

CNN based automatic detection of photovoltaic cell defects in electroluminescence images

M. Waqar Akram ^{a, d}, Guiqiang Li ^{b, *}, Yi Jin ^{a, **}, Xiao Chen ^c, Changan Zhu ^a,
Xudong Zhao ^b, Abdul Khaliq ^d, M. Faheem ^d, Ashfaq Ahmad ^{a, e}

^a Department of Precision Machinery and Instrumentation, University of Science and Technology of China, Hefei, Anhui, 230026, China

^b School of Engineering, University of Hull, Hull HU6 7RX, UK

^c State Key Laboratory of Fire Science, University of Science and Technology of China, Hefei, Anhui, 230026, China

^d Faculty of Agricultural Engineering & Technology, University of Agriculture, Faisalabad, Pakistan

^e Department of Electronics and Electrical Systems, The University of Lahore, Pakistan

ARTICLE INFO

Article history:

Received 24 April 2019

Received in revised form

29 September 2019

Accepted 10 October 2019

Available online 16 October 2019

Keywords:

photovoltaic (PV) modules

Automatic defect detection

Electroluminescence

Deep learning

Convolutional neural network (CNN)

PV cell cracking

ABSTRACT

Automatic defect detection is gaining huge importance in photovoltaic (PV) field due to limited application of manual/visual inspection and rising production quantities of PV modules. This study is conducted for automatic detection of PV module defects in electroluminescence (EL) images. We presented a novel approach using light convolutional neural network architecture for recognizing defects in EL images which achieves state of the art results of 93.02% on solar cell dataset of EL images. It requires less computational power and time. It can work on an ordinary CPU computer while maintaining real time speed. It takes only 8.07 ms for predicting one image. For proposing light architecture, we perform extensive experimentation on series of architectures. Moreover, we evaluate data augmentation operations to deal with data scarcity. Overfitting appears a significant problem; thus, we adopt appropriate strategies to generalize model. The impact of each strategy is presented. In addition, cracking patterns and defects that can appear in EL images are reviewed; which will help to label new images appropriately for predicting specific defect types upon availability of large data. The proposed framework is experimentally applied in lab and can help for automatic defect detection in field and industry.

© 2019 Elsevier Ltd. All rights reserved.

1. Introduction

Photovoltaic (PV) modules experience thermo-mechanical stresses during production and subsequent life stages. These stresses induce cracks and other defects in the modules which may affect the power output [1]. Cell cracking is one of the major reasons for power loss in PV modules [2]. Therefore, PV modules and cells need to be monitored during manufacturing and later stages for obtaining maximum output [3]. In this context, maintenance and inspection of PV modules has received significant attention during the last few years.

Various approaches are used to detect defects in the PV modules and to assess their impact. Electrical measurements are a basic approach for silicon cell characterization [4]. The analysis of

variations in current-voltage (IV) characteristics can help to detect module degradation, if its magnitude is large. However, small defects hardly influence IV characteristics [5], making them difficult to identify through IV-curve analysis. Although the IV-curve gives information about effect of series and parallel resistances, it does not identify the defective cell/region [6]. Thermography is also used in few studies [6–9] for defect detection, but it has few limitations. The author [10] concludes that a hot region in IR image is not always a defect. Furthermore, it is difficult to distinguish between high series and low parallel resistance in IR images [11]. It is also difficult to identify exact position of defect in IR images [11]. In addition, IR cameras have relatively lower resolution making it difficult to identify small defects such as micro-cracks [3].

Electroluminescence (EL) imaging is another approach used for defect detection in PV modules which offers imaging at much higher resolution and can identify micro-cracks. It is one of the mostly used cell characterization technique [2,12–16], which involves imaging of infrared radiations emitting from PV modules during forward bias condition. The EL images are greyscale images

* Corresponding author.

** Corresponding author.

E-mail addresses: guiqiang.li@hull.ac.uk (G. Li), jinyi08@ustc.edu.cn (Y. Jin).

obtained in dark. The micro-cracks and other defects are appeared as dark grey lines and/or regions in EL images. The visual examination of EL images is a time consuming and expensive task. It requires proper experience and study to examine EL image. Furthermore, visual inspection is only possible at small-scale. For large scale, automatic detection methods are of huge importance. In addition, rising production quantities of PV modules and World's increasing interest in PV made automatic PV inspection even more important and necessary.

Various studies are conducted in this field, but with a few applications to EL imaging. The study [17] presented a fault detection scheme using image processing techniques. The study [18] used Support Vector Machine (SVM) for crack classification. However, photoluminescence (PL) images are used in study. The study [19] presented a pattern recognition approach based on local descriptors and SVM for crack detection in PV modules. However, PL/IR images are used in study. The study [20] use Fourier image reconstruction to detect solar cells having defects. However, this method has complexities in defect detection with more complex shapes due to shape assumption and it takes 0.29 s to inspect only one solar cell. Their study inspected only few defects such as small cracks, finger interruptions, and breaks. The study [21] proposed defect detection technique based on independent component analysis. However, finger interruptions are treated equally to micro-cracks and deep-cracks. The study [22] detected snail trails and dust in large-scale PV farm with use of unmanned aerial vehicle. It used image processing algorithms for automatic detection. However, it considers only visible defects. The study [23] presented an enhanced crack segmentation setup for EL images using vesselness algorithm. However, they only work on cracks having length greater than 20 mm. They did not consider other defects and cracks having length less than 20 mm.

In addition to image processing techniques and other traditional pattern recognition methods, few studies [24–27] used deep learning but with application to visible defects only. The study [24] carried out deep learning-based inspection of large-scale PV farm with use of unmanned aerial vehicle. The study [25] also carried out deep learning-based automatic inspection of PV system with use of unmanned aerial vehicle. However, these studies only consider visible defects. The study [26] also carried out deep learning-based detection of visible defects with use of aerial imaging for PV modules. The study [27] automatically detects visible defect in solar cells using multispectral CNN. Another study [28] used CNN for quality rating and process control during PV cell manufacturing. The study [29] used CNN to predict soiling localization and soiling category in PV modules. In contrast to deep learning application to detect visible and other defects, a study [3] used deep learning for classification of defects in EL images. This study used public solar cell dataset [30,31]. However, it is more hardware-demanding approach and there is still a large room for improvement in performance.

The literature review shows limitations of existing EL studies like they detect only few defects or gain low performance with high computational cost or use small datasets that capture only specific defects, etc. Based on excellent performance of deep learning in other fields and literature review, we presume that it is possible to achieve efficient automatic PV defect detection with low computational cost by adopting appropriate CNN architectures and generalization strategies. In this context, to overcome existing limitations, we proposed a novel approach using a light CNN architecture for automatic defect detection in EL images of PV modules that achieved state of the art results (93.02%) on first publicly available solar cell dataset. It requires less computational power and time; and can work on an ordinary non-GPU computer while maintaining real time speed. Dealing with data scarcity and model

performance, we have addressed these challenges: (1) Improving defect detection accuracy in PV EL images is a challenging task as presence of crystallographic defects also shown up as dark lines in the EL images which makes automatic classification difficult, (2) Generalization of deep learning models on small datasets is a real challenge.

Addressing the afore-mentioned challenges, we have obtained high defect detection accuracy coping with data scarcity. The specific contributions of the present study are:

- 1) We proposed an approach using a light CNN architecture that achieved state of the art results on EL image classification. We have achieved a high accuracy of 93.02% using low computational power and time. The proposed framework is experimentally applied and can help for automatic PV defect detection in the field and industry.
- 2) Dealing with data scarcity, we apply various data augmentation operations on EL images and evaluate them through experiments.
- 3) We have also discussed cracking patterns and other defects that can be appeared in EL images; which will help to label images appropriately for predicting specific defect types upon availability of additional training data.

2. Detection of failures in PV module

This section firstly introduces faulty conditions in PV module and the failures leading to these conditions. The effects of these failures on IV characteristics of modules is also discussed. Subsequently, electroluminescence used to detect these failures and defects appeared in electroluminescence images are discussed. Last of all, image recognition for defect detection in EL images with CNN is discussed.

2.1. Faulty conditions in PV module

The common faulty conditions in the PV module are mismatch fault, open circuit fault, bridging fault, earth fault, etc. [32]. These faults may occur due to different kind of failures in module. For example, the defective cells may have non-uniform current density under reverse bias condition due to presence of shunts that may arise from material/manufacturing defects [33]. Moreover, resistive/failed solders and cell shading also lead to mismatch [34]. Similarly, poor connection between cells leads to open circuit fault. These failures affect I–V characteristics of module [35]: (1) a lower short-circuit current is caused by transparency loss in encapsulant and due to glass corrosion, (2) I–V curve near short-circuit current gets sloped due to shunt paths in solar cells and cell interconnections. The slight mismatch in cells may also leads to slope, (3) I–V curve near open circuit voltage has lower slope, if module series resistance is increased, (4) a lower open circuit voltage is due to failure of cell interconnections, short circuit in cell to cell, etc., (5) steps in I–V curve occurs due to damaged cells, defect in diode, and heavy cell mismatch conditions [35].

The cracks and other faults in solar cells like shunt fault on solar cell, shunt fault due to cell interconnect, finger failure, silicon material defect, etc. can be detected by EL imaging [35]. The EL imaging and defects appearance in EL images are described in following sub-sections.

2.2. Electroluminescence for failure detection

In electroluminescence (EL), PV cell emits radiation due to electron-hole recombination during forward bias condition. The

emitted radiations are in near infrared range. The rate of electron-hole recombination per energy for nondegenerate electron & hole concentrations is given in equation (1) [36].

$$g_r(x, \hbar\omega) = \alpha(\hbar\omega) \frac{(\hbar\omega)^2}{4\pi^2 \hbar^3 c^2} \exp\left(\frac{-\hbar\omega}{K_B T}\right) \exp\left(\frac{eV}{K_B T}\right) \quad (1)$$

In the above equation, $g_r(x, \hbar\omega)$ is rate of spontaneous recombination, $\alpha(\hbar\omega)$ is absorption coefficient, \hbar is reduced Planck constant, K_B is Boltzmann constant, e is electron charge, and V is voltage. If crack and/or other defect is present in a PV cell/cell region, the current passage is reduced or blocked during forward bias condition at that location depending upon the type and severity of defect. Such region emits reduced or zero near infrared radiations as there is reduced or not at all electron-hole recombination. The cell/region with defects is shown up relatively darker in EL image as it emits less radiations. For example, cracks are shown up as dark grey lines in EL images [as shown in sub-section 2.3, see Fig. 2]. Similarly, other defects appear as dark grey regions in EL images [as shown in sub-section 2.3, see Fig. 3]. On the other hand, normal operating cells/regions experienced normal electron-hole recombination during forward bias condition and emits relatively more radiations; thus, appear relatively brighter in EL images. For example, upper half region of cell in Fig. 2a (next sub-section 2.3) is relatively brighter than lower half region.

The EL imaging setup is shown in Fig. 1. For EL imaging, a variable DC power supply capable of applying short circuit current (I_{SC}) of modules under study is used i.e. it should provide enough voltage to achieve I_{SC} . The modules under study are supplied with current greater than I_{SC} in forward direction in dark environment. The laboratory temperature should be maintained at 25°C during experiments [35]. The dark environment is chosen because proper

EL exposures start at approximately 50 mW/m² [35]. Near infrared radiations are emitted which are imaged by EL cameras. The EL camera must be sensitive to emission spectrum of PV material under testing [11].

2.3. Detected failures in electroluminescence images

This sub-section introduces various kinds of defects and cracking patterns, which can be detected from EL images of PV modules.

For a supervised learning system, each EL image needs to be classified manually first for labelling purpose. This is an expensive and time-consuming task; and involves possibility of human error. Therefore, it needs to be completed with great care. Once, this task gets completed and an efficient model is developed; afterwards, all future images can be classified automatically. For this purpose, EL images need to be analyzed carefully with respect to cracks and other defects. The defects appeared in EL images are discussed in following paragraphs.

Cracks can appear in different lengths and location in a PV cell. Manufacturing process is largest source for initiation of cracks and risk of crack growth increases afterwards [1]. There are three sources for cell cracking during manufacturing process with different probabilities [35]:

1. Stresses experienced during soldering initiate cracks at cell interconnect ribbon. Such cracks are mostly found at the end or start of connector as the highest stress is experienced by this region [1,37]. These kinds of cracks are most common.
2. Crack caused by needle pressing on silicon wafer during manufacturing. This is termed as cross crack/cross crack line.

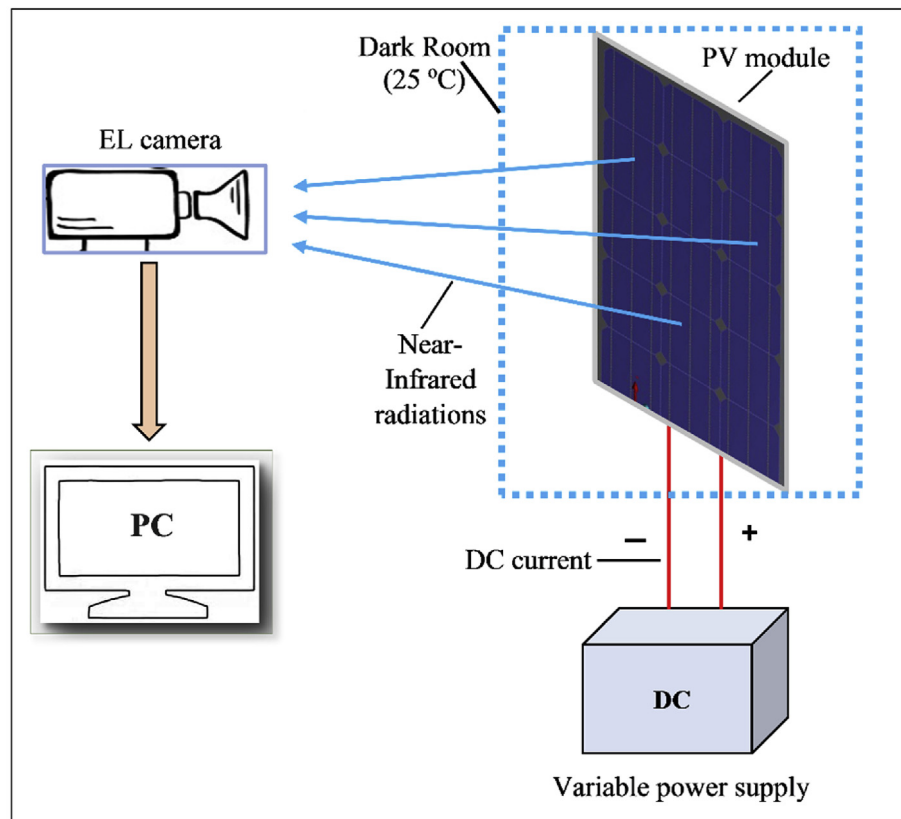


Fig. 1. EL imaging setup.

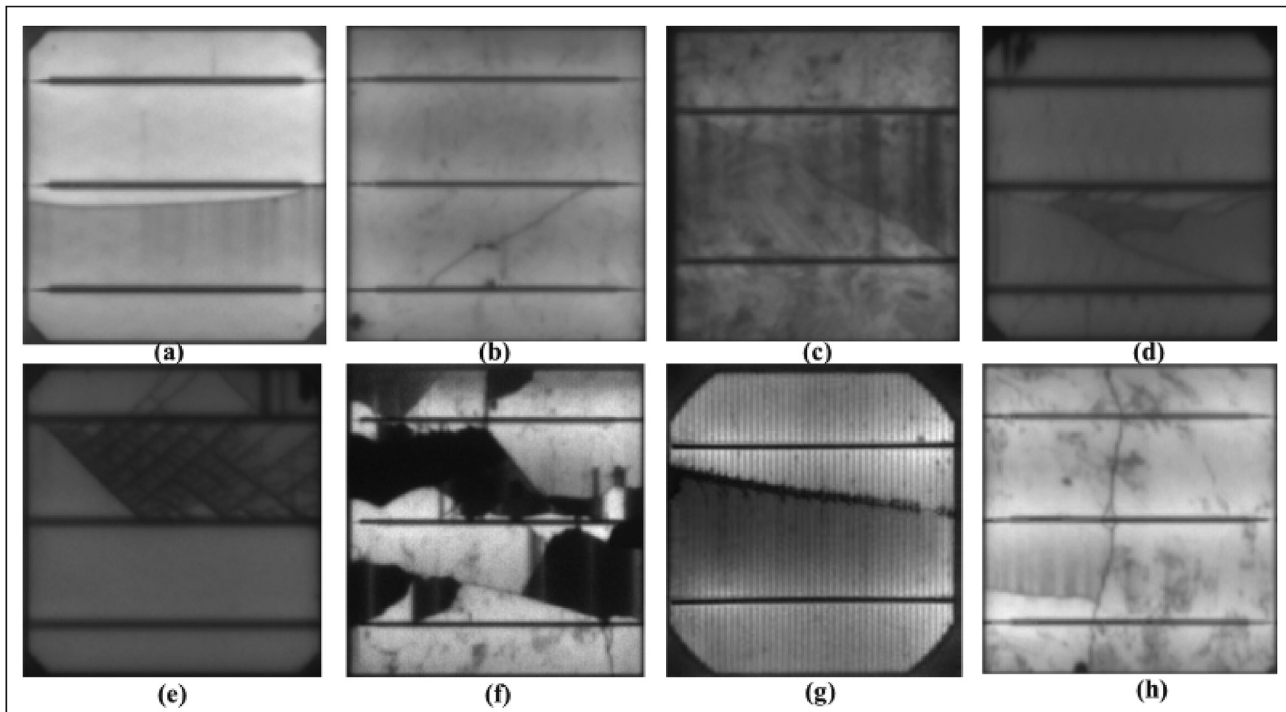


Fig. 2. Crack orientations appeared in EL images of PV modules (a) parallel to busbar (b) $+45^\circ$ (c) -45° (d) several directions (e) dendritic/branched (f) deep cracks isolating cell parts (g) cross line [35] (h) perpendicular to busbar.

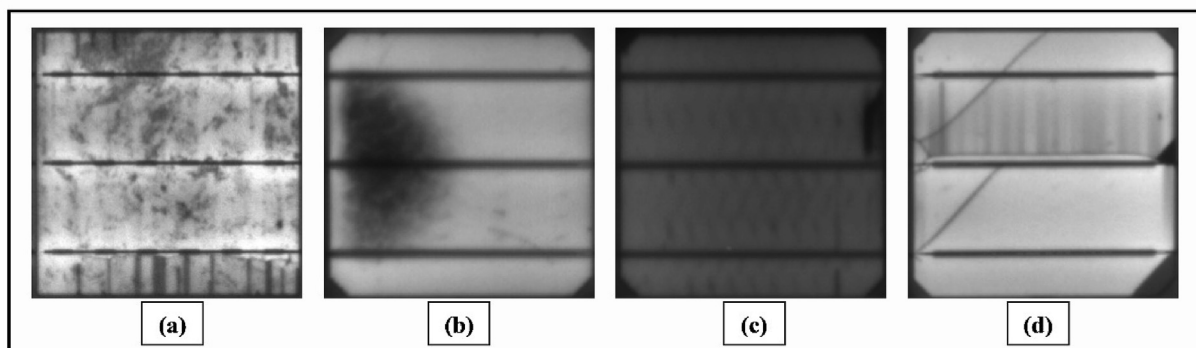


Fig. 3. Other defects detectable from EL images of modules (a) finger failure (b) silicon material defect (c) Contact forming failure (tire like imprint) (d) finger failure along cracks.

3. Cell bouncing against some hard object initiate crack at cell edge.

After manufacturing, other kinds of cracks may appear during subsequent stages as various kinds of thermo-mechanical loads are experienced by modules [1]. Dendritic or branched pattern of cracks appears in a module when it is exposed to heavy shock loads. According to study [35], such kind of loads are applied in field during transport, module dropping, snow loads, etc.

Form literature review as discussed above and study [35], it is inferred that a crack may be of following pattern in an EL image:

1. A deep crack isolates cell parts that appear as dark black regions. These isolated cell parts are electrically disconnected.
2. A cell crack is shown up as a grey color line in an EL image. The width and greyscale of line is mostly consistent over whole length.

3. A crack having an angle of $\pm 45^\circ$ to about $\pm 5^\circ$ to the fingerprint of cell is partially parallel to cell fingers. In this case, crack is shown up as a wavy step function.
4. A dark grey line may be a structural defect in a silicon wafer that was neighbor in ingot. Such cases must be dealt carefully.
5. If the intensity of EL emission changes rapidly in a grey line it is a crack. Therein, the crack already lowers the electrical conductivity of metallization.
6. The cracks mostly started or ended at busbar or cell edge except cross cracks. It is very rare to find a crack showing some other characteristics.
7. Cross cracks or cross crack lines are mostly found in cell center.

The common crack orientations appeared in EL images are parallel to busbar, perpendicular to busbar, $+45^\circ$, -45° , several directions, dendritic, deep cracks, cross line, etc. These are shown in Fig. 2. It is observed that cracks perpendicular to busbar and cross cracks are very rare.

In addition to cracks, other common defects which are found in EL images of modules are shunt faults, finger failure, material defect, contact forming failure, etc. as shown in Fig. 3.

2.4. Convolutional neural networks for defects recognition in EL images

The Convolutional neural networks (CNNs) are developed from the study of visual cortex structure in brain [38]. In past few years, they are widely used in self-driving vehicles, image classification, voice recognition, etc.

Convolutional layer is most important block of a CNN structure. The convolutional layer is made of neurons that have learnable weights and biases (trainable parameters) [38]. Neurons in the first conv. layer are only connected to the pixels present in their receptive fields. In turn, each neuron in the second conv. layer is only connected to the neurons present in a specific local field in first layer [38]. This structure makes network to focus on low-level (local) features in first layer and subsequently these features are assembled into higher level features in following layer and then, this process continues. Therefore, the initial layers of CNN extract low-level (local) features of images like edges, curves, colors, etc. and the upper layers extract specific high-level features of images like shapes, etc. This structure is also common in natural real-world pictures that makes CNN networks to work well for image recognition. The mathematical equation for computation of neuron output in a convolutional layer is given below [38].

$$z_{i,j,k} = b_k + \sum_{u=1}^{f_h} \sum_{v=1}^{f_w} \sum_{k'=1}^{f_n} x_{i',j',k'} \cdot w_{u,v,k',k} \quad \text{with} \quad \begin{cases} i' = u \cdot s_h + f_h - 1 \\ j' = v \cdot s_w + f_w - 1 \end{cases} \quad (2)$$

where, $z_{i,j,k}$ is the neuron output. This neuron is located in i th row and j th column in a feature map k of a convolutional layer l . s_w is horizontal stride, s_h is vertical stride, f_w is width of receptive field, f_h is height of receptive field, and f_n is number of feature maps present in preceding layer (layer $l-1$). $x_{i',j',k'}$ is the output of neuron which is present in preceding layer $l-1$. This neuron is present in i' th row and j' th column in feature map k' (or channel k' , if it is input layer). b_k is bias for k feature map (in l layer). $w_{u,v,k',k}$ is the connection weight between any of the neuron in map k of l layer and its input present at u th row and v th column, and the k' feature map.

Typical CNN architectures consist of few convolutional layers in start, then pooling layer, then few other convolution layers, then other pooling, and so on. At top of network, few fully connected layers and final decision layer is present. Unlike convolutional layers, neurons in fully connected layers do not have a limited receptive field. In addition to above layers, CNN networks usually consist of batch normalization and dropout layer [39].

The main advantages of neural networks over traditional machine learning algorithms are high accuracy and less prediction time. But there are few limitations that they have large computational requirements and they require large training data. Among Neural networks, the major benefits of CNN networks over fully connected deep neural networks (DNN) for image classification are [38]: (1) CNN has relatively fewer parameters that leads to fast training, reduction of overfitting risk, and requirement of less train data, (2) can detect a specific feature located anywhere on the image if it once learned a filter which can detect that feature. As images have repetitive features, CNNs can be generalized relatively better, (3) lower layers of CNN detect lower-level features and the upper layers combine these features into larger higher-level features. This usually works well with real-world natural images, that

give CNNs a decisive head start in comparison to DNNs, etc.

3. Methodology

This section presents proposed framework and CNN architecture selection process. We used isolated deep learning approach in this study. Isolated learning is task learning from scratch [40] without considering previously learned knowledge relevant to other tasks. The model developed in our study is just based on our subject EL images dataset. It is fully trained from scratch.

Series of CNN architectures are investigated and finally an excellent light network is proposed. Their details are given in this section. To cope with data scarcity, data augmentation approaches are used to expand our training data. The details of data augmentation are also given here. In addition, the model optimization process and the model evaluation criteria are also discussed in this section.

The flow chart of used approach is given in Fig. 4. Firstly, data augmentation operations are carried out on EL images to obtain more data. These operations help to improve the model performance. Then the images are fed to final selected six-layer CNN architecture (four convolutional layers and two fully connected layers). The model is trained and optimized to learn to classify EL images. The initial convolutional layers learn low-level features of EL images and the upper convolutional layers learn high-level specific features of EL images. Subsequently, the model is evaluated on the testing data with respect to different performance measures. The developed model can predict whether an EL image is defective or normal.

3.1. Data augmentation

Large datasets are mandatory in computer vision for achieving good performance to efficiently map a particular input to some output. As we have small training data, we need to expand our dataset. To get more data, data augmentation operations are applied to original images in our study. For classification in our case, it is not important whether a crack orientation is at positive or negative angles. Similar to this, crack appearing at upper edge or lower edge does not matter. Thus, data augmentation can be successfully applied.

We performed offline data augmentation. We have rotated images to 180° and $\pm 3^\circ$. Since busbars are also found in vertical setup, we also rotated the images to 90° and 270° . In addition, flipping is also applied along x-axis and y-axis. Furthermore, the images of defective three-busbar cells are firstly cropped to obtain two busbar shape keeping in view the importance of useful information and then, they are resized. This enlarges the defect size. If the sub image does not contain defective area, we select random cropping regions repeatedly to get it. Similarly, the images of defective two-busbar cells are cropped and resized in such a way that useful information is retained. This enlarges the defect size. The generated new samples are carefully checked with respect to labels.

We also performed few additional operations on specific defective cell images to get more useful data. In study [41], gaussian blur was found a successful data augmentation strategy. Similarly, we also try this strategy in present case and find it useful (discussed in section 4.2). Therein, the images showing dendritic and deep cracks are augmented using gaussian blur to obtain new images of same labels with different information. It reduces the details of darker images giving different useful information. The authors [38] recommended to use contrast for expanding image dataset, to make model more tolerant to lighting conditions. We use contrast augmentation operation for images showing parallel, perpendicular, several directions, and $\pm 45^\circ$ cracks. This operation also helps

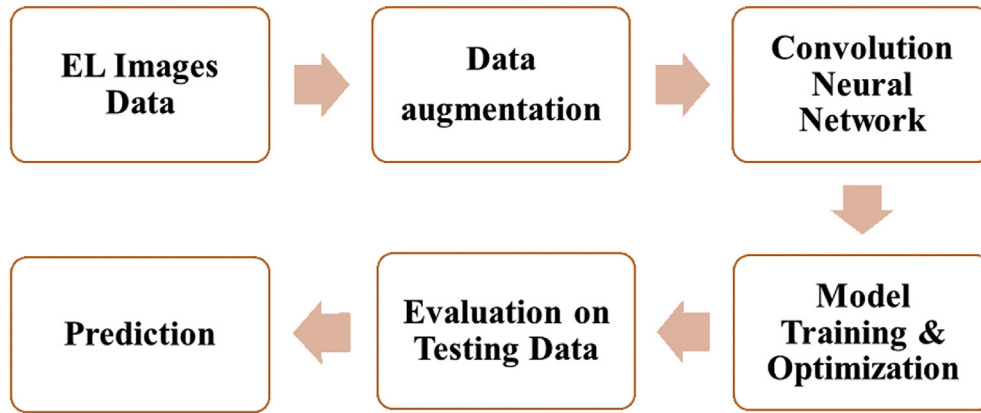


Fig. 4. Flow chart of proposed approach.

in improving model performance (discussed in section 4.2). It is noteworthy that, number of images for each class is kept equal to help the learning stability.

3.2. CNN architecture selection

For finding an optimal CNN architecture, we performed experimentation with series of architectures with different layers, number of filters, etc. In this sub-section, we discuss the step-by-step process of CNN architecture selection. During this process, experiments are performed with various structures and finally an architecture is selected for our task. The accuracy curves obtained with few of significant structures (other than the final selected structure) are given in this sub-section. At the end, the details of final selected architecture are given. While, the results obtained with final selected architecture are given later in section 4 of results.

As mentioned above, we investigated different deep learning architectures for proposing an excellent light architecture that can achieve good performance and is computationally inexpensive. We started experiments from VGG-based deep architectures [42] i.e. VGG 19, VGG 16, VGG 13 and VGG 11. These deeper networks are actually trained on a large dataset containing 1.3 million images. On the other hand, our dataset is small. Therefore, we select a relatively shallow VGG 11 structure and started reducing/adjusting its layers and parameters in a uniform manner. During this search process, we investigated various architectures. Few of the significant architectures are briefed in the remaining sub-section.

For VGG deep network structures [42] like VGG 19, VGG 16, VGG 13, and VGG 11, we obtain flat or irregular curves of accuracy and loss as our dataset is small. As VGG 11 is relatively shallower among them, we started with it and reduce its number of layers in a uniform manner. For 10 and 9 layers, we obtain the similar results as obtained by deeper networks i.e. flat or irregular curves of accuracy and loss. However, with further reduction in number of layers, the model starts learning. The VGG structure-based networks with reduced layers are [64, 128, 256, 256, 512, 512, 4096 fully connected (FC), 2 output(O)], [64, 128, 256, 256, 512, 4096(FC), 2(O)], and [64, 128, 256, 512, 4096(FC), 2(O)] represented as VGG structure-based 8, 7, and 6 layered networks respectively. ReLU as an activation function, max-pooling after each convolution layer, dropout of 0.5, and softmax classifier is used in all these networks. The filter sizes used in convolutional and pooling layers are 3×3 with stride 1 and 2×2 with stride 2 respectively. These models achieve accuracies ranging between 87 and 90% as shown in Fig. 5.

It can be observed from Fig. 5 that networks with 6 and 7 layers

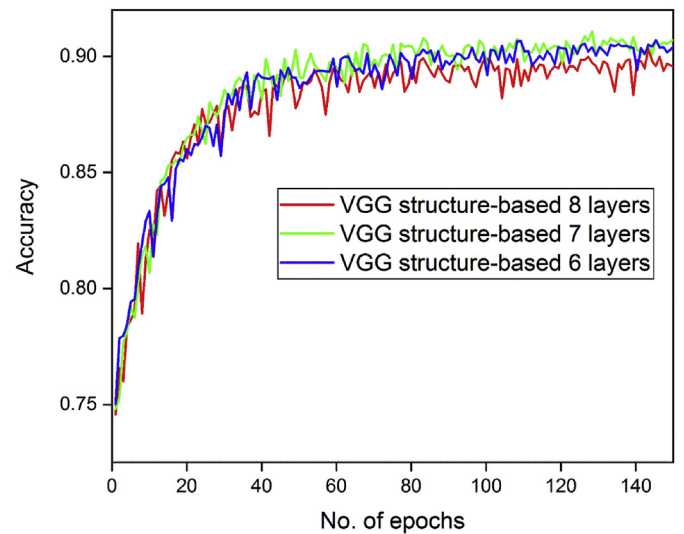


Fig. 5. Test accuracies of VGG structure-based models with reduced (8, 7 and 6) layers for public solar cell dataset (20% test set).

show almost same performance. However, to get extra computational benefits and to reduce the generalization gap, we further adjust parameters of model with 6 layers and try different regularization strategies. Few of the significant modified forms of 6-layered network are [64, 128, 128, 256, 2048(FC), 2(output)], [64, 64, 128, 128, 1024(FC), 2(output)], [32, 32, 64, 128, 512(FC), 2(output)], and [16, 32, 32, 96, 512 (FC), 2(output)] represented as A, B, C and D respectively. ReLU as an activation function, max-pooling after each convolution layer, dropout of 0.5, and softmax classifier is used in all these 6-layered networks A, B, C, and D. The filter sizes used in convolutional and pooling layers are 3×3 with stride 1 and 2×2 with stride 2 respectively. The accuracies obtained by these models are shown in Fig. 6. It can be clearly seen that by modifications, number of parameters can be reduced while maintaining performance. The reduction in redundant parameters is important in terms of less computational power and time.

The 6-layered network D is a light architecture which achieved high performance while using less parameters. However, introduction of generalization strategies in this architecture reveals that the generalization gap can be further reduced. Therefore, we further carried out series of experiments with different generalization strategies to propose a best architecture. The effect of generalization strategies adopted in our experiments is discussed

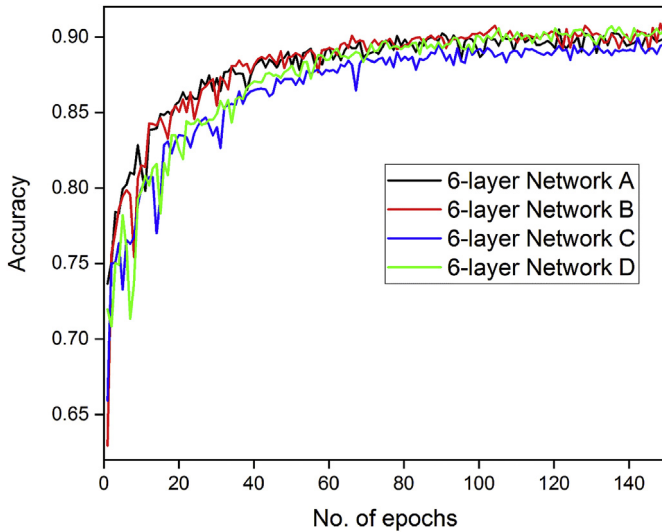


Fig. 6. Test accuracies of 6-layered A, B, C, and D models for public solar cell dataset (20% test set).

in results section. In short, after series of experiments with introducing different generalization strategies, we found that the addition of batch normalization (BN) layer and L2 weight regularization in 6-layered network D helps in reducing overfitting. After their addition, the model accuracy reaches up to 93% approximately. It is noteworthy that, the accuracy of the network with batch normalization layer and L2 weight regularization reaches up to 90% after just 10 epochs [see Fig. 7]. On the other hand, without these layers, the accuracy reaches up to 90% after 120 epochs [see Fig. 6]. For weight regularization, it was observed that L2 regularization helps in achieving relatively better performance compared to L1 [see Fig. 7]. Therefore, finally we selected L2 weight regularization for our final proposed architecture.

After extensive experimentation (as mentioned above), the final best architecture is chosen which includes batch normalization and weight regularization. The details of this architecture are given in subsequent sub-section.

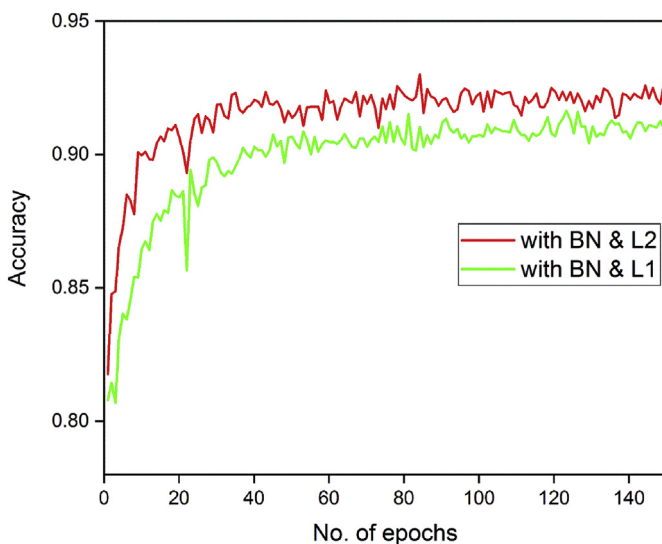


Fig. 7. Test accuracy of 6-layered network D after introduction of batch normalization (BN) and weight regularization (L1 & L2) for public solar cell dataset (20% test set).

3.2.1. Details of final proposed light CNN architecture

As mentioned in preceding sub-section, the final best architecture is proposed after extensive experimentation with a series of architectures by adjusting their layers and parameters. The details of this proposed light CNN architecture are discussed in this sub-section.

The proposed architecture consists of convolutional layers, pooling layers, batch normalization layers, rectified linear unit (ReLU) function, weight regularization, fully connected layers and loss. These units are described below.

- Input contains the pixel values of greyscale EL images. The dimensions of the image are 100*100. They are greyscale images.
- Convolutional layers consist of neurons that have learnable parameters (weights and biases). The computations performed by these layers depend upon learnable parameters as well as activation function. Based on experimentation, input image size and dataset size, we found that four convolutional layers are appropriate enough to achieve good performance. Each convolutional layer uses rectified linear unit (ReLU) activation function. L2 weight regularization is also added to each layer. The convolution process is a basic operation for neurons which is given as below [38]:

$$(k * f)[n] = \sum_{m=0}^w k[m]f[m+n] \quad (3)$$

The equation (3) defines convolution operation for one-dimensional discrete signals. Here, * shows convolution operation, f is input, k is kernel function, and w is width of kernel/filter.

- ReLU applied an activation function such as $\max(0, X) = X$, if X is an input to the neuron in convolution layer. The use of ReLU activation function prevents the vanishing gradient problem. It has no effect on the size of volume. The activation function introduces non linearity to the result of neuron convolutional with the input.
- Pooling layer is used for down sampling purpose. It down samples the output of convolution layer in spatial dimensions i.e. width and height. For example, the 2*2 max-pooling of [100*100*A] volume results in [50*50*A] volume, where A is the number of neurons/filters. We choose max-pooling layer to increase the translation invariance and avoid fitting.
- Batch normalization [39] makes model to learn optimal scale and mean of inputs for each layer. It occurs before activation function. It standardizes each feature firstly in a mini batch. A standardization procedure includes subtraction of dataset mean from each point of data and that difference obtained after subtraction is divided by standard deviation of the data. Subsequent to standardization, scaling and shifting of standardized value is carried out by using two new parameters for each layer during batch normalization process. One parameter is for scaling process and other parameter is for shifting process. In neural networks, if a weight become drastically large during model training leading to instability (exploding/vanishing gradient problem), then the batch normalization plays its role. The whole process is given in equations (4)–(7) [38].

$$\mu_B = \frac{1}{m_B} \sum_{i=1}^{m_B} X^{(i)} \quad (4)$$

$$\mu_B = \frac{1}{m_B} \sum_{i=1}^{m_B} (X^{(i)} - \mu^B)^2 \quad (5)$$

$$\hat{X}^{(i)} = \frac{X^{(i)} - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}} \quad (6)$$

$$Z^{(i)} = \gamma \cdot \hat{X}^{(i)} + \beta \quad (7)$$

In the above equations, μ_B is the mean, σ_B is standard deviation, m_B is number of instances in mini batch, $\hat{X}^{(i)}$ is input after zero centering and normalization, γ is scaling parameter, β is shifting parameter, ε is smoothing term (a very small number to prevent division by zero), $Z^{(i)}$ is output after batch normalization process that is scaled and shifted form of inputs.

- Fully connected (FC) layer: As the name shows, each neuron of FC layer is connected to all the neurons in preceding layer. In contrast to this, the convolution layers have different connection setup and they are also termed as hidden layers in literature. The dropout rate for the fully connected layer is set to 0.5. Dropout [43] causes dropping out of units randomly in neural networks. Every neuron has a p probability (here p is 50%) of being dropped out temporarily during a training step. A neuron may be ignored in one step, but it may be active in other step/steps. The neurons are only dropped during training process. When the neurons are trained with dropout technique, they do not co-adapt with the adjacent neurons and they are as useful as possible on their own. They are less sensitive to small input changes. These characteristics makes the model robust.

After passing through all these layers, the pixel values are transformed into class scores. The details of these layers and their hyper parameters are given in Table 1.

In practice, these layers are stacked in a CNN architecture. The final proposed light CNN architecture is shown in Fig. 8.

3.3. Optimization

Optimization process is known for minimizing loss or cost function to improve the model performance. The objective of optimizing model during training process is to find the weights that minimizes the loss or cost function. The neurons or filters in the layers of convolutional neural network have trainable parameters i.e. weights and biases. These neurons learn to capture image

features on the basis of learned weights during optimization process. These weights are randomly initialized during training process. The loss is computed after each training epoch and following this, the weights are updated in next epoch. Repeating this procedure for enough times will lead to weights that give minimum loss. In this way, the neural network learns.

Loss function measures divergence among predicted and correct labels. In our model, we use the cross-entropy loss function. It is given in equation (8). It is commonly used in binary classification to measure the inconsistency among two probability distributions. At the end of each epoch during model training, loss is computed on model predictions with respect to correct labels; and following this, weights are updated. With the decrease in loss function value, the robustness of model increases.

$$L = -\frac{1}{N} \sum_{i=1}^N \left(\sum_x p(x) \log q(x) \right) \quad (8)$$

where, $p(x)$ indicates the correct label of the i th sample, and $q(x)$ expresses the probability of the i th sample.

Furthermore, we use adaptive moment estimation (Adam) optimizer that computes adaptive learning rates for each parameter during weight update. In contrast to stochastic gradient descent, it computes individual learning rate for different parameters. It did computation from estimation of first two moments of gradients. It combines the functions of momentum optimization and RMSProp i.e. it keeps track of exponentially decaying average of the past gradients and squared gradients. The Adam algorithm is given below from step 1 to 5 [38].

$$m \leftarrow \beta_1 m - (1 - \beta_1) \nabla_{\theta} J(\theta) \quad (\text{step 1})$$

$$X \leftarrow \beta_2 X + (1 - \beta_2) \nabla_{\theta} J(\theta) \otimes \nabla_{\theta} J(\theta) \quad (\text{step 2})$$

$$m \leftarrow \frac{m}{1 - \beta_1^T} \quad (\text{step 3})$$

$$X \leftarrow \frac{X}{1 - \beta_2^T} \quad (\text{step 4})$$

$$\theta \leftarrow \theta + \eta m \cdot Y \sqrt{X + \varepsilon} \quad (\text{step 5})$$

Here, m is momentum vector, β_1 is momentum decay hyperparameter, η is learning rate, θ shows weights, $J(\theta)$ denotes gradient of the cost function, and β_2 is scaling decay hyperparameter. With this, we use learning rate of 0.001, decay rates of $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and epsilon = $1e-08$.

3.4. Evaluation criteria

In classification field, different model evaluation criterions are used. We have reported accuracy, precision, recall rate, and F1 score of the model (given in sub-section 4.3).

The accuracy is the ratio of correctly predicted observations to the total events [see equation (9)]. Therein, true values are correctly predicted observations.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (9)$$

where, TP represents true positive values, FP represents false positive values, TN represents true negative values and FN represents false negative values. If $\text{argmax}(P_0, P_1) = \text{argmax}(y_0, y_1)$, then result is considered to represent correctly predicted observations

Table 1
Hyper parameters of layers of final proposed light CNN architecture.

Layer type	Parameters setting
Image input	[100 × 100 × 1] scaled to the range [0,1]
Convolution	16 (3 × 3) filters with stride 1 [ReLU, L2]
Max pooling	2 × 2 filter with stride 2
Batch normalization	
Convolution	32 (3 × 3) filters with stride 1 [ReLU, L2]
Max pooling	2 × 2 filter with stride 2
Batch normalization	
Convolution	32 (3 × 3) filters with stride 1 [ReLU, L2]
Max pooling	2 × 2 filter with stride 2
Batch normalization	
Convolution	96 (3 × 3) filters with stride 1 [ReLU, L2]
Max pooling	2 × 2 filter with stride 2
Batch normalization	
Fully connected (FC)	512 [ReLU, 0.5 dropout]
Output	2 (softmax classifier)

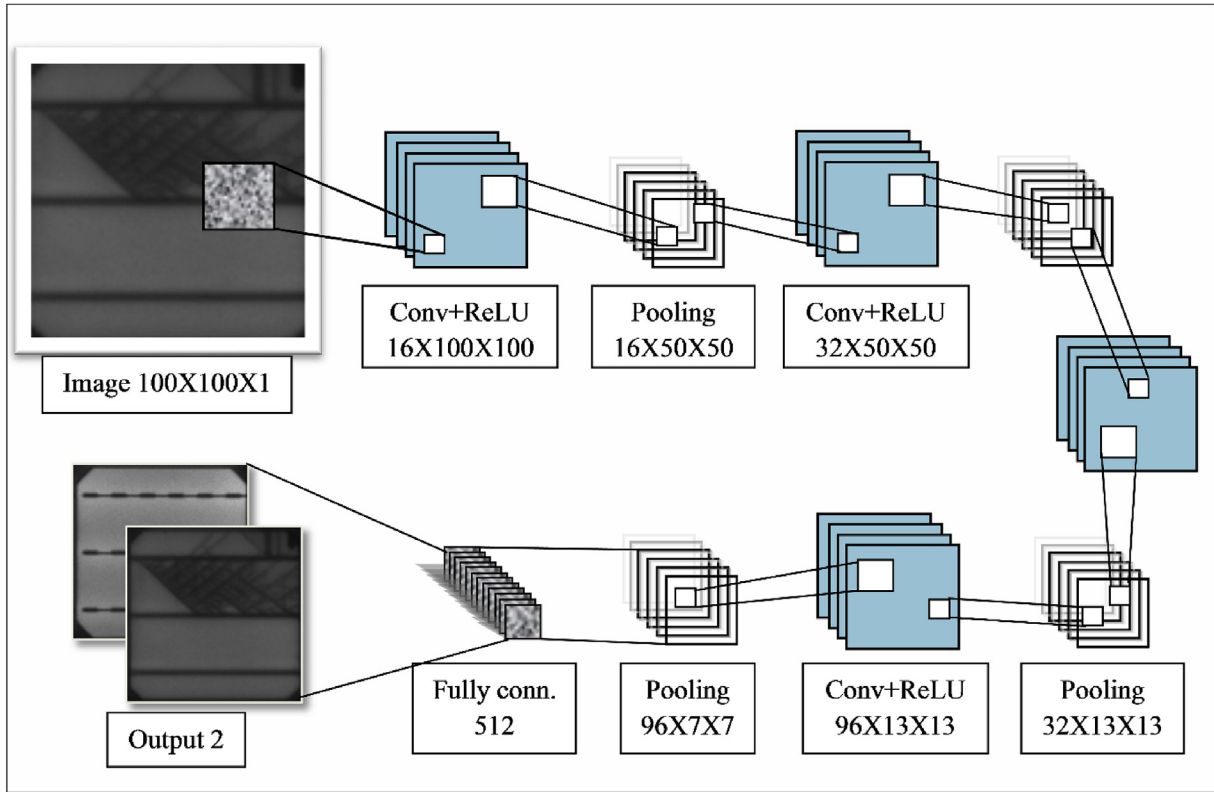


Fig. 8. Final proposed light CNN architecture.

i.e. true positive (TP) and true negative (TN) values. Otherwise, it represents false positive (FP) and false negative (FN) values. Where, P_0 denotes the probability of presence of defect in cell, P_1 denotes the probability of absence of defect in PV cell, Y_0 and Y_1 define the scores of the true labels.

Precision and Recall rate are also used when data is imbalance. Precision and Recall rate are given in equations (10) and (11) respectively. F1 score is also used commonly to show a harmonic mean of Precision and Recall rate. It is given in equation (12). In addition to these criterions, confusion matrix can be used to define the relation and the difference between classes.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (10)$$

$$\text{Recall rate} = \frac{TP}{TP + FN} \quad (11)$$

$$\text{F1 score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (12)$$

3.5. Experimental testing

The proposed automatic framework is experimentally applied. The experimental setup is shown in Fig. 9. The setup is placed on an optic table for proper alignment and stable positioning of samples and camera. Few mono-crystalline and poly-crystalline silicon PV cells are selected for testing. Current equivalent to short circuit current of cells is supplied to the solar cells using DC power supply. Therein, the leads of power supply are connected to the cell side with same polarity. The busbar side of cells is negative, and the

other side is positive. Electron-hole recombination process takes place inside the solar cells which results in emission of near infrared radiations. These radiations are captured by an electroluminescence camera which provides electroluminescence images. The optical axis of camera is adjusted perpendicular to the face of solar cell. The camera is computer controlled and the acquired images are transferred electronically from camera to the computer. Therein, images are imported to the tfl (TFlearn) model in Anaconda Spyder integrated development environment which shows real time predictions.

4. Experimental results and discussion

This section introduces the dataset and results of experiments conducted with final proposed light architecture. The strategies adopted for model generalization are also discussed with their results. Finally, the results obtained with cross validation are discussed.

The present study uses OpenCV, NumPy, TensorFlow, TFLearn, and PIL libraries in Anaconda Spyder (Python) integrated development environment. An Intel Core i5, 3.20 GHz CPU was used to perform all the experiments in Windows 7 environment.

4.1. Dataset

The dataset used in present study is a public solar cell dataset [30,31]. The other datasets used in existing studies except the study [3] are small and capture only specific defects. Furthermore, those datasets are not made public and each researcher work with different datasets leading to lack of comparison between different studies. This dataset is the first publicly available dataset of its kind which initiated the development of automatic inspection methods

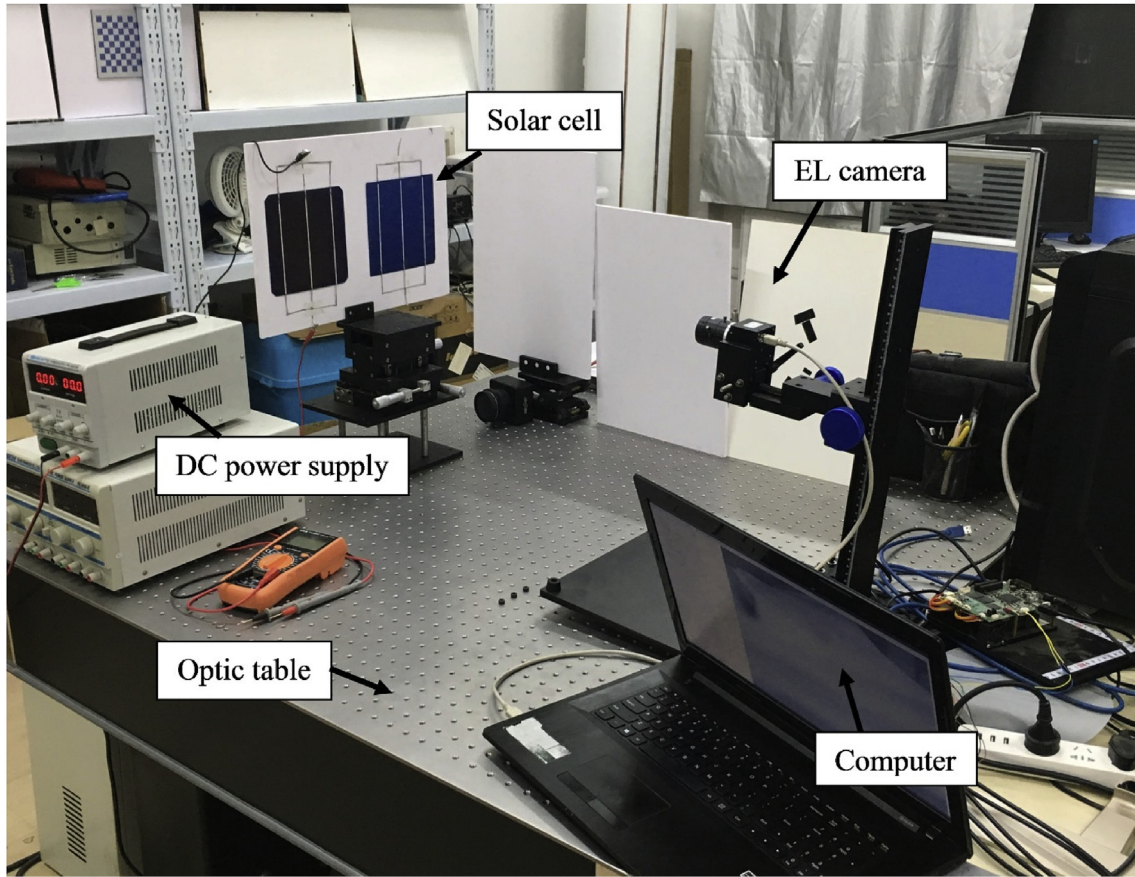


Fig. 9. Experimental setup for testing of proposed framework.

in PV field.

It consists of 2624 EL images of PV cells. These images are extracted from 44 different PV modules. They are of both polycrystalline (full square shape) and monocrystalline (pseudo Square shape) type cells. The further categories are: (1) two-busbar cells and three-busbar cells, (2) with soldering pads and without soldering pads, and (3) with three soldering pads and with six soldering pads. The samples from the dataset are shown in Figs. 2 and 3.

4.2. Generalization of proposed model

Generalization of deep learning models on small datasets is a real challenge. As the training data is of small size, the overfitting arises as a serious problem in present case. We have performed extensive experimentation to select optimal hyperparameters and regularizes for generalizing the model. The best approach to avoid overfitting is to get more training data. When getting more data is no longer possible, then we adopt other strategies. Introducing simply one or few of these strategies does not give maximum performance in our case. So, we tried series of different strategies and hyperparameters to achieve maximum performance. The strategies adopted to avoid overfitting are discussed in this section. We have followed the recommendations mentioned in Ref. [38].

4.2.1. Data augmentation

The best method to avoid overfitting is to get more training data. To get more data, data augmentation operations are applied to original images. We have carried out rotation, flipping, cropping, contrast, and blurring for data augmentation. The details of data

augmentation are given in sub-section 3.1. The new obtained images were carefully analyzed in terms of generating new samples with same labels, variation from original images, and extraction of useful information. The experiments have shown that each data augmentation operation results in enhancement of model accuracy leading to the conclusion that selected data augmentation operations are successful. These operations make the model more tolerant to orientation, position and size of defects. They reduce the difference between images and help the model to learn more representative feature subsets, which greatly improves the recognition performance.

Each data augmentation operation is analyzed with respect to its ability to improve the model performance through experiments. The testing accuracy after each data augmentation operation gets improved. The testing accuracies before and after data augmentation operations are shown in Table 2.

The data augmentation experiments showed that the model results can be significantly improved by increasing the size of training data. Thus, with the availability of large training datasets, the proposed CNN architecture can achieve high accuracies.

Table 2
Accuracy of test set before and after data augmentation (20% test set).

Sr. #	Data Augmentation operation	Before operation	After operation
1	Rotation	86.44%	91.13%
2	Flipping	86.44%	89.76%
3	Cropping	86.44%	90.41%
4	Contrast + Blur	86.44%	90.02%
5	All operations	86.44%	93%

Therefore, after the model being trained with large datasets at industrial and field level, the proposed framework can achieve even more high accuracy.

4.2.2. Model size

Reducing model size is a simplest way to get rid of overfitting. Reducing model size means reduction in number of learnable parameters. On the other side, if the learnable parameters are too less, there is not enough fitting on training data. Therefore, our goal is to achieve a balance between 'too large model size' and 'too small model size'. There is no method to determine an optimum sized model directly; instead, a series of different architectures need to be experimented.

For obtaining an excellent light architecture, we investigated a series of architectures. Finally, we propose a light six-layer (4 convolutional and 2 fully connected) architecture that achieved high accuracy and requires less computational power and time. The details are already discussed in sub-section 3.2.

4.2.3. Dropout

Dropout avoids overfitting by dropping out units randomly in neural networks [43]. In our study, a dropout of 0.5 is finally chosen for fully connected layer to obtain best results. However, it increases the training time.

4.2.4. regularization

As discussed in section 3.2, we conducted experiments with L1 and L2 regularization techniques and finally choose L2 regularization for our model.

4.2.5. Normalization layers

In our study, we perform experiments with batch normalization and local response normalization. The batch normalization layer shows relatively better performance. It also reduces training time.

4.2.6. Batch size

Batch size is also a model hyper parameter which can be tuned for model generalization purpose. Batch sizes of 32, 64 and 96 are selected for experiments. For batch size of 96, the training speed is higher than the other two values, but the accuracy is low. For batch sizes of 32 and 64, it is observed that there is no significant difference in accuracy. Therefore, we finally choose batch size of 64 as it gives high training speed.

4.3. Cross validation

We used four-fold cross validation to report our results. Cross validation is an approach to estimate model performance with less variance than a single test-train data split. Therein, data is split in to k-parts and each part or subset is called a fold; thus, it is also termed as k-fold cross validation. The model is trained on k-1 folds with one hold for testing purpose. This procedure is repeated k times in such a way that each data subset must be tested once. In our study, we use four-fold cross validation i.e. dataset is randomly divided into four subsets. Each subset contains 25% data. We end up with four different accuracy scores for four turns of training-testing process; where, three subsets are selected for training and the remaining one for testing during each turn. Finally, we report the average accuracy with standard deviation in our paper.

The model is trained for 150 epochs during each turn. Our four-fold cross validation results are 92.9%, 93.1%, 93.5% and 92.6% (mean 93.025%, standard deviation 0.37%) for first, second, third and fourth folds respectively [see Table 3]. The testing accuracy reaches to an approximately stable state with small oscillations. The training and testing accuracy curves with respect to number of

Table 3

Test accuracies (%) for four data folds.

Data folds	1st fold	2nd fold	3rd fold	4th fold	Average
Accuracy	92.9%	93.1%	93.5%	92.6%	93.025%

epochs for four data folds are shown in Fig. 10. The comparison of test accuracies for four data folds is given in Fig. 11.

4.4. Other performance measures

In above sub-section, we reported model accuracies for four dataset folds. Besides accuracy measurement, we also calculated other performance measures for the model i.e. Precision, Recall and F1 score. This sub-section discusses these performance measures. These measures are reported for random train-test data split of 80% and 20% respectively. The Precision, Recall, F1 score and accuracy of the model at 150th epoch for random 0.8–0.2 train-test data split are given in Table 4. The confusion matrix obtained at this point is shown in Fig. 12. 6.3% of the normal PV modules are misclassified as defective PV modules; while, 8.1% of the defective PV modules are misclassified as normal PV modules.

It is noteworthy that, if we train our model on just mono crystalline cells, then we achieved higher accuracy of 95.1% compared to overall efficiency. The reason is explained in following lines. The dataset used in our experiments consists of both mono crystalline and poly crystalline PV cells. The mono-crystalline cells have homogeneous surface and the defects in those cells are dissimilar from the background in EL image. Therefore, it is easy to detect defects in mono crystalline cells. On the other hand, the poly crystalline cells have large variety of textures and the defects in those cells are often similar to the background in EL image that makes it difficult to detect defects. The presence of crystallographic defects in poly crystalline cells also shown up as dark lines in the EL images.

4.5. Qualitative evaluation

Qualitative evaluation of predictions made by the proposed framework are shown in Fig. 13. It shows few correct and wrong predictions made by the developed model. Prediction results for both mono crystalline and poly crystalline PV cells are shown. In addition to the prediction made by the model, the corresponding label (true label) for each EL image is also given in the figure.

The first row in the Fig. 13 shows correct predictions by the model; while, the second row shows wrong predictions. True and predicted labels for each of EL image is present at the top of the images. The prediction results show that the poly crystalline cell images having defects similar to the background have high probability of misclassification. In case of mono crystalline cell images, the EL images with minor defects are misclassified. There are two possible reasons for these misclassification results: (1) the number of such images in the dataset is small. The increase in number of such images will make the model to learn more about these images that will lead to correct predictions, (2) such EL images of the cells are at the boundary of clearly distinguishable normal and defective cells, leading to wrong predictions. Again, the large number of training samples for such images may help.

4.6. Computation cost

The proposed architecture is a light network that requires less computational power and time. The present study is carried out on an ordinary non-GPU computer. An Intel Core i5, 3.20 GHz CPU was

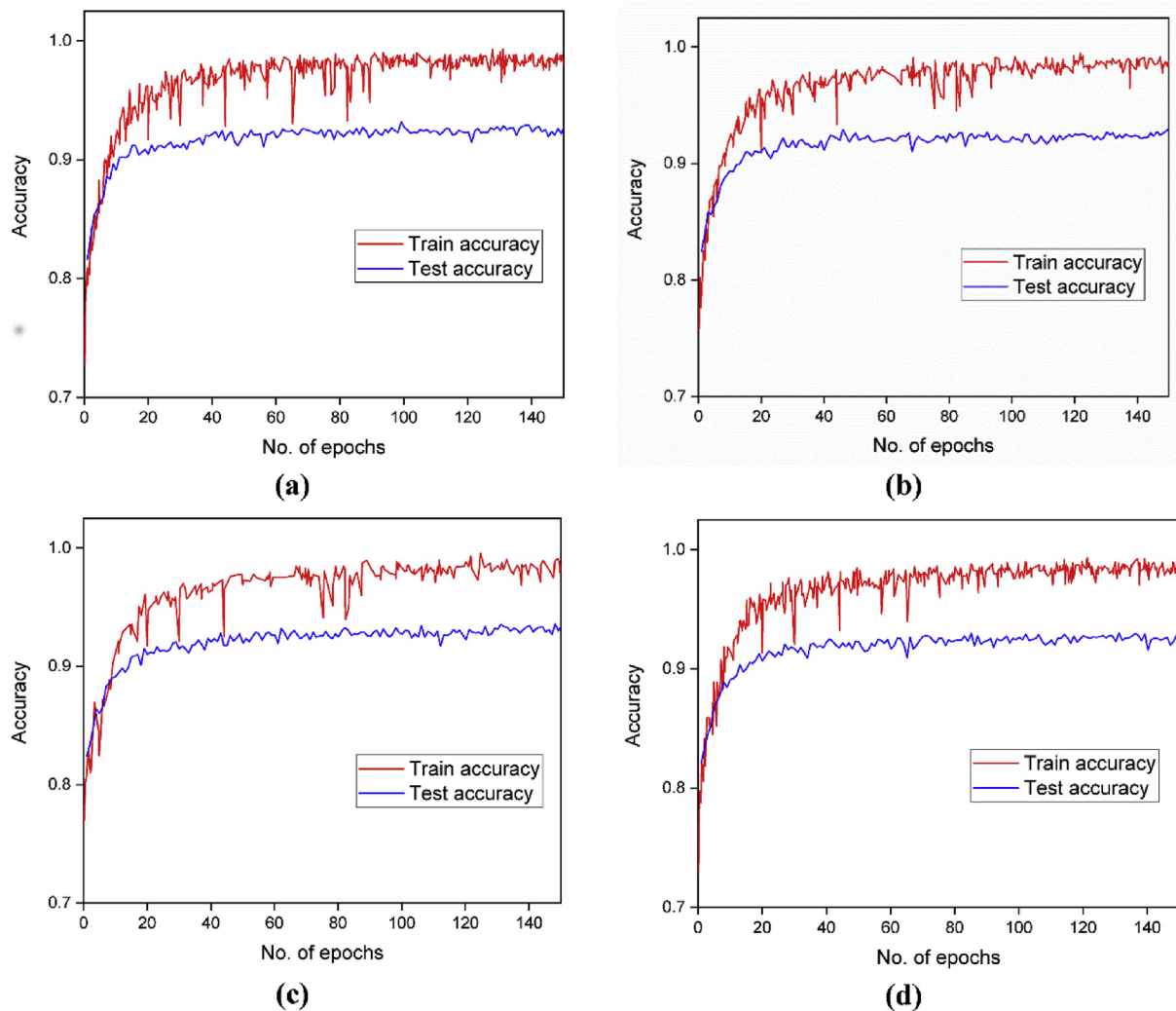


Fig. 10. Train and test accuracy curves with respect to number of epochs for the four data folds 7(a) first fold 7(b) second fold 7(c) third fold 7(d) fourth fold.

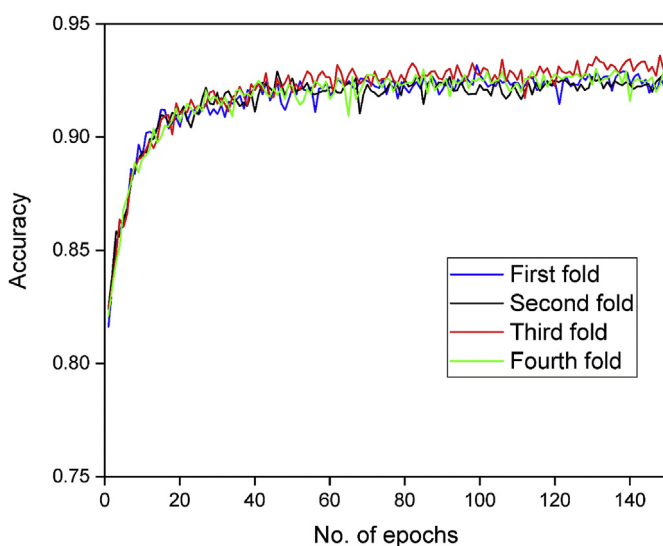


Fig. 11. Comparison of test accuracies for four data folds.

Table 4

Precision, Recall, F1 score and accuracy of the model at 150th for random 0.8–0.2 train-test data split.

Class	Precision	Recall	F 1 score
Normal PV module	0.92	0.93	0.9249
Defective PV module	0.93	0.91	0.9198
Accuracy of the developed model at 150th epoch = 92.8%			

used to perform all the experiments in Windows 7 environment. The model is trained for 150 epochs and the batch size is 64. The training time is about 13 h and 45 min. While for prediction, it requires only 8.07 s to handle 1000 images. It means that it would take 8.07 ms (ms) for predicting one image on a non-GPU ordinary computer. It maintained real-time speed.

4.7. Experimental testing results

As mentioned in sub-section 3.5, few mono-crystalline and poly-crystalline silicon PV cells are selected for experimental testing. The prediction results for these cells are shown in Fig. 14. It takes about 8–9 ms for predicting one image i.e. it maintained real

Normal PV module	93.7 %	6.3 %
	8.1 %	91.9 %
Defective PV module	Normal PV module	Defective PV module

Fig. 12. Confusion matrix for the model at 150th epoch for random 0.8–0.2 train-test split.

time speed. This framework does not require any pre-processing of acquired images to be tested. The camera is computer controlled and the acquired images are directly fed to the Anaconda Spyder integrated development environment. The proposed framework can be modified for use with infrared thermal imaging, visible red-green-blue (RGB) imaging, photoluminescence imaging, etc. and can also be extended for use with unmanned aerial vehicles (UAVs).

5. Comparison

The results obtained by existing methods on public solar cell dataset (same used in our study) are compared with our results in this section. The comparison is shown in Table 5. Deutsch S. et al. [3] used convolutional Neural Network (CNN) and SVM on public solar

cell dataset. For SVM, they obtained best results using KAZE/VGG features; and for CNN, they got best results using transfer learning. They achieved an average accuracy of 88.42% and 82.44% using CNN and SVM respectively. Our method outperforms the existing methods. We get state of the art results of mean 93.02% with standard deviation of 0.37% as compared to other existing methods. In addition, our method requires less computational power and time.

6. Conclusions and future work

We presented a novel approach using a light Convolutional Neural Network (CNN) architecture for automatic detection of photovoltaic cell defects in electroluminescence images. The proposed approach achieved state of the art results on first publicly available solar cell dataset of EL images. It achieved an average accuracy of 93.02% and can work on an ordinary non-GPU computer. It requires low computational power and time. The model is trained for 150 epochs with the batch size of 64. The training time is about 13 h and 45 min. While for prediction, it takes only 8.07 ms for predicting one image on an ordinary computer i.e. it maintained real time speed. For proposing this light architecture, we perform extensive experimentation on a series of architectures.

With the aim of coping with small data, we generalize the proposed model after extensive experimentation and found that merely adoption of a single or few generalization strategies does not give high accuracy. Rather, combination of data augmentation, appropriate model size, L2 weight regularization, batch normalization, and dropout gives best results. Furthermore, we have reviewed different kind of defects in EL images of PV cells which will help to label images manually for predicting specific defect types upon availability of new large data. To deal with data scarcity, we evaluated data augmentation operations for the EL images dataset. Selection of appropriate data augmentation strategies like rotation, flipping, cropping, contrast, and blurring is found very

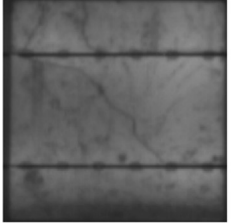
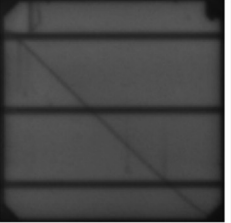
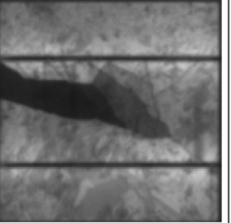
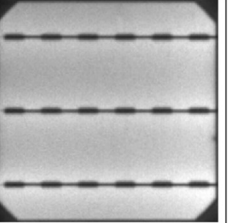
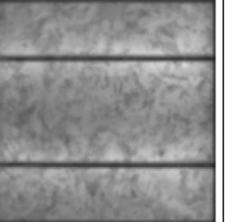
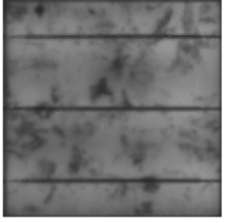
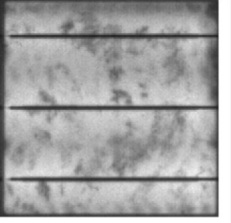
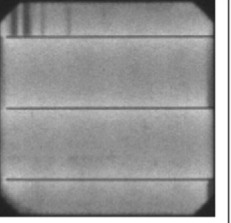
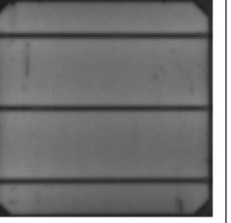
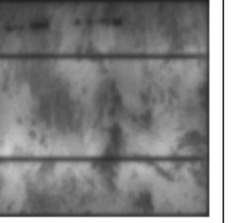
Correct predictions	True = Defective; Predicted = Defective	True = Defective; Predicted = Defective	True = Defective; Predicted = Defective	True = Normal; Predicted = Normal	True = Normal; Predicted = Normal
					
Wrong predictions	True = Defective; Predicted = Normal	True = Normal; Predicted = Defective	True = Defective; Predicted = Normal	True = Normal; Predicted = Defective	True = Defective; Predicted = Normal
					

Fig. 13. Qualitative evaluation of the predictions made by the proposed framework: first row shows few correct predicted EL images, and second row shows few wrong predicted EL images (True and predicted labels are written at the top of each image).

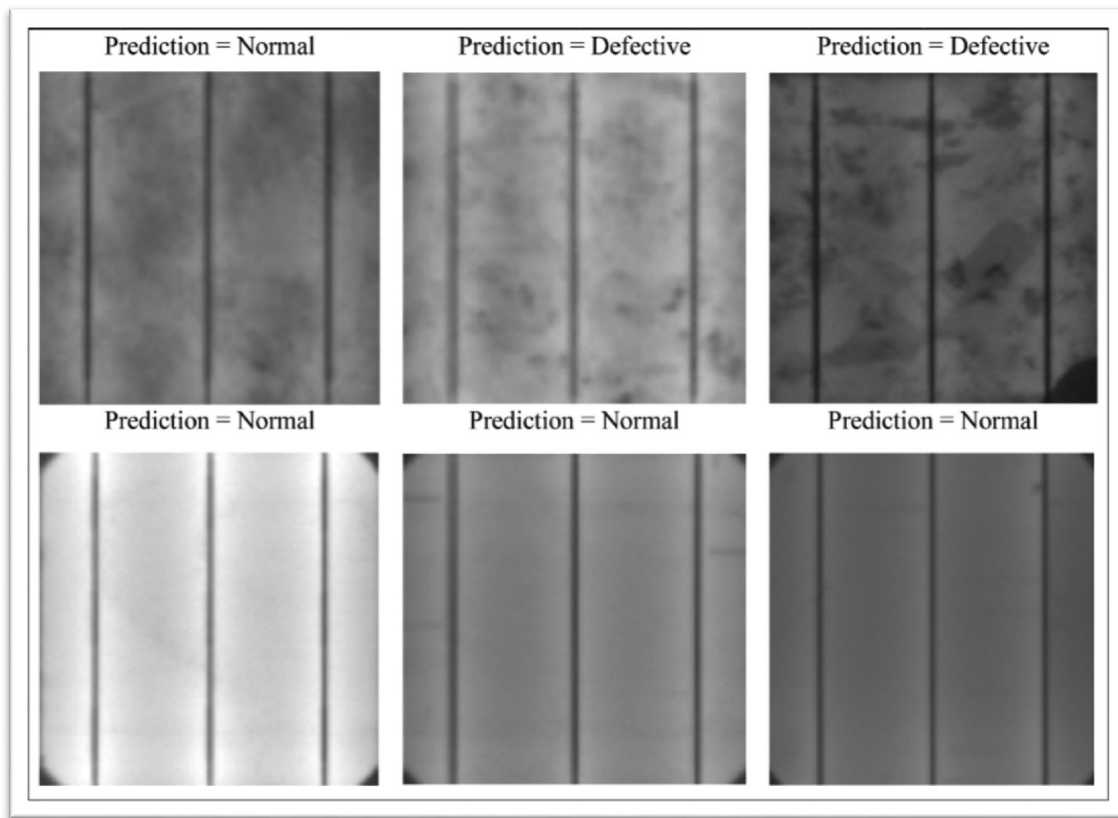


Fig. 14. Predictions of EL images of solar cells under experimental testing.

Table 5

A Comparison of our results with existing methods.

Study	Method	Dataset	Input (WXHXC)	Accuracy (%)
[3]	SVM (KAZE/VGG features)	public solar cell dataset	$300 \times 300 \times 1$	82.44%
[3]	CNN (Transfer learning)	public solar cell dataset	$300 \times 300 \times 1$	88.42%
Our model	CNN	public solar cell dataset (same as used in above two methods)	$100 \times 100 \times 1$	$93.02\% \pm 0.37\%$

important for enhancing performance. These data augmentation operations help to increase the model accuracy up to 6.5% approximately.

The proposed framework is experimentally applied in the lab and can help for automatic PV defect detection in the field and industry. The experiments showed that the results can be significantly improved by increasing the size of training data. With the availability of large datasets, the proposed CNN architecture can achieve higher accuracy. In future work, the defects can be classified into different types upon availability of large training data.

Acknowledgement

The authors would like to acknowledge Chinese Academy of Sciences, and the World Academy of Sciences for awarding CAS-TWAS President's fellowship to Ph.D. scholars. In addition, we also want to thank Dr. Claudia Buerhop-Lutz and her research team from Germany for providing first public PV EL image dataset.

References

- [1] Li G, Akram MW, Jin Y, Chen X, Zhu C, Ahmad A, et al. Thermo-mechanical behavior assessment of smart wire connected and busbar PV modules during production, transportation, and subsequent field loading stages. *Energy* 2019;168:931–45. <https://doi.org/10.1016/j.energy.2018.12.002>.
- [2] Paggi M, Berardone I, Infuso A, Corrado M. Fatigue degradation and electric recovery in Silicon solar cells embedded in photovoltaic modules. *Sci Rep* 2014;4:1–7. <https://doi.org/10.1038/srep04506>.
- [3] Deitsch S, Christlein V, Berger S, Buerhop-Lutz C, Maier A, Gallwitz F, et al. Automatic classification of defective photovoltaic module cells in electroluminescence images. 2019. arXiv preprint arXiv:1807.02894.
- [4] Tománek P, Škarvada P, MacKú R, Grmela L. Detection and localization of defects in monocrystalline silicon solar cell. *Adv Opt Technol* 2010;2010. <https://doi.org/10.1155/2010/805325>. Article ID 805325.
- [5] Osawa S, Nankano T, Mastsumoto S, Katayama N, Saka Y, Sato H. Fault diagnosis of photovoltaic modules using AC impedance spectroscopy. In: Proceedings of IEEE international conference on renewable energy research and applications. UK: ICRERA) Brimingham; 2016. p. 210–5. <https://doi.org/10.1109/ICRERA.2016.7884539>.
- [6] Schuss C, Remes K, Leppänen K, Saarela J, Fabritius T, Eichberger B, et al. Detecting defects in photovoltaic panels with the help of synchronized thermography. *IEEE Trans Instrum Meas* 2018;67:1178–86. <https://doi.org/10.1109/TIM.2018.2809078>.
- [7] He Y, Du B, Huang S. Noncontact electromagnetic induction excited infrared thermography for photovoltaic cells and modules inspection. *IEEE Trans Ind Inf* 2018;14(12):5585–93. <https://doi.org/10.1109/TII.2018.2822272>.
- [8] Akram MW, Li G, Jin Y, Chen X, Zhu C, Zhao X, et al. Improved outdoor thermography and processing of infrared images for defect detection in PV modules. *Sol Energy* 2019;190:549–60. <https://doi.org/10.1016/j.solener.2019.08.061>.
- [9] Buerhop C, Fecher FW, Pickel T, Häring A, Adamski T, Brabec CJ. Verifying defective PV - modules by IR - imaging and controlling with module

- optimizers. *Prog Photovolt Res Appl* 2018;622–30. <https://doi.org/10.1002/pip.2985>.
- [10] Ebner R, Zamini S, Újvári G. Defect analysis in different photovoltaic modules using electroluminescence (EL) and infrared (IR)-Thermography. In: *Proc. 25th European Photovoltaic Solar Energy Conference and Exhibition (25th EUPVSEC)*; 2010. p. 333–6. <https://doi.org/10.4229/25thEUPVSEC2010-1DV.2.8>.
 - [11] Jahn U, Herz M, Köntges M, Parlevliet D, Paggi M, Tsanakas I, et al. reportReview on infrared and electroluminescence imaging for PV field applications report IEA-PVPS T13-10:2018. International Energy Agency. ISBN: 978-3-906042-53-4. (accessed in December 2018).
 - [12] Buerhop C, Hohmann P, Glück B, Wirsching S, Bemm A, Pickel T, et al. Evolution of cell cracks in PV - modules under field and laboratory conditions. *Prog Photovolt Res Appl* 2018;26(4):261–72. <https://doi.org/10.1002/pip.2975>.
 - [13] Bedrich K, Bokalić M, Bliss M, Topić M, Betts TR, Gottschalg R. Electroluminescence imaging of PV devices: advanced vignetting calibration. *IEEE J Photovolt* 2018;8:1297–304. <https://doi.org/10.1109/JPHOTOV.2018.2848722>.
 - [14] Bedrich KG, Luo W, Pravettoni M, Chen D, Chen Y, Wang Z, et al. Quantitative electroluminescence imaging analysis for performance estimation of PID-influenced PV modules. *IEEE J Photovolt* 2018;8:1281–8. <https://doi.org/10.1109/JPHOTOV.2018.2846665>.
 - [15] Luo W, Khoo S, Hacke P, Naumann V, Lausch D, et al. Potential-induced degradation in photovoltaic modules: a critical review. *Energy Environ Sci* 2017;10:43–68. <https://doi.org/10.1039/C6EE02271E>.
 - [16] Islam MA, Hasanuzzaman M, Rahim NA. Investigation of the potential induced degradation of on-site aged polycrystalline PV modules operating in Malaysia. *Measurement* 2018;119:283–94. <https://doi.org/10.1016/j.measurement.2018.01.061>.
 - [17] Alsafasfeh M, Abdel-Qader I, Bazuin B, et al. Unsupervised fault detection and analysis for large photovoltaic systems using drones and machine vision. *Energies* 2018;11(9):1–18. <https://doi.org/10.3390/en11092252>. 2252.
 - [18] Demant M, Welschehold T, Nold S. Micro-cracks in silicon wafers and solar Cells: detection and rating of mechanical strength and electrical quality. In: *Proc. of 29th European PV Solar Energy Conference and Exhibition*; 2014. <https://doi.org/10.4229/EUPVSEC20142014-2B0.1.3>.
 - [19] Demant M, Welschehold T, Oswald M, Bartsch S, Brox T, Schoenfelder S. Microcracks in silicon wafers I: inline detection and implications of crack morphology on wafer strength. *IEEE J Photovolt* 2016;6:126–35. <https://doi.org/10.1109/JPHOTOV.2015.2494692>.
 - [20] Tsai D, Wu S, Li W. Defect detection of solar cells in electroluminescence images using Fourier image reconstruction. *Sol Energy Mater Sol Cells* 2012;99:250–62. <https://doi.org/10.1016/j.solmat.2011.12.007>.
 - [21] Tsai D, Wu S, Chiu W. Defect detection in solar modules using ICA basis images. *IEEE Trans Ind Inf* 2013;9(1):122–31.
 - [22] Li X, Yang Q, Chen Z, Luo X, Yan W. Visible defects detection based on UAV-based inspection in large-scale photovoltaic systems. *IET Renew Power Gener* 2017;11(10):1234–44. <https://doi.org/10.1049/iet-rpg.2017.0001>.
 - [23] Stromer D, Vetter A, Oezkan HC, Probst C, Maier A. Enhanced crack segmentation (eCS): a reference algorithm for segmenting cracks in multicrystalline silicon solar cells. *IEEE J Photovolt* 2019;1–7. <https://doi.org/10.1109/JPHOTOV.2019.2895808>.
 - [24] Li X, Yang Q, Lou Z, Yan W. Deep learning based module defect analysis for large-scale photovoltaic farms. *IEEE Trans Energy Convers* 2018;34(1):520–9. <https://doi.org/10.1109/TEC.2018.2873358>.
 - [25] Li X, Wang J, Chen Z. Intelligent fault pattern recognition of aerial photovoltaic module images based on deep learning technique. *J Syst Cybern Inf* 2018;16(2):67–71.
 - [26] Ding S, Yang Q, Member S, Li X, Yan W, Ruan W. Transfer learning based photovoltaic module defect diagnosis using aerial images. *Int Conf Power Syst Technol* 2018. <https://doi.org/10.1109/POWERCON.2018.8602188> (POWERCON) 20184245–50.
 - [27] Chen H, Pang Y, Hu Q, Liu K. Solar cell surface defect inspection based on multispectral convolutional neural network. *J Intell Manuf* 2018;1–16. <https://doi.org/10.1007/s10845-018-1458-z>.
 - [28] Demant M, Virtue P, Kovvali AS, Yu SX, Rein S. Deep learning approach to inline quality rating and mapping of multi-crystalline Si-wafers. In: *Proc. 35th European photovoltaic solar energy conference and exhibitio. 35th EU PVSEC*; 2018. p. 814–8. <https://doi.org/10.4229/35thEUPVSEC20182018-2DV.3.62>.
 - [29] Mehta S, Azad AP, Chemmengath SA, Raykar V. Deep solar eye: power loss prediction and weakly supervised soiling localization via fully convolutional networks for solar panels. *IEEE Winter Conf Appl Comput Vis* 2018; 2018. p. 333–42. <https://doi.org/10.1109/WACV.2018.00043>.
 - [30] Buerhop C, Deitsch S, Maier A, Gallwitz F, Berger S, Doll B, et al. A benchmark for visual identification of defective solar cells in electroluminescence imagery. In: *Proc. of 35th European PV Solar Energy Conference and Exhibition*; 2018. p. 1287–9. <https://doi.org/10.4229/35thEUPVSEC20182018-5CV.3.15>.
 - [31] Deitsch S, Buerhop-lutz C, Maier A, Gallwitz F, Riess C. Segmentation of photovoltaic module cells in electroluminescence images. 2018. arXiv pre-print arXiv:1806.06530.
 - [32] Mano Raja M, Mahalakshmi R, Karuppusamyandian M, Bhuvanesh A, Jai Ganesh R. Classification and detection of faults in grid connected photovoltaic system. *Int J Sci Eng Res* 2016;7(4):149–54.
 - [33] Hoyer U, Buerhop C, Jahn U. Electroluminescence and infrared imaging for quality improvements of PV modules. In: *Proceedings of 23rd EU-PVSEC* 2913–6; 2008.
 - [34] Spagnolo GS, Del Vecchio P, Makary G, Papalillo D, Martocchia A. A review of IR thermography applied to PV systems. In: *Proc 11th Int Conf Environ Electr Eng EEEIC*; 2012. p. 879–84.
 - [35] Köntges M, Kurtz S, Packard C, Jahn U, et al. reportReview of failures of photovoltaic modules report IEA-PVPS T13-01:2014. International Energy Agency. ISBN: 978-3-906042-16-9. (accessed in December 2018).
 - [36] Peter Würfel. *Physics of solar cells: from principles to new concepts*. Weinheim: Wiley-VCH; 2005. ISBN 3-527-40428-7.
 - [37] Sander M, Dietrich S, Pander M, Schweizer M, Ebert, et al. Investigations on crack development and crack growth in embedded solar cells. In: *Proceedings of SPIE Reliability of Photovoltaic Cells, Modules, Components, and Systems IV*; 2011. p. 811209. <https://doi.org/10.1117/12.893662>.
 - [38] Géron A. *Hands-on machine learning with scikit-learn and TensorFlow*. O'Reilly Media USA; 2017.
 - [39] Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. In: *Proceedings of the 32nd International Conference on Machine Learning (PMLR)*; 2015. p. 448–56.
 - [40] Chollet F. *Deep learning with Python*. New York USA: Manning Publications Co.; 2018.
 - [41] Chen J, Jin Y, Akram MW, Li K, Chen E. Novel multi-convolutional neural network fusion approach for smile recognition. *Multimed Tools Appl* 2018; 1–21. <https://doi.org/10.1007/s11042-018-6945-x>.
 - [42] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: *Proceedings of Int. Conf. Learn. Represent.*; 2015. ICLR2015.
 - [43] Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res* 2014;15:1929–58.