Machine Learning Prediction of Defect Types for Electroluminescence Images of Photovoltaic Panels

Claire Mantel[a], Frederik Villebro[a], Gisele Alves dos Reis Benatto[a], Harsh Rajesh Parikh[b], Stefan Wendlandt[c], Kabir Hossain[a], Peter Poulsen[a], Sergiu Spataru[b], Dezso Sera[b], and Søren Forchhammer[a]

[a]Departments of Photonics Engineering, Technical University of Denmark, DK-2800, Kongens Lyngby, Denmark
[b]Department of Energy Technology, Aalborg University, DK-9220, Aalborg, Denmark
[c]PI Photovoltaik-Institut Berlin AG, Wrangelstr. 100, D-10997 Berlin, Germany

ABSTRACT

Despite recent technological advances for Photovoltaic panels maintenance (Electroluminescence imaging, drone inspection), only few large-scale studies achieve identification of the precise category of defects or faults. In this work, Electroluminescence imaged modules are automatically split into cells using projections on the x and y axes to detect cell boundaries. Regions containing potential defects or faults are then detected using Hough transform combined with mathematical morphology. Care is taken to remove most of the bus bars or cell boundaries. Afterwards, 25 features are computed, focusing on both the geometry of the regions (e.g. area, perimeter, circularity) and the statistical characteristics of their pixel values (e.g. median, standard deviation, skewness). Finally, features are mapped to the ground truth labels with Support Vector Machine (RBF kernel) and Random Forest algorithms, coupled with undersampling and SMOTE oversampling, with a stratified 5-folds approach for cross validation. A dataset of 982 Electroluminescence images of installed multi-crystalline photovoltaic modules was acquired in outdoor conditions (evening) with a CMOS sensor. After automatic blur detection, 753 images or 47244 cells remain to evaluate faults. All images were evaluated by experts in PV fault detection that labelled: Finger failures, and three types of cracks based on their respective severity levels (A, B and C). Our results based on 6 data series, yield using Support Vector Machine an accuracy of 0.997 and a recall of 0.274. Improving the region detection process will most likely allow improving the performance.

Keywords: Machine learning, solar panel, defect detection, fault detection, electroluminescence imaging.

1. INTRODUCTION

The solar panel is the core component of the whole photovoltaic (PV) power plant.[1] Damage to the solar panel integrity such as defects and specially faults can significantly affect the power efficiency of the plant. Faults can occur for many reasons e.g., due to mistake in the installation and production process, thermal stress, and mechanical stress (e.g., hail, wind, snow, vibration due to transport). Additionally, minor and initially harmless faults can be gradually worsened with time, due to daily temperature cycles and weather conditions, and cause major power losses and/or safety hazards. Therefore it is desirable to monitor solar panels regularly and detect the faults to ensure maximum efficiency.

Solar panels can be inspected within three spectrums, i.e., Longwave Infrared (LWIR) for Infrared (IR) thermography, Visible Spectrum (VIS), and Shortwave Infrared (SWIR) for Electroluminance (EL). Not all PV panel faults can be detected by visual inspection; therefore thermographic and luminescence based imaging are employed to detect such faults. IR thermography has been implemented in the operations and maintenance market for automatic fault detection. However, generally, IR images are mainly suitable for hot spot detection.[1] The EL imaging is another technique[1–4] that has been widely used for inspection of the solar panels as it presents a material (semiconductor) level resolution. Therefore it is much more accurate to defect faults than IR
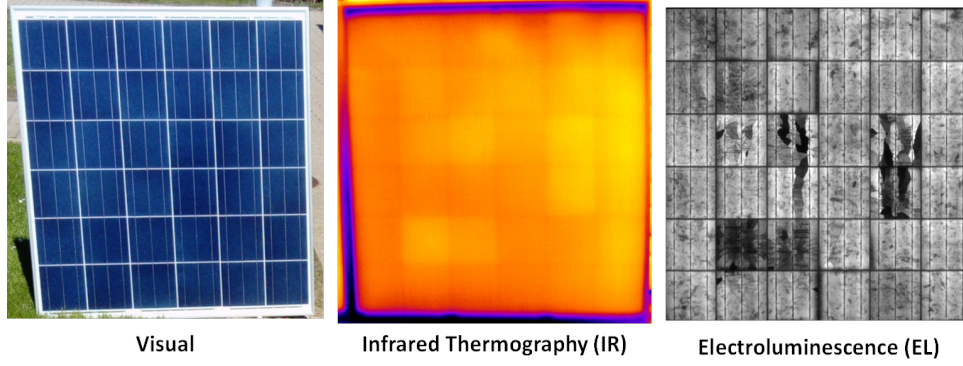
Figure 1. Visual, IR and EL images of the same PV panel. Crystallographic features, cracks and electrically inactive areas could only be detected by electroluminescence.
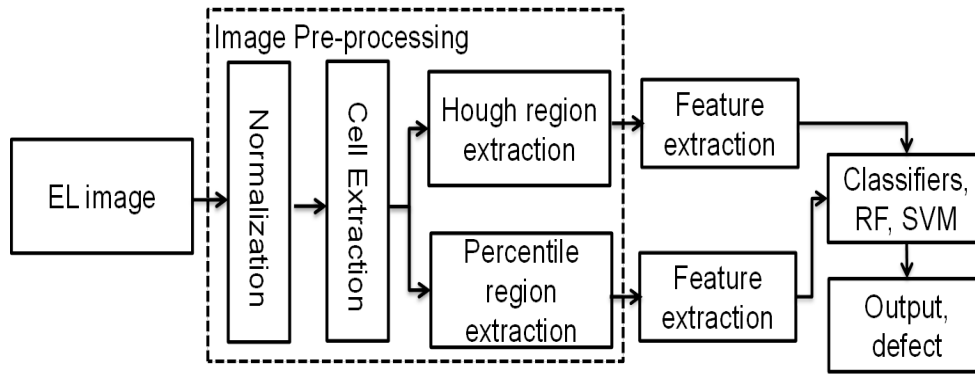

Figure 2. Overall diagram of the proposed model

thermography. In EL, a DC current is injected into a panel and the solar cells emit radiation the range of their band gap. Therefore, EL imaging provides more information concerning the health of PV modules, including mechanical and electronic properties on solar cell level. Such level of information that this technique provides can not be acquired by visual or IR inspection, as can be seen in Fig. 1.

Automatic analysis and fault detection holds the promise of providing fast, reliable and reproducible results. The analysis of the PV panels may be done overnight or even on the fly, and it will not be depending on a person assessing the imaged results. Machine learning can provide a frame work for training, validating, and developing an initial setup, as provided in this paper, over time into an automatic and robust solution.

The most recent work related to the automatic detection of PV faults in EL images is presented in these papers.[1,2] Mingjian Sun et al.,[1] present an automatic fault detection using deep learning-based Convolutional Neural Network (CNN) on EL images. They performed their investigation on small datasets, and therefore, data augmentation is applied to their datasets. Sergiu Deitsch et al.,[2] present automatic solar panel fault detection using CNN and machine learning classifiers, using Support Vector Machine (SVM). They applied their method on limited datasets and investigated a binary decision of detects, namely functional or defective.

Here we propose an automatic solar panel fault detection using two machine learning classifiers. We trained and evaluated our models to four types of faults, namely finger failures, and three types of cracks based on their severity level (A, B and C) as well as cells with no faults. In our evaluation, we investigate our model for two region extraction algorithms.

The remainder of this paper is organized as follows: The proposed method is detailed in Section 2. In Section 3, experimental results are presented: the performance of the proposed fault detection along with the result discussion. Finally, conclusions are provided in Section 4.
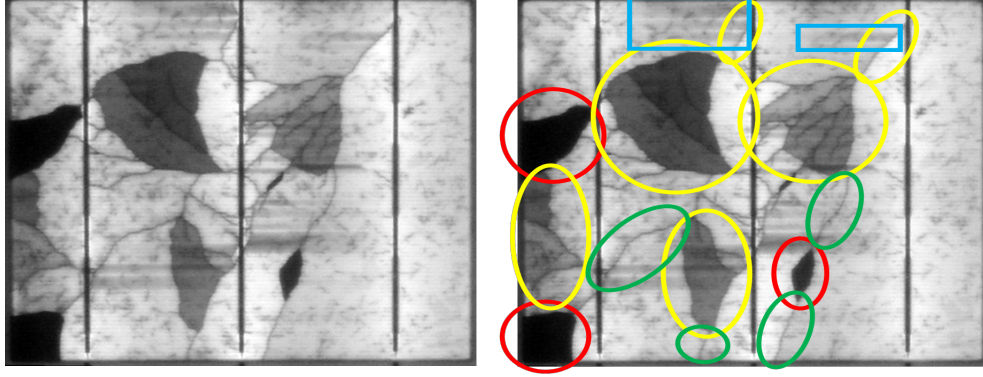
Figure 3. Example of the labelling of a damaged PV cell performed in this work. Left: Cell without labels for faults visualization; Right: The cracks are labelled as cracks A (green), B (yellow), C (red) and finger failures (blue) here as a consequence of cracks.

2. METHODOLOGY

We have applied two Machine Learning (ML) classifiers on EL images to detect and differentiate fault types of PV modules. The overall diagram of the proposed model can be seen in Fig. 2. In this process, at first,a normalisation of the input EL image is performed and then the cells are extracted. Afterwards, we have employed two region extraction algorithms (e.g., Hough and Percentile) to extract regions. Thereafter, 25 features have been extracted from each region detected. Independent of this, an expert evaluated each of the image sets before any region were detected and labelled them into five categories, i.e. No Failure and four types of faults (Finger failure, and three types of cracks based on their severity (A, B and C)). Finally, the extracted features and their corresponding labels are fed into the classifiers for the classification of each detected region into one of the five categories. In this Section, each of the steps is described briefly.

This section is organized as follows: first, the fault types are defined in Section 2.1, then the image pre-processing is presented in Section 2.2, ML classifiers along with feature extraction procedure are described in Section 2.3.2.

2.1 Fault Types

Four different types of faults were considered as class labels for the ML fault detection procedure. Those faults are composed of cracks with three different levels of severity and finger interruptions or dead fingers. The cracks are denominated A, B and C mode cracks, and here are defined slightly differently compared to the literature.[5, 6] Here we consider: i) Crack A as a crack which does not influence electrically its surroundings, so there is no perceptible EL intensity difference in the background around the crack; ii) Crack B as a crack which does influence the current flow in its surroundings, so there is a perceptible EL intensity difference around or at one side of the crack; iii) Crack C as a crack that completely isolates electrically a region of the cell, appearing as dark as the image background, or as defined in.[5, 6] For finger failures, we included both finger interruptions alone or associated with cracks.[5] The example of the faults labelled in a PV cell is shown in Fig. 3.

The differences to[5, 6] are due to the restriction in performing other EL measurements with the excitation current close to the short circuit current (the case for outdoor EL). The references point out that a low current (10% of the short circuit current) EL must be compared to define cracks A, B or C. Cracks A can present resistance when the current is lower. Cracks B show very low EL emission at low current, and if a crack does present considerable EL intensity at low current, it is a crack A; therefore, cracks B can be mistaken as A cracks. C cracks are easier to identify in short circuit current EL, as they should present no change at any current level. In the approach we adopted, we might be considering cracks B when they are actually cracks A, but they do present a certain level of crack resistance, differently of micro-cracks.

2.2 Image Pre-processing

The pre-processing steps, before any ML model is applied, can be split into 3 steps; a normalisation step, a cell extraction step and a region extraction step. Normalisation ensures consistency between images of different datasets. Cell extraction enables analysis on a cell level in contrast to panel level. Finally interesting regions of each cell in the panel is extracted. All of these steps will be explained in further detail in the following sections.

2.2.1 Normalisation

The normalisation step ensures that the image values are consistent between datasets. Panels within each data set shows signs of a gradient within the background, thus the normalisation step chosen rectifies this. Normalisation is carried out as described in.[2]

2.2.2 Cell Extraction

Cell extraction can be thought of as an extension of the algorithm in .[7] In order for the cell extraction to work it is assumed that all cell borders are parallel to the row and column plane of the image. This is satisfied for all datasets used in this paper. However, a further pre-processing as described in[7] can be added if this was not the case to rectify the perspective distortion due to angle of acquisition. The number of cells in rows and columns is another prerequisite for the algorithm to work.

Horizontal as well as vertical cell borders are found by summing the pixel values along the axis in question. Hereafter the auto correlation of the resulting vector is computed and the maximum found. This value is the approximate distance between cell borders and is used to further refine the coordinates of individual cell borders.

A Dirac comb with the appropriate period corresponding to approximate distance between cells is generated and used to refine the coordinates. This is done by looking at the distance between the Dirac peaks and the peaks of the summed axis in question. Once cell coordinates are found individual cells are extracted for further analysis.

2.2.3 Region extraction

The aim of the developed region extraction algorithms is to identify potential regions within a cell containing a fault. The region extraction pipeline can be seen in Fig. 4. Faults such as cracks or finger failures, which are investigated in this paper, are known to form lines or being darker than their surroundings. Hence 2 different approaches to the region detection have been developed. One specifically looking for lines within the cell utilising the Hough line transform, while the second approach identifies darker areas within each cell. However, before any region extraction, the bus bars are located and removed. Location of the bus bars are found in a similar manner as finding cell borders by summing the cell image in the appropriate dimension and finding the peaks. Removal is carried out by looking at the neighbouring pixel values and adjusting the pixel values of the bus bar.

The first approach, that we will subsequently call the Hough approach computes the gradients within the cell (blurred by a gaussian filter) by using a Sobel filter. A threshold in the form of the 80% Percentile of the gradients is used to turn the gradient image into a binary image. Everything above the found threshold is set to True and False otherwise. This image is then fed to the Hough line transform and a maximum of the 30 first elements with highest energy are used. Lines extracted are checked to be of a certain length and horizontal/ vertical lines close to the borders of the panel are removed. These lines are then dilated using a square 12x12 structuring element. This concludes the first region extraction algorithm as the dilated lines form the regions to be investigated.

The second approach, that we will subsequently call the Percentile approach is more simplistic and easier to implement. The goal is to find regions that are darker than their surroundings. In order to do so the 15% Percentile of the average pixel value of the cell image is found and used as a threshold. Once again a binary image is formed, but this time any pixel below this threshold is set to True and otherwise False. The resulting regions are dilated using the same 12x12 square structuring element. Thus, the second approach is completed and the regions found in the form of the dilated regions.
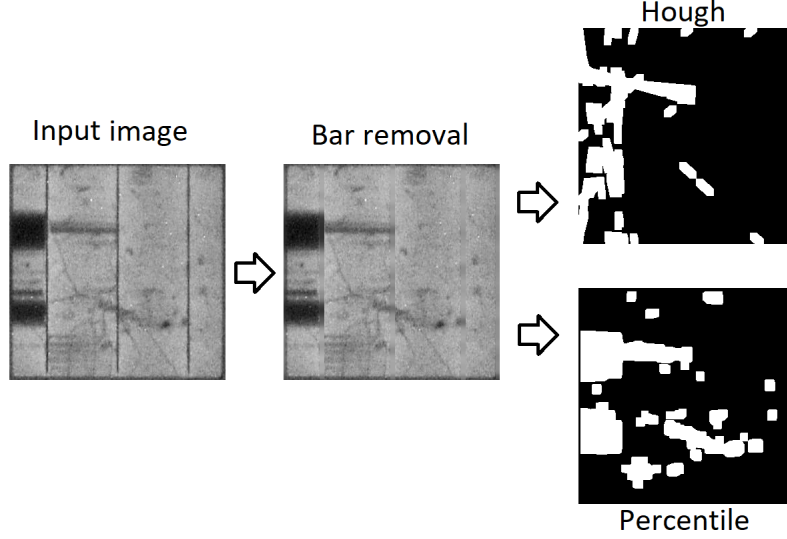
Figure 4. Visualization of the region extraction pipeline.

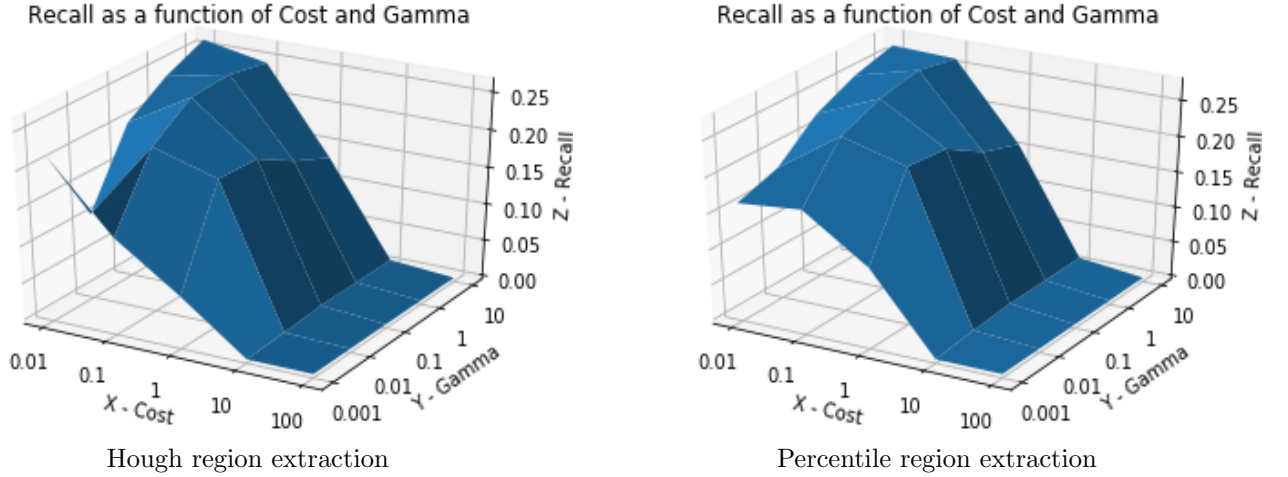Hough region extraction   Percentile region extraction

Figure 5. Parameters plots for recall for SVM

2.3 Machine Learning Classifiers

We have applied two ML classifiers (Support Vector Machine (SVM) and Random Forest (RF)) to evaluate the faults. The general approach to classify using ML classifiers are as follows: 1) feature extraction from each region and 2) train the classifiers using the estimated features.

In this section, the feature extraction process, including the description of ML models, is described.

2.3.1 Feature Extraction

The feature extraction technique is essential for training with machine learning classifiers, e.g., SVM and RF. We have extracted a set of 25 features focusing on the geometry of region and then applied feature engineering on the extracted features (e.g. median, standard deviation). The list of features used in our implementation can be seen in Table 1.

Table 1. Overview of the features calculated for each region

Features	Description
Mean	Mean of the region
Median	Median of the region
Mode	Mode of the region
Range	Minimum pixel value subtracted from the maximum value
Standard deviation	Standard deviation of the region
First Percentile	The 10% percentile of the region
Last Percentile	The 90% percentile of the region
Skewness	Calculated skewness of the values within the region
Kurtosis	Calculated kurtosis of the values within the region
Area	Area of the region
Perimeter	Perimeter of the region
Ellipsis Major Axis	Size of the longest side in the ellipsis around the region
Ellipsis Minor Axis	Size of the shortest side in the ellipsis around the region
Eccentriticty	Computed eccentricity of the region
Compactness	Defined as $\frac{Perimeter^2}{Area}$
Circularity	Defined as the $\frac{Area}{Area\ of\ circle\ with\ same\ perimeter}$
Rectangularity	Defined as $\frac{Area}{Area\ of\ bounding\ box}$
Jaccard	The Jaccard score of the region bounding box and a box the same size
Neighbours	The number of regions within a certain distance to region in question
Boundary Mean	Mean of the boundary in a certain radius arund the region
Boundary Std	Standard deviation of the boundary in a certain radius arund the region
Elongatedness	Defined as $\frac{Area}{4*Number\ of\ erosions\ for\ region\ to\ disappear^2}$
Centroid Distance	Euclidean distance between the centroid and the weighted centroid
Concentricity Max	Distance from centroid to maximum value
Concentricity Min	Distance from centroid to minimum value

Table 2. Overview of the amount of images and cells contained within each dataset

Series	Images	Rows	Cols	Cells	Total Cells
1	160	10	6	60	9600
2	102	12	6	72	7344
3	120	10	6	60	7200
4	140	10	6	60	8400
5	70	12	6	72	5040
6	161	10	6	60	9660
Total	753				47244

2.3.2 Support Vector Machine and Random Forest

We have applied two non-linear classifiers, i.e. SVM (RBF kernel)[8] and Random Forest (RF)[9] in our work. For SVM, the Radial Basis Function (RBF) is used as a kernel function in this paper. For RF, the number of estimators was 200.

3. EXPERIMENTAL RESULTS

3.1 EL Datasets

The EL datasets were obtained from different solar farms installed in UK. The PV panels were from different manufacturers, but all have the same type (156 mm multi-crystalline silicon) of PV cells. From six series of images, two had PV panels with 72 cells arranged in a 12 x 6 cells matrix and the other four had 60 cells

Table 3. Overview of the amount of faults within each dataset

Series	Crack A	Crack B	Crack C	Finger Failure	Total cracks
1	246	221	9	613	1089
2	7	8	12	65	92
3	1	12	13	0	26
4	4	11	18	164	197
5	8	18	51	154	231
6	33	94	70	15	212
Total	299	364	173	1011	1847

Table 4. Number of regions extracted using the 2 different algorithms.

Series	Hough	Percentile
1	147588	212680
2	112680	151876
3	75072	168227
4	104228	181687
5	73970	107311
6	155663	223671
Total	669201	1045452

arranged into a 10 x 6 cells matrix. The EL images of installed PV panels were acquired with a NIKON D750 modified camera, which has a CMOS sensor. The modification consisted of removal of the near-infrared filter and the addition of daylight filter during measurements, allowing it to acquire the luminescence of silicon-based solar cells. All the images were obtained during the evening in outdoor conditions, with a tripod holding the camera as perpendicular as possible to the PV module.

Each dataset consists of a series of images containing PV panels with either 60 or 72 cells in each panel. A total of 753 panels (one panel in each image) is investigated accomodating 47244 cells (Table 2). Further details about each series of images can be found in Table 2.

A total of 1847 faults was annotated and fitted into one out of 4 categories (Crack A, Crack B, Crack C and Finger Failure). A detailed overview of the faults in each dataset can be found in Table 3.

The Hough approach resulted in a total of 669,201 potential regions containing faults, while the Percentile approach resulted in 1,045,452 regions. Details about the number of regions in each dataset can be found in Table 4.

Based on our experiments, the results on the four types of faults are reported by accuracy, recall, and precision in this Section.

3.2 Performance Analysis and Discussion

Table 5. Results for different types of fault (SVM) using Hough

Fault types	Crack A	Crack B	Crack C	Finger Failure	No Failure
Crack A	39	51	1	29	519
Crack B	29	109	8	52	515
Crack C	2	8	29	37	91
Finger Failure	16	25	9	230	323
No Failure	11679	11454	805	14585	628556

Due to the massive imbalance in the datasets, we have applied 10x SMOTE[10] oversampling on all regions containing a fault followed by random undersampling to have a balanced dataset. In our experiment, all metrics were computed as the average of 10 repetitions using 5-fold stratified cross-validation.

Table 6. Results for different types of fault (RF) using Hough

Fault types	Crack A	Crack B	Crack C	Finger Failure	No Failure
Crack A	58	64	4	47	466
Crack B	60	116	18	54	465
Crack C	8	26	43	28	62
Finger Failure	53	56	28	176	290
No Failure	29461	26193	3858	18729	588838

It could be noted, for SVM, we found the parameters by grid search based on SVR results (for each of the metrics). As an example, Fig. 5 shows the recall as a function of gamma and cost parameters. For RF, we used the default parameters from the implementation by scikit-learn.[11] The detailed results are shown in Table 5 and 6, where the rows represent the true, and the columns represent the predicted values.

For SVM, in the case of the Hough region extraction method, the accuracy, precision, and recall are 0.997, 0.022, and 0.262, respectively. For Percentile, accuracy, precision, and recall are 0.997, 0.028 and 0.274, respectively. As an example, the corresponding optimal cost and gamma parameters are 10.00 and 0.001, respectively for accuracy (for both Hough and Percentile), 0.01 and 10, respectively for recall in the case of Hough, and 0.10 and 1.00, respectively in the case of Percentile. For RF, we got an accuracy, precision, and recall of 0.967, 0.0251 and 0.193, respectively for the Hough method. For Percentile approach, the accuracy, precision, and recall are 0.965, 0.038, and 0.258, respectively.

If we compare to the region extraction method (e.g., Hough and Percentile), in the case of SVM, the Percentile method gave little improvement in terms of precision and recall, even though accuracies are the same for both methods. For RF, the percentile method gave improvement in terms of accuracy, precision, and recall. As per our analysis, we found that the region extraction algorithm has little influence on the performance. If we compared the results of SVM and RF, both of these classifiers gave quite similar results. However, the SVM gave marginally improved results in terms of accuracy. Comparing Tables 5 and 6, we see that RF has a bias of declaring faults, thus capturing more faults, but also having more false positives.

4. CONCLUSION

We presented a method for solar cells fault detection in EL images using SVM and RF. In our experiment, a set of predefined features (mean, standard deviation and circularity etc) are used. Then SVM (RBF kernel) and RF are applied to detect the faults in the solar cells. Two region extraction algorithms have been employed in this work. Expert evaluated the images, categorized and labelled them into four types of faults (finger failure, and three types of cracks based on their severity).

As per the experiment, the best results were achieved by SVM, which gave an accuracy of 0.997 and a recall of 0.274. Our RF model gave 0.967 and a recall of 0.193. Thus, the SVM gave around 3% improvement compared to the RF in terms of accuracy. However, the high accuracy makes both our classifiers (SVM and RF) promising for detecting PV panel faults from EL images.

The accuracy is high, but the datasets are very imbalanced. For further work, the search can be carried out on the region extraction algorithms, since they influence the overall performance of fault detection of PV panel. Furthermore, different ML methods, including a deep learning-based structure, can be tried out.

ACKNOWLEDGMENTS

REFERENCES

[1] Sun, M., Lv, S., Zhao, X., Li, R., Zhang, W., and Zhang, X., "Defect detection of photovoltaic modules based on convolutional neural network," in [*Machine Learning and Intelligent Communications*], Gu, X., Liu, G., and Li, B., eds., 122–132, Springer International Publishing, Cham (2018).

[2] Deitsch, S., Christlein, V., Berger, S., Buerhop-Lutz, C., Maier, A. K., Gallwitz, F., and Riess, C., "Automatic classification of defective photovoltaic module cells in electroluminescence images," *CoRR* **abs/1807.02894 (on-revision)** (2018).

[3] Fuyuki, T., Kondo, H., Yamazaki, T., Takahashi, Y., and Uraoka, Y., "Photographic surveying of minority carrier diffusion length in polycrystalline silicon solar cells by electroluminescence," *Applied Physics Letters - APPL PHYS LETT* **86** (06 2005).

[4] Fuyuki, T. and Kitiyanan, A., "Photographic diagnosis of crystalline silicon solar cells utilizing electroluminescence," *Applied Physics A* **96**, 189–196 (Jul 2009).

[5] Köntges, M., Kurtz, S., Packard, C., Jahn, U., Berger, K., Kato, K., Friesen, T., Liu, H., Van Iseghem, M., Wohlgemuth, j., Miller, D., Kempe, M., Hacke, P., Reil, F., Bogdanski, N., Herrmann, W., Buerhop, C., Razongles, G., and Friesen, G., "Review of failures of photovoltaic modules," (01 2014).

[6] Spataru, S., Hacke, P., Sera, D., Glick, S., Kerekes, T., and Teodorescu, R., "Quantifying solar cell cracks in photovoltaic modules by electroluminescence imaging," in [*2015 IEEE 42nd Photovoltaic Specialist Conference (PVSC)*], 1–6 (June 2015).

[7] Mantel, C., Spataru, S., Parikh, H., Sera, D., Benatto, G., Riedel, N., Thorsteinsson, S., Poulsen, P., and Forchhammer, S., "Correcting for perspective distortion in electroluminescence images of photovoltaic panels," 0433–0437 (06 2018).

[8] Hastie, T., Tibshirani, R., and Friedman, J., [*The elements of statistical learning*], 2nd ed. Springer (2008).

[9] Breiman, L., [*Random forests*], 5–32, Springer (2001).

[10] "https://imbalanced-learn.readthedocs.io/en/stable/generated/imblearn.over_sampling.smote.html,"

[11] *Scikit-learn: https://scikit-learn.org/stable/modules/classes.html* .