AI-assisted Cell-Level Fault Detection and Localization in Solar PV Electroluminescence Images

Ahan M R, Akshay Nambi, Tanuja Ganu, Dhananjay Nahata, Shivkumar Kalyanaraman[1]
Microsoft Research, India and [1]Microsoft, India

ABSTRACT

With the increasing adaption of solar energy worldwide, there is a huge interest to develop systems that help drive efficiency during manufacturing and ongoing operations. Due to various real-world conditions and processes, solar panels develop faults during their manufacturing and operations. The objective of this work is to build an End-to-End Fault Detection system to detect and localize faults in solar panels based on their Electroluminescence (EL) Imaging. Today, the majority of fault detection happens through manual inspection of EL images. To this end, we propose the design and implementation of an end-to-end system that firstly divides the solar panel into individual solar cells and then passes these cell images through a classification + detection pipeline for identifying the fault type and localizing the faults inside a cell. We propose a hybrid architecture that contains an ensemble of multiple CNN model architectures for classification and detection. The ensemble is capable of serving both – monocrystalline and polycrystalline solar panels. The proposed system significantly helps in increasing the efficiency of solar panels and reducing warranty and repair costs. We demonstrate the performance of the proposed system using an open EL image dataset with 95% of cell-level fault prediction accuracy and high recall. The proposed algorithms are applicable and can be extended for other solar applications that use RGB, EL, or thermal imaging techniques.

CCS CONCEPTS

• **Computing methodologies → Computer vision problems**.

KEYWORDS

Fault Detection, Fault Localization, EL Imaging

ACM Reference Format:
Ahan M R, Akshay Nambi, Tanuja Ganu, Dhananjay Nahata, Shivkumar Kalyanaraman[1]. 2021. AI-assisted Cell-Level Fault Detection and Localization in Solar PV Electroluminescence Images. In *The 3rd International Workshop on Challenges in Artificial Intelligence and Machine Learning for Internet of Things (AIChallengeIoT 21), November 15–17, 2021, Coimbra, Portugal.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3485730.3493455

1 INTRODUCTION

Over the last decade, there has been a significant increase in the development and deployment of photovoltaic solar energy generation across the globe. As of 2020, nearly 105 countries have invested in Solar Energy [14] with a total of 580GW of installed capacity.

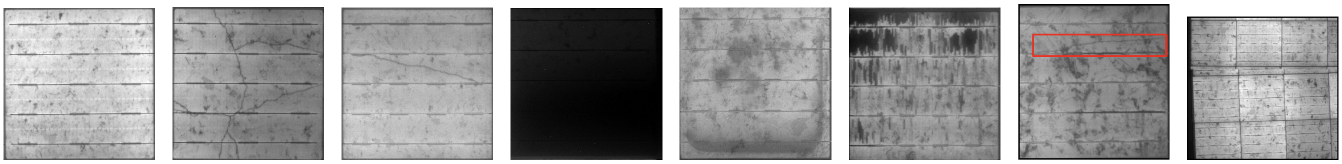
Photo Voltaic (PV) modules are the commercially available basic building block in the solar deployment. PV modules (also interchangeably called as solar panels) can be broadly classified into two types based on the type of the silicon used: (i) Polycrystalline and (ii) Monocrystalline. While majority of the installed PV setups are polycrystalline, there is a significant push towards monocrystalline due to their increased efficiencies. With the increasing capacity of PV modules, it is important to monitor the quality and performance of these modules during manufacturing time as well as during their operations. Due to various real-world conditions during the PV modules manufacturing process, there is a possibility of certain PV cells inside PV modules getting damaged or causing an abnormal behavior with a fall in performance. If these defects are identified during early stages of manufacturing process, the affected cells can be either replaced or repaired, thereby effectively saving major losses in plant efficiency and performance [15] [22] as well as monetary losses and warranty issues.

Electro-luminescence (EL) Imaging is more commonly used in detecting faults and degradation in PV cells [4]. During the EL imaging process, the solar panel being inspected is placed inside a dark chamber and excited by feeding current into the solar cells. The radiative recombination carriers causes light emissions. This light emission is captured by one or multiple cameras mounted above the PV modules, thereby generating the EL image of the PV module. Figure 1 shows EL images of different solar PV cells, which are either working or have some manufacturing defect. These defects are typically categorized into different defect classes such as - (i) Cracks (Micro and Branch Cracks), (ii) Grid Defect and Power Mismatch, (iii) PECVD Marks, (iv) Short Circuiting of the cell, (v) Chipped/Broken cell and (vi) Soldering Defect. Among these defect classes, cracks are the most frequently observed defect type during manufacturing, transportation or deployment of these solar modules. Based on the severity, these cracks can also cause open-circuits in the panel, disconnecting the cell completely from the panel. Grid Defect and Power Mismatch is another common fault in PV cells, which occur mainly due to the short-circuiting of the bypass diode between two bus-bars in a cell, thereby incurring power losses of nearly 18-25% in polycrystalline, and 20-28% in monocrystalline cells [3, 28].

Today, majority of PV manufacturing plants employ a human operator to manually visualize the EL image to determine if the panel has a fault and further investigate which cell in the panel has a fault (see Figure 1). Detecting and identifying the faulty cell and its type is critical to ensure it is either repaired or replaced. Such a manual inspection process is cumbersome and puts significant pressure. Thus resulting in human fatigue and also prone to errors.

AIChallengeIoT 21, November 15–17, 2021, Coimbra, Portugal

ACM ISBN 978-1-4503-9097-2/21/11…$15.00
https://doi.org/10.1145/3485730.3493455



(a) Working **(b) Branch Crack** **(c) Micro Crack** **(d) Short Circuit** **(e) PECVD Marks** **(f) Grid Defect** **(g) Noise Crack** **(h) Bad Stitching**

Figure 1: Different types of Faults in EL Images of PV Cells and current challenges

Recently, automated fault detection approaches in EL images has gained interest in both- academia and industry [10]. However, there are several challenges that still needs to be addressed [11]. ❶ Cell-level fault detection: As we discuss in Section 3.2 majority of the works focus on classifying if an entire PV panel is faulty or working as opposed to detecting the cell which has a fault. Detecting faults at cell level, requires precise cell segmentation which is non-trivial as each EL image is obtained by stitching images from multiple cameras resulting in incorrect stitching which can pose problems during cell segmentation (see Figure 1h). ❷ Presence of wafer marks: EL images are typically low contrast gray-scale images, which makes it difficult to distinguish between foreground (fault) and background (cell image with wafer marks) (see Figure 1g). ❸ Variation in image orientation and brightness: Since EL images are taken through overhead camera, they generally have perspective issues with the presence of angle of tilt, and need perspective distortion corrective along with orientation correction [24]. ❹ Localization of faults: As mentioned earlier, it is critical to detect individual fault types in a cell such as crack, grid defect, etc., and also the location of the fault. Given these faults are generally small in size and are typically hard to distinguish from background it is challenging to localize the faults accurately.

To this end, we have developed an automated end-to-end system for cell-level, fine-grained PV module fault detection using EL images. Specifically, we have developed a robust cell-segmentation pipeline that takes the complete PV module EL image and segments it to the individual cells accurately. The individual cells are then fed to a CNN-based Deep learning pipeline to either classify the status of the cell, i.e., working or specific fault type, or localize the fault in a cell. We have performed extensive testing of the proposed approach on a popular public dataset [6]. During the development of such an automated algorithm, we have focused on reducing the false negatives as missing a faulty cell would result in poor performance and warranty issues. Further, we present several learnings in developing such an automated algorithm. Finally, the proposed algorithms are quite generic and can be applied to different solar images such as EL, RGB, and thermal images. For example, the cell segmentation algorithms can also be used to segment rooftop solar panels and the fault detection algorithms can be applied to detect hotspots in thermal images and so on. Our main contributions are:

(1) We have developed an end-to-end hybrid pipeline to accurately detect and localize faults in EL images for both mono and polycrystalline PV modules.
(2) We have designed a robust algorithm to accurately segment individual cells in a PV module by considering real-world challenges such as variations in image orientation and brightness, etc.
(3) We have evaluated our proposed pipeline on a popular dataset ELPV [6] and our CNN pipeline has an accuracy of 95%.

2 RELATED WORK

While manual quality testing of EL images is the norm today, there are several automated approaches that are being currently proposed. The earliest attempt on understanding and detecting faults in EL images was introduced by Deitsch et al., [7, 10]. The work focused on using classical image processing to segment PV cells by extracting edge-level features, but the algorithm does not perform well in case of high contrast images. The work also does not classify or detect faults in the cell explicitly and is focused on panel level fault detection. Parikh et al., [20] focuses on detecting micro-cracks, finger failures as two top faults in the PV cells as part of a drone-mounted system. The paper solves perspective issues by correcting it using homography transformation. However, the approach primarily focuses on traditional algorithms such as Random Forest, kNN and SVM for classification.

Over the past few years, a lot of papers have focused on solving this problem using deep learning. Ding et al. proposed a transfer learning approach for large scale PV solar plants for detecting faults [12]. Li et al., [16] proposed a VGG based network along with a SVM classifier achieving nearly 90% accuracy. But the drawbacks of these papers are that, the same models do not generalize well on polycrystalline panels, and also is not robust to high wafer marks in the images as all the layers are not trained on EL Dataset due to lack of training data. The work also does not capture many other kind of faults that occur in PV cells which we focus and detect as part of this paper. Akram et al. [1] uses CNN based approach with data augmentation techniques and achieves 93.02% accuracy on classification task but the work is on a binary classification task of faulty and not-faulty classes. Tang et al., [26] proposed a more robust approach by generating samples of EL images using generative methods for augmentation and also benchmarked the model against VGG16, ResNet50[13], etc. Deitsch et al., [9] devised an segmentation approach in [7] and further they proposed an SVM based approach by extracting features using keypoint detection methods such as KAZE [2], SIFT, and used a deep learning based CNN approach. These papers generalized the use of traditional image processing along with Deep Learning approach to differentiate between defective and non-defective panels, obtaining state-of-the-art results. Majority of the above works focus on just panel level fault classification, i.e., is there a fault in the panel instead of which cell has a fault and what type of fault. Further, even if a cell level approach is proposed, they are mostly binary classification, i.e., working or faulty. However, as we described earlier detecting the exact type of fault is critical as the operator can take appropriate actions with regard to replacement or repair. Finally, most of the networks are designed for either mono or polycrystalline, but not both. Since these two PV modules have different characteristics this requires additional considerations to efficiently detect fault in EL images. We will next present our automated end-to-end fault detection pipeline on EL images that can detect faults and its type on each individual cell of the panel.

3 EL IMAGE FAULT DETECTION

EL Imaging is a technique used to determine the quality of electrical contact between PV cells. In this technique, cells are forward biased and made to emit radiations detected by silicon-charged coupled camera device in a dark chamber. The area between busbars in scanned cell images with high luminescence proves higher efficiency of silicon; thereby, proving that dark regions in these EL Images represent faulty parts of the cells with cracks, grid defect or even a broken chipped cell. After generating EL Images of a Solar Panel, it needs to be examined to determine the working condition of every PV cell in the panel. To this end, we have a two-stage pipeline, where we first pass the EL image (entire panel) as input to a cell segmentation algorithm (see Section 3.1). The output of cell segmentation algorithm is individual cells of the panel. These individual cells are then passed through a Deep learning pipeline to classify, detect and localize the type of faults (see Section 3.2).

3.1 Cell Segmentation Algorithm

The primary aim of the segmentation algorithm is to extract individual cells from the panel. We exploit the horizontal and vertical lines in an EL Panel Image [8], such that for every image, we can extract and crop the individual EL cell images (72 in case of 12x6 panel) using Edge and Line detection algorithms along with advanced Image Processing techniques as discussed next.

3.1.1 ***Cropping a panel image into 12 different sub-images.*** We first crop the whole EL Image into K small sub-images or tiles (12 in this case) as shown in the Figure 2. Specifically, to crop the panel image to tiles, we rely on the image dimensions of the panel. We crop the image in such a way that for each of the K cropped images, we get at least X complete cells (6 cells in this case), so that we get all (K x X) cells (72 cells in total - 12x6) from all the cropped images.

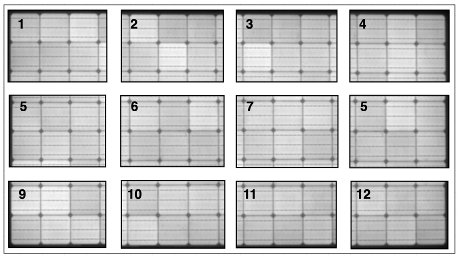


Figure 2: Cropping a panel image into sub-images

3.1.2 ***Deriving cell boundaries for each individual cells.*** For each cropped image or tile, we apply a sequence of pre-processing steps to derive the boundaries of individual cells.

Masking and Line detection: Since the tiled EL images may have noise in the form of wafer markes, we first apply Gaussian Blurring and Histogram Equalization to derive a mask (see Figure 3(b)). After this we process the same image using Laplacian Kernel along with the inRange Thresholding for detecting the lines, based on the range of the pixel values (see Figure 3(c)).

Filtering: We use the K-Nearest Neighbor (KNN) approach, along with the "`Sliding Window`" algorithm with a pixel width of W, based on prior experimentation. We slide the window both horizontally and vertically along the complete cropped image. While sliding through the image, our algorithm counts the black and white pixels concentration, and if number of black pixels exceeds the number of white pixels: *max(blackPixels, whitePixels)*, we assign all the pixels in the window width W in the image to black and vice-versa. As a result, we are able to remove the noise completely from these tiled images as shown in Figure 3(d).

Cell Boundary Extraction: To extract the intersections of all the horizontal and vertical lines, we compare the ratio of the number of black to number of white pixels for the vertical lines and horizontal lines respectively. If the pixel-ratio is greater than a specified threshold T, then we replace the whole vertical/horizontal line to white (255) otherwise we assign whole line to black pixels (0) respectively. After extracting all the horizontal and vertical lines, we operate the Bitwise-AND OpenCV technique on the array consisting of horizontal and vertical lines, to get their intersections. Furthermore, after calculating the intersection of the horizontal and vertical lines we get a series of points for each of the intersection. From these series of points we extract the exact centroid for each of the corners of the cells, by averaging the horizontal and vertical dimension coordinates for each of the intersection.

3.1.3 ***Extracting individual cells.*** Once we have coordinates for the cell boundaries, we now segment the individual cells. The solar cells in the EL Image are almost similar in size and are nearly symmetrical in nature. Thus, we can use this domain knowledge to filter cells whose length, width and diagonal distance between the corners of the cell is within the expected range of an ideal cell (500x500px in this case). We calculate the median of the length, width and diagonal distance for each of the extracted cells from all the K cropped images (12 in this case). If all three dimensions of an extracted cell are within the range of the said threshold from their median lengths, then we label them as correct segmented cells.

3.1.4 ***Extrapolating the Corrected Cells for whole image.*** Even if one out of six cells in the cropped image is extracted correctly as described in the previous step, then we are able to extract all the six cells using the extrapolation. However, there can be few cropped images out of 12 sub-panel images, where the algorithm may not able to detect even a single cell correctly after application of post-processing mentioned previously. As a result due to the lack of reference coordinates for extrapolation, the algorithm might fail to detect cells in the cropped image. Hence, using the rules of symmetry, we assume horizontal length = vertical width of a cell and extrapolate the horizontal and vertical lines extracted to get the coordinates of these cells in cropped images where earlier none of the cells were detected as an alternative.

3.1.5 ***Additional check on the extracted Cells to remove Cell borders and additional margins.*** We found that there were few cells which were not segmented properly, because of alignment and perspective issues and because of the problems we faced while extrapolating the lines of the corrected cells. These incorrect cropped

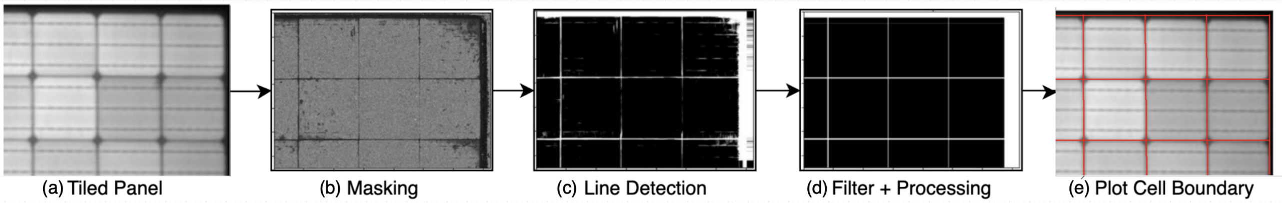

(a) Tiled Panel (b) Masking (c) Line Detection (d) Filter + Processing (e) Plot Cell Boundary

Figure 3: Step-wise Segmentation Output using Classical Image Processing techniques

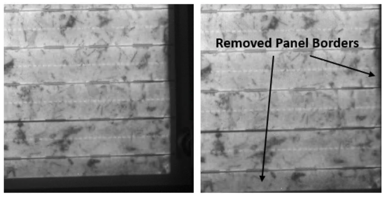


Figure 4: Cell border removal

cells are shown in Figure 4. When these cells were passed through the classification model, these cells are often misclassified because of improper extraction of the cell. So, to improve the performance of our model and make it more robust to change to contrast and brightness of EL Panel Image, we apply two additional steps, i.e., one for removing the cell borders and other for removing the Panel borders present in cropped cell images, generated while cropping the border cells. We crop the borders of the cell as well as the panel image using the "`Contour Detection`" algorithm, thus generating a cell image with no cell borders (black edges). Finally, all the *(K x X)* 72 cropped cell images extracted are passed to the classification pipeline discussed in the next section.

3.2 Fault Classification: Polycrystalline Images

As mentioned earlier, we have developed a hybrid pipeline which works with both mono and polycrystalline EL cells. Figure 5 shows the overview of our end to end pipeline. In this section, we describe the working of the pipeline for polycrystalline EL images. Here, we first classify the cell to one of the classes (working or specific fault) and then if it is a faulty cell, we run an additional object detection pipeline to localize the fault. This is done mainly because of the noise (wafer marks, etc.) present in the polycrystalline images making it harder for object detection algorithms to both detect and localize different faults accurately. We first describe the fault classification approach and then the fault localization.

3.2.1 ***Fault classification using EfficientNet.*** EfficientNet [25] based Classification Model is implemented as part of the pipeline for multi-class classification to differentiate between faulty cell from working cell. We employ a supervised learning approach to train the classification model [19]. The PV cell classification for a polycrystalline type cell is complex because of the lack of clear distinctive features between most of the faulty classes, which often leads to increased False Positives (FPs) and False Negatives (FNs) because of loss in differentiating power between micro cracks and wafer marks (noise) often generated during EL image generation. As a trade-off between number of parameters to train/training time for model to be deployed vs performance metrics, EfficientNet B7 achieves much higher performance on this dataset for multi-class and binary classification task. The architecture is built using a multi-objective Neural Architecture Search (NAS) with the primary function of the base network to scale Depth, Resolution and width by performing a grid search on the parameters (α, β, γ) optimising the objective (Compound Scaling) as follows:

$$\begin{aligned} &\text{depth: } d = \alpha^{\phi}, \text{ width: } w = \beta^{\phi}, \text{ resolution: } r = \gamma^{\phi} \\ &\text{s.t. } \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2, \quad \alpha \geq 1, \beta \geq 1, \gamma \geq 1 \end{aligned} \tag{1}$$

We obtain strong spatial information from EfficientNet model (F) by passing it as an input to a Global Average Pooling layer to generate `GAP(F)`, as it helps collect distinctive features from top of the network and suppresses the noise in the background of the input image by adding "`attention`" to the fault and reducing the unnecessary spatial dimensional information. Thus, concatenating "`X = [GAP(F), F]`" the feature vectors obtained from both, EfficientNet and EfficientNet+GAP, is useful in extracting complete information from the input to generate class labels.

3.2.2 ***Fault localization using Feature Pyramid Network.*** Once the cells are classified to a specific class, the faulty cells are then passed through a object detection network to localize the fault. We employ RetinaNet [17] a Single Shot Detector (SSD) to localize the fault. All the layers of the network is trained on the EL Image Dataset (Full training). In the remainder of this section, we will use localization of crack as an example. However, note that this pipeline can be applied to localize all fault types such as grid defect, short circuit, chipped cells, etc. Our Modified ResNet50 (Backbone) + RetinaNet model focuses mainly on localizing the area of fault type. It differs mainly in terms of the architecture of residual block, as each block has an additional Batch Normalization(BN) layer along with Dropout after the last conv layer of base network. This model architecture is used in our paper for calculating the following:

- Length of fault detected (crack length)
- Total Faulty surface area of the cell (Percentage of defect area)
- Number of faults in EL cell (number of cracks, etc.)

The backbone of the network features a Feature Pyramid Network (FPN) [17] built over ResNet50 which generates convolutional feature vectors for EL cell images at multiple levels of the feature pyramid. As we go higher up in the FPN of the bottom-up architecture, the spatial resolution of input image reduces, thereby making the model more robust and it's capability to detect minute cracks more efficiently.

Another important feature of the network is the usage of focal loss which solves the issue of extreme imbalance classes formed from anchor candidates (positive and negative samples) bounding boxes generated during training.

$$\mathrm{FL}\left(p_{\mathrm{t}}\right) = -\alpha_{\mathrm{t}}\left(1 - p_{\mathrm{t}}\right)^{\gamma} \log\left(p_{\mathrm{t}}\right) \tag{2}$$

It improves the performance of models during training without letting easy negatives immensely contribute to gradients, by focusing

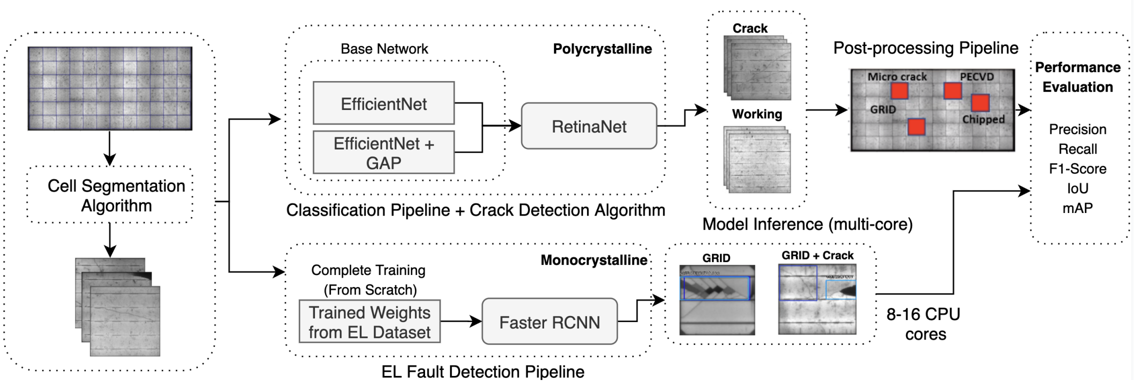


Figure 5: End-to-End EL Fault Detection Architecture

on hard samples and it's contribution to the loss being higher during training. Lastly, the FPN layer output is passed onto Classification and Regression Subnets to classify the generated anchor box candidates from FPN and regress over these EL faulty cell candidates to generate bounding boxes of cracks in the image. These features of the architecture helps us practically meet the requirements setup during the manufacturing process of Solar Panels by determining the characteristics properties of cells as discussed, allowing floor operators to send faulty panels for being reworked or reject them before final lamination in much lesser inference time.

3.2.3 ***Data Augmentation.*** The augmentation module is deployed at training to mainly enhance the performance of the model by improving the generalisation and reduce the impact of overfitting by regularisation. Using Data Generator of tensorflow, Vertical and Horizontal Flips, Randomised cropping, angled crops, edge enhancement and adding gaussian noise proved to be effective in generating more training samples. We mainly focus on proposing a two-stage Data augmentation process, Augmentation during model training and ensembling cell augmentation during model inference. Ensembling the output predictions of the test image over different augmentations improved the precision-recall scores by 9.82% and an improved accuracy on the multi-class classification problem. Thus, the training computation remains the same, but with a slight increase in inference time, the model performance is improved by approximately 10%.

3.3 Fault Detection: Monocrystalline EL Images

In this part of the pipeline, we detect faults occurring primarily in Monocrystalline EL Images using a Faster R-CNN architecture, a two-stage detection algorithm [21]. Faster R-CNN has the best trade-off in terms of accuracy (performance) and latency (speed).

As part of this architecture, all cells from EL Panel image are passed through the base model of the detector which extracts feature vectors. In our case, we used ResNeXt-50 [27] instead of VGG [23] to enhance the speed of training and also reduce overfitting due to the presence of residual and skip connections. The model architecture uses a Region Proposal Network (RPN) to create regions of interest (RoI) by generating k-anchor boxes using a sliding-window approach for different aspect ratios.

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*)$$

The RPN optimises the below loss function, where i is the index of anchor box and P is the predicted probability, to generate suitable proposals which is then passed onto the Classification and Regression heads to detect and label the fault in the cell images. The detector is trained on EL Image Dataset to detect Micro Cracks, Branch Cracks and Grid Defect/Chipped Cells with 85.3% mAP as we discuss in Section 4.3.

4 EXPERIMENTAL SETUP & EVALUATION

In this Section, we discuss the experimental setup of the proposed system- specifying the details of the dataset for training and testing, the evaluation metrics, and specific implementation details.

4.1 Dataset

We use the dataset curated by ZAE Bayern [6] which contains both mono-crystalline and poly-crystalline EL Images. This dataset consists of total 2624 PV Cells, out of which 1074 cells (from 18 PV module EL images) are Monocrystalline and 1550 cells (from 26 PV module EL images) are polycrystalline cells . The dataset is curated and annotated based on defect probability and the level of degradation in the cells in the range of (0,1), split into 4 classes with the defect probabilities of 0, 0.3333, 0.6666 and 1. Each cell image is of size 300x300 pixels and it contains 3 bus-bars in each cell. Firstly, We converted this dataset into binary classification dataset by assigning cell images to either of the two classes- (i) Working (fault probability <0.5) and (ii) Faulty (Fault probability > 0.5). Then, we annotate the faulty cell images for Crack and Grid defect with fault bounding box for training and evaluation.

4.2 Evaluation Metrics

The performance of the classification model for multi-class and binary classification tasks are evaluated using the F1-Score, Precision, Recall and Accuracy.

We also understand the performance of the Object Detection model - RetinaNet and Faster R-CNN using mAP (Mean Average Precision) and by calculating Precision, Recall from IoU (Intersection over Union). IoU is computed based on the overlapping area between predicted box and ground truth bounding box. The prediction is considered True Positive (TP), if IoU >=0.5, False Positive (FP) if IoU<0.5, and False Negative (FN) if there was no overlapping predicted bounding box with respect to the Ground Truth.

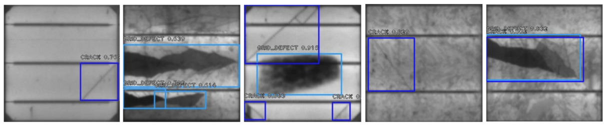


Figure 6: Object Detection Output from ELPV Dataset

Model Architecture	Precision		Recall		F1 Score		Accuracy	Parameters
	Working	Faulty	Working	Faulty	Working	Faulty	(in %)	(in M)
XceptionNet	0.90	0.99	1.00	0.76	0.95	0.86	92.19	22
Inception V3	0.84	0.97	0.99	0.60	0.91	0.74	87.58	23
InceptionResNet V2	0.82	0.93	0.98	0.55	0.89	0.69	84.71	55
DenseNet 101	0.78	0.87	0.97	0.41	0.86	0.56	79.23	24
EfficientNet B7 + GAP	**0.92**	**0.99**	**0.99**	**0.87**	**0.96**	**0.91**	**94.24**	**60**
ResNeXt 50	0.80	0.84	0.95	0.50	0.87	0.62	81.17	23

Table 1: Comparative Performance of Model on ELPV Dataset

Type	Model Architecture	mAP		Inference Time(s)
		mAP@ [0.5:0.95]	mAP@ 0.5	
Poly	**RetinaNet (FPN)**	**0.836**	**0.892**	**0.135**
	Faster RCNN	0.807	0.872	0.288
Mono	RetinaNet (FPN)	0.832	0.887	0.078
	Faster RCNN	**0.88**	**0.942**	**0.154**

Table 2: Performance of Detection Models on both Datasets

4.3 Results and Implementation

In this section, we discuss experimental results for solar panel fault detection using our proposed system. As shown in Figure 5, we firstly execute cell segmentation algorithm on the solar panel EL images. After performing Cell Segmentation, the segmented cells are passed as input to two model pipelines: (i) Approach 1: EfficientNet based classification + FPN based Detection pipeline in case of Polycrystalline cell images, and (ii) Approach 2: Modified Faster R-CNN approach in case of Monocrystalline cells.

In Table 1, we share the performance of Approach 1 on evaluation dataset to show the performance improvement in terms of precision-recall for EfficientNet + Global Average Pooling approach, benchmarked against some of the current state-of-the-art classification algorithms. This architecture outperformed all the other architectures with high recall and precision for Faulty classes, which was one of our primary objectives ensuring the model predictions detect all faults with very low False Negatives. We can see that, the overall accuracy of our approach is 95% with working class having 96% F1 score and faulty class having 91% F1 score.

We also benchmark and validate the performance of both the detection models proposed in the paper- RetinaNet (FPN Detection) + EfficientNet B7(Classification) and Faster R-CNN on monocrystalline and poly-crystalline datasets in Table 2. Due to the presence of high wafer marks and noise, polycrystalline images as input to both models clearly show that RetinaNet being a single stage detector still outperforms a two-stage detector due to the advantages of model architecture discussed in earlier sections with nearly 50% lower inference time. However, on monocrystalline dataset, Faster RCNN maintains a clear performance boost with mAP@[0.5:0.95] at 0.88 as compared to 0.832 for RetinaNet. This demonstrates that Faster R-CNN outperforms RetinaNet pipeline on mono-crystalline cells. Figure 6 shows some qualitative results of our pipeline on both mono and poly-crystalline panels.

Optimizing Cell Segmentation: The cell segmentation algorithm is the first process in the fault detection pipeline [5]. Since the algorithm splits the EL Panel into K sub-panels and performs a cell-splitting algorithm on each of these K sub-panels to obtain individual cells, we optimise it by parallelizing this process and running it on a multi-core CPU. We used storage optimized Azure F-16 VMs, which have 16 CPU cores with 32GB memory to run 16 processes in parallel using the Multiprocessing library [18]. The additional speed-up was achieved by parallelizing the process, making the pipeline nearly 4.8x times faster than a 4-core CPU.

Speed-up in Model Inference: Once the cells are extracted, the Classification + Crack Detection pipeline is executed with these cell images as input. The model inference was also parallelized using Map and Pool methods of multiprocessing library to infer 16 cell images in parallel at once, thereby performing model inference on an entire EL image (i.e., 72 cells) in 0.86s per panel. Depending on the number of cores used on the deployed machine, the inference time per panel ranges between 0.8-2s per Panel.

5 CONCLUSIONS

In this work, we present a holistic system for detection and localization of faults in monocrystalline and polycrystalline solar panels based on their electroluminescence imaging during manufacturing time. The main objective of this fault detection system is to provide high accuracy at the cell level and no fault should go undetected through the system (low false negatives). We present the system architecture and algorithms that take EL image of a solar panel as an input and firstly run it through the cell segmentation algorithm to divide this image into individual cell-images. Then the individual cell images are passed through fault classification and detection pipeline based on CNN model architecture to classify the cell as working vs. faulty and localize the fault within the faulty cell. We also provide the evaluation results of the proposed approach on open ELPV dataset [6]. We further present our learnings based on error analysis during evaluation, and the practical system implementation details for running the proposed system for real-world manufacturing setup. The main advantage of the proposed system is that it helps in detecting the faults not only at the panel-level, but also at the individual cell-level within the panel and also capable of localizing this fault within the cell image. This significantly reduces the time required to identify the faulty cell and repair/replace that cell during early stages of the manufacturing and thereby increasing the efficiency of these solar panels. It further reduces warranty and labor costs remarkably. The proposed algorithms for cell segmentation and fault detection are quite generic and can be easily applied in both manufacturing and on-field scenarios, and also to other problem setting during solar operations for solar panel images captured through RGB, EL and thermal images.

REFERENCES

[1] M. Waqar Akram, Guiqiang Li, Yi Jin, Xiao Chen, Changan Zhu, Xudong Zhao, Abdul Khaliq, M. Faheem, and Ashfaq Ahmad. 2019. CNN based automatic detection of photovoltaic cell defects in electroluminescence images. *Energy* 189 (2019), 116319. https://doi.org/10.1016/j.energy.2019.116319

[2] Pablo Fernández Alcantarilla, Adrien Bartoli, and Andrew J. Davison. 2012. KAZE Features. In *Computer Vision – ECCV 2012*, Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 214–227.

[3] Ali Assi and Mohammad Al-Amin. 2013. Reducing the parasitic loss of c-Si solar cells. In *2013 25th International Conference on Microelectronics (ICM)*. 1–4. https://doi.org/10.1109/ICM.2013.6734955

[4] Karl Bedrich, Matevz Bokalič, Martin Bliss, Marko Topič, Thomas R. Betts, and Ralph Gottschalg. 2018. Electroluminescence Imaging of PV Devices: Advanced Vignetting Calibration. *IEEE Journal of Photovoltaics* 8, 5 (2018), 1297–1304. https://doi.org/10.1109/JPHOTOV.2018.2848722

[5] G. Bradski. 2000. The OpenCV Library. *Dr. Dobb's Journal of Software Tools* (2000).

[6] Claudia Buerhop-Lutz, Sergiu Deitsch, Andreas Maier, Florian Gallwitz, Stephan Berger, Bernd Doll, Jens Hauch, Christian Camus, and Christoph J. Brabec. 2018. A Benchmark for Visual Identification of Defective Solar Cells in Electroluminescence Imagery. In *European PV Solar Energy Conference and Exhibition (EU PVSEC)*. https://doi.org/10.4229/35thEUPVSEC20182018-5CV.3.15

[7] Sergiu Deitsch, Claudia Buerhop-Lutz, Andreas K. Maier, Florian Gallwitz, and Christian Riess. 2018. Segmentation of Photovoltaic Module Cells in Electroluminescence Images. *CoRR* abs/1806.06530 (2018). arXiv:1806.06530 http://arxiv.org/abs/1806.06530

[8] Sergiu Deitsch, Claudia Buerhop-Lutz, Evgenii Sovetkin, Ansgar Steland, Andreas Maier, Florian Gallwitz, and Christian Riess. [n.d.]. Segmentation of photovoltaic module cells in uncalibrated electroluminescence images. 32, 4 ([n. d.]). https://doi.org/10.1007/s00138-021-01191-9

[9] Sergiu Deitsch, Vincent Christlein, Stephan Berger, Claudia Buerhop-Lutz, Andreas Maier, Florian Gallwitz, and Christian Riess. 2019. Automatic classification of defective photovoltaic module cells in electroluminescence images. *Solar Energy* 185 (2019), 455–468. https://doi.org/10.1016/j.solener.2019.02.067

[10] Sergiu Deitsch, Vincent Christlein, Stephan Berger, Claudia Buerhop-Lutz, Andreas K. Maier, Florian Gallwitz, and Christian Riess. 2018. Automatic Classification of Defective Photovoltaic Module Cells in Electroluminescence Images. *CoRR* abs/1807.02894 (2018). arXiv:1807.02894 http://arxiv.org/abs/1807.02894

[11] Mahmoud Dhimish and Violeta Holmes. 2019. Solar cells micro crack detection technique using state-of-the-art electroluminescence imaging. *Journal of Science: Advanced Materials and Devices* 4, 4 (2019), 499–508. https://doi.org/10.1016/j.jsamd.2019.10.004

[12] Shihao Ding, Qiang Yang, Xiaoxia Li, Wenjun Yan, and Wei Ruan. 2018. Transfer Learning based Photovoltaic Module Defect Diagnosis using Aerial Images. In *2018 International Conference on Power System Technology (POWERCON)*. 4245–4250. https://doi.org/10.1109/POWERCON.2018.8602188

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. *CoRR* abs/1512.03385 (2015). arXiv:1512.03385 http://arxiv.org/abs/1512.03385

[14] IEA. [n.d.]. IEA (2019), World Energy Outlook 2019, IEA, Paris. ([n. d.]). https://www.iea.org/reports/world-energy-outlook-2019

[15] Nallapaneni Manoj Kumar, Ramjee Prasad Gupta, Mobi Mathew, Arunkumar Jayakumar, and Neeraj Kumar Singh. 2019. Performance, energy loss, and degradation prediction of roof-integrated crystalline solar PV system installed in Northern India. *Case Studies in Thermal Engineering* 13 (2019), 100409. https://doi.org/10.1016/j.csite.2019.100409

[16] Xiaoxia Li, Qiang Yang, and W. Yan. 2018. Intelligent Fault Pattern Recognition of Aerial Photovoltaic Module Images Based on Deep Learning Technique.

[17] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. 2017. Focal Loss for Dense Object Detection. *CoRR* abs/1708.02002 (2017). arXiv:1708.02002 http://arxiv.org/abs/1708.02002

[18] M.M. McKerns, L. Strand, T. Sullivan, A. Fang, and M.A.G. Aivazis. [n.d.]. Building a framework for predictive science. *Proceedings of the 10th Python in Science Conference, 2011* ([n. d.]). http://arxiv.org/pdf/1202.1056

[19] Urtzi Otamendi, Iñigo Martinez, Marco Quartulli, Igor G. Olaizola, Elisabeth Viles, and Werther Cambarau. 2021. Segmentation of cell-level anomalies in electroluminescence images of photovoltaic modules. *Solar Energy* 220 (2021), 914–926. https://doi.org/10.1016/j.solener.2021.03.058

[20] Harsh Rajesh Parikh, Yoann Buratti, Sergiu Spataru, Frederik Villebro, Gisele Alves Dos Reis Benatto, Peter B. Poulsen, Stefan Wendlandt, Tamas Kerekes, Dezso Sera, and Ziv Hameiri. 2020. Solar Cell Cracks and Finger Failure Detection Using Statistical Parameters of Electroluminescence Images and Machine Learning. *Applied Sciences* 10, 24 (2020). https://www.mdpi.com/2076-3417/10/24/8834

[21] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (Eds.), Vol. 28. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf

[22] Lorenzo Rigutti, Ivan Blum, Deodatta Shinde, David Hernández-Maldonado, Williams Lefebvre, Jonathan Houard, François Vurpillot, Angela Vella, Maria Tchernycheva, Christophe Durand, Joël Eymery, and Bernard Deconihout. 2014. Correlation of Microphotoluminescence Spectroscopy, Scanning Transmission Electron Microscopy, and Atom Probe Tomography on a Single Nano-object Containing an InGaN/GaN Multiquantum Well System. *Nano Letters* 14, 1 (2014), 107–114. https://doi.org/10.1021/nl4034768 arXiv:https://doi.org/10.1021/nl4034768 PMID: 24397602.

[23] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv:1409.1556 [cs.CV]

[24] Evgenii Sovetkin and Ansgar Steland. 2018. Automatic Processing and Solar Cell Detection in Photovoltaic Electroluminescence Images. *CoRR* abs/1807.10820 (2018). arXiv:1807.10820 http://arxiv.org/abs/1807.10820

[25] Mingxing Tan and Quoc V. Le. 2019. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *CoRR* abs/1905.11946 (2019). arXiv:1905.11946 http://arxiv.org/abs/1905.11946

[26] Wuqin Tang, Qiang Yang, Kuixiang Xiong, and Wenjun Yan. 2020. Deep learning based automatic defect identification of photovoltaic module using electroluminescence images. *Solar Energy* 201 (2020), 453–460. https://doi.org/10.1016/j.solener.2020.03.049

[27] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. 2016. Aggregated Residual Transformations for Deep Neural Networks. *arXiv preprint arXiv:1611.05431* (2016).

[28] Hong Yang, He Wang, Dingyue Cao, Dangmin Sun, and Xiaobao Ju. 2015. Analysis of Power Loss for Crystalline Silicon Solar Module during the Course of Encapsulation. *International Journal of Photoenergy* 2015 (04 2015), 1–5. https://doi.org/10.1155/2015/251615