wenjun Yana, "Deep learning based automatic defect identification of photovoltaic module using electroluminescence images," *Solar Energy*, vol. 201, pp. 453–460, 2020.
10. Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.
11. Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
12. Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.
13. Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.
14. Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.