

Received 20 December 2023, accepted 1 January 2024, date of publication 4 January 2024,
date of current version 12 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3349975

RESEARCH ARTICLE

Photovoltaic Cell Defect Detection Based on Weakly Supervised Learning With Module-Level Annotations

HYUNGU KANG¹, JEONGMIN HONG¹, JIWON LEE^{1,2}, AND SEOKHO KANG¹

¹Department of Industrial Engineering, Sungkyunkwan University, Jangan-gu, Suwon 16419, Republic of Korea

²Shinsung CS, Bundang-gu, Seongnam 13543, Republic of Korea

Corresponding author: Seokho Kang (s.kang@skku.edu)

This work was supported in part by the Shinsung-Sungkyunkwan University (SKKU) Industrial AI Solution Center, and in part by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government through the Ministry of Science and ICT (MSIT) under Grant RS-2023-00207903.

ABSTRACT Recently, convolutional neural networks (CNNs) have proven successful in automating the detection of defective photovoltaic (PV) cells within PV modules. Existing studies have built a CNN based on fully supervised learning, which requires a training dataset consisting of PV cell images annotated according to whether the individual cells are defective. However, manually annotating all the PV cells is labor-intensive and time-consuming, leading to substantial annotation costs. In this study, we propose a weakly supervised learning method to build a CNN for cell-level defect detection in a cost-efficient manner. Our method uses a training dataset solely with module-level annotations indicating whether each PV module contains defective cells, thereby substantially reducing the required annotation costs. The CNN is trained in a weakly supervised manner such that all cells in a normal module are classified as normal and at least one cell in a defective module is classified as defective. The CNN can then be used to detect cell-level defects in new PV modules. The effectiveness of the proposed method is validated through experiments using real-world data provided by a PV module manufacturer.

INDEX TERMS Photovoltaic cell defect detection, photovoltaic module manufacturing, weakly supervised learning, convolutional neural network.

I. INTRODUCTION


Photovoltaic (PV) cells, also known as solar cells, are semiconductor devices that convert light energy into electricity through photovoltaic effects. In the PV module manufacturing, a PV module consists of several PV cells that are wired and sealed in a protective laminate. During the manufacturing process, PV modules are occasionally damaged for various reasons, such as thermo-mechanical stresses and assembly failures [1], [2]. Therefore, manufacturers inspect the fabricated PV modules for quality assurance.

Electroluminescence (EL) imaging is a standard inspection method for detecting defects in the cells in PV modules [3]. Various defects in the cells, such as micro-cracks and wiring

problems, can be identified by inspecting the EL image of a PV module, an example of which is shown in Figure 1. After the EL inspection, the engineers replace the defective cells with new cells.

Typically, experienced engineers visually inspect EL images. However, the visual inspection of numerous EL images is laborious and time-consuming. Engineers working for long periods of time can become visually fatigued, which increases the risk of inspection errors. Therefore, the automation of EL inspection has been in demand [1].

Research attempts have been made to apply machine learning to automate the inspection of defective cells in PV modules. Existing studies have built a convolutional neural network (CNN) that uses a cell image as input to predict whether it is defective [1], [2], [4], [5], [6]. In the context of fully supervised learning, a CNN is trained using a cell-level

The associate editor coordinating the review of this manuscript and approving it for publication was Mostafa M. Fouda .

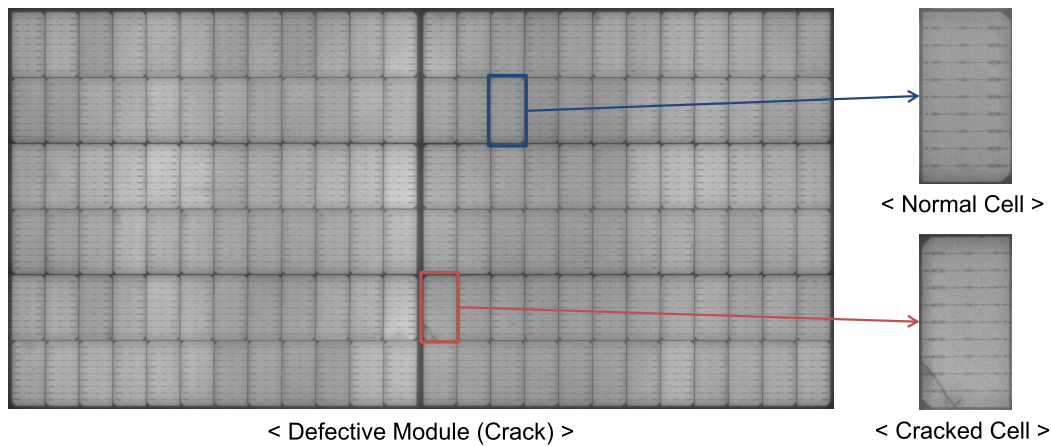


FIGURE 1. Example of EL image for PV module.

annotated dataset, where each cell is annotated with its class label, as shown in Figure 2a. To construct a training dataset of N PV modules with each module containing K cells, a total of $N \times K$ cell-level annotations are required. Although ensuring a large training dataset improves the classification accuracy of the CNN, performing the cell-level annotation task on a large scale is highly labor-intensive and time-consuming.

In this study, we propose a weakly supervised learning method to build a CNN with reduced annotation costs for PV cell defect detection. Unlike the existing studies, the proposed method uses a training dataset annotated at the module level as shown in Figure 2b, where each module is annotated to indicate whether it contains at least one defective cell. Since annotating a module can be performed more efficiently by engineers than annotating each individual cell within the module, acquiring N module-level annotations incurs substantially lower costs than acquiring $N \times K$ cell-level annotations. To train the CNN using the module-level annotated training dataset, we introduce a weakly supervised learning objective such that all cells in a normal module are classified as normal and at least one cell in a defective module is classified as defective. The effectiveness of the proposed method is verified by experiments using a real-world dataset provided by a PV module manufacturer.

The remainder of this paper is organized as follows. We review related work in section II. The proposed method is presented in section III. section IV describes the experimental settings and results. Finally, section V presents the conclusions.

II. RELATED WORK

A. AUTOMATION OF PHOTOVOLTAIC CELL DEFECT DETECTION

To automate the detection of defective PV cells, most previous studies have attempted to build a classification model based on supervised learning using a cell-level

annotated training dataset. Demant et al. [7] manually extracted handcrafted features from cell images and built a support vector machine based on the handcrafted features as a classification model to detect defective cells with micro-cracks. With advances in deep learning for computer vision tasks, recent studies have adopted CNNs as classification models [1], [5], [6]. A CNN processes a cell image without requiring manual feature extraction to predict the defect score. Deitsch et al. [2] and Ahmad et al. [4] showed that CNNs were superior to models based on manual feature extraction in detecting defective cells. Otamendi et al. [8] and Mayr et al. [9] built CNNs for the pixel-level defect detection in defective cell images.

Some studies have implemented an unsupervised learning approach using a training dataset consisting of normal cell images only in the absence of defective cell images. This approach is useful when defective cell images are unavailable during the training phase. Tsai et al. [10] and Tsai et al. [11] applied the Fourier transformation and independent component analysis, respectively, to learn the normal class structure of a training dataset. A query cell image was identified as an anomaly when the reconstruction error was high. Shou et al. [12] sequentially trained a generative adversarial network and an autoencoder network using a training dataset. They determined the anomaly score by calculating the residuals between the query image and the image produced by the generator.

The supervised learning approach generally performs better than the unsupervised learning approach, because defective cell images are explicitly used to train the classification model. However, a major challenge is the high cost of acquiring cell-level annotations that indicate whether a cell is defective. The objective of this study is to alleviate the annotation cost required to build a classification model for cell-level defect detection. We propose a method based on a weakly supervised learning approach that utilizes module-level annotations instead of cell-level annotations.

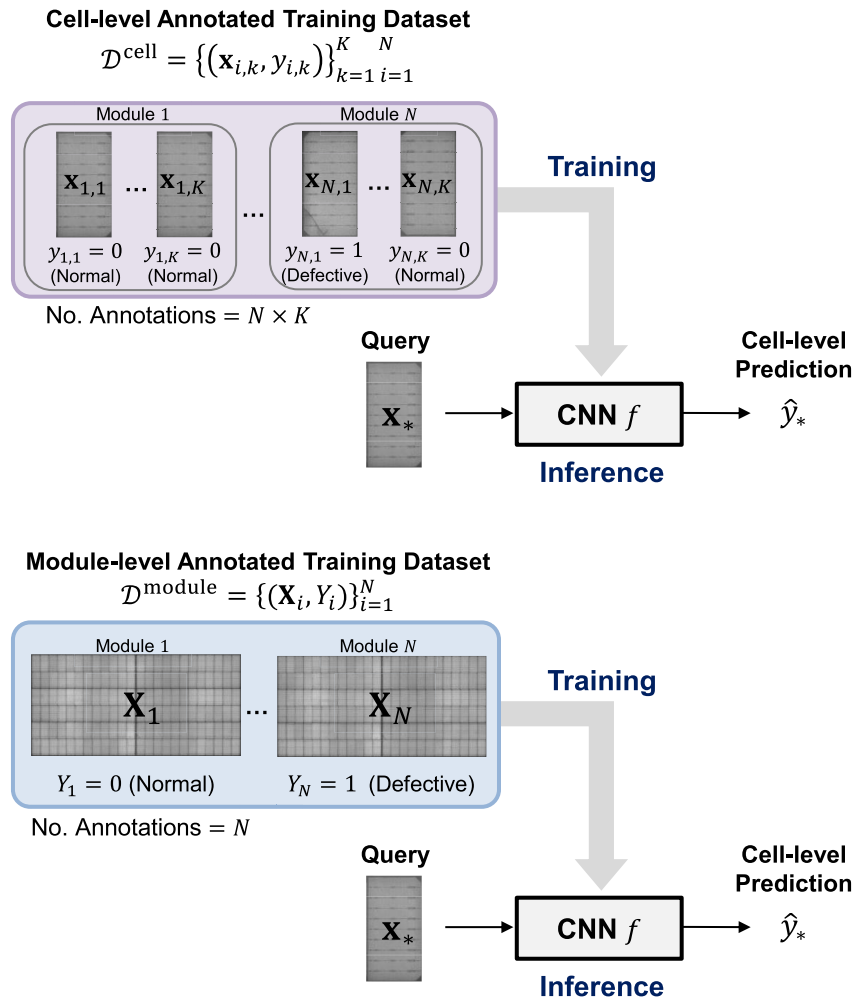


FIGURE 2. Schematic of training and inference for PV cell defect detection.

B. WEAKLY SUPERVISED LEARNING

Weakly supervised learning is an umbrella term that encompasses various attempts to build a model by learning from a training dataset under weak supervision [13]. It is practically useful when it is difficult to acquire strong supervision information for the training dataset, owing to the high annotation cost. Weak supervision can be categorized as incomplete, imprecise, and inaccurate supervision.

Incomplete supervision refers to a situation in which only some training instances are labeled, whereas others are not. Active learning [14] is used to selectively annotate unlabeled instances to improve the performance of the model. Semi-supervised learning [15] uses both labeled and unlabeled instances in a partially annotated training dataset to train the model.

Imprecise supervision refers to a situation where the training dataset contains only coarse-grained labels. The multiple-instance learning approach [16], [17] uses a set of

labeled bags, each containing many instances, as a training dataset to build a model for instance-level prediction. Each bag is annotated with a single label and the labels of individual instances are not provided.

Inaccurate supervision refers to a situation where the labels of the training instances are noisy. Noisy labels in the training dataset negatively affect the training of the model, thereby degrading its performance [18]. The preprocessing approach [19] identifies mislabeled training instances before training the model. The re-labeling approach [20], [21] iteratively re-annotates labeled instances whose labels are expected to be incorrect by the model. The robust learning approach [22] uses a model architecture, loss function, or regularization that is robust to noisy labels.

This study aims to use a module-level annotated training dataset to build a CNN for cell-level prediction, which corresponds to imprecise supervision. In the training dataset, each module consists of multiple cells and is annotated as

TABLE 1. Notations.

Symbol	Description
$\mathcal{D}^{\text{module}}$	module-level annotated dataset
$\mathcal{D}^{\text{cell}}$	cell-level annotated dataset
N	number of PV modules
K	number of PV cells per module
\mathbf{X}_i	collection of PV cells in i -th module
Y_i	class label of i -th module (0: normal, 1: defective)
$\mathbf{x}_{i,k}$	image of k -th PV cell in i -th module
$y_{i,k}$	class label of k -th PV cell in i -th module (0: normal, 1: defective)
$\hat{y}_{i,k}$	predicted defect score of k -th PV cell in i -th module
θ	defect score threshold
f	CNN ($\hat{y}_{i,k} = f(\mathbf{x}_{i,k})$)
\mathcal{L}	weakly supervised loss function

defective if at least one cell is defective and as normal otherwise. We adapt the multiple-instance learning approach by considering a module and a cell as a bag and an instance, respectively.

III. METHODOLOGY

A. OVERVIEW

We formulate the PV cell defect detection as a binary classification problem, where a classification model f takes a PV cell image \mathbf{x} as the input to predict whether the cell is defective as follows:

$$\hat{y} = f(\mathbf{x}), \tag{1}$$

where $\hat{y} \in [0, 1]$ is the predicted defect score. A high defect score signifies an increased likelihood that a given cell is defective. As the classification model f , the objective is to build a CNN by learning from a module-level annotated training dataset $\mathcal{D}^{\text{module}} = \{(\mathbf{X}_i, Y_i)\}_{i=1}^N$, where $\mathbf{X}_i = (\mathbf{x}_{i,1}, \dots, \mathbf{x}_{i,K})$ is the i -th module containing K cells and $Y_i \in \{0, 1\}$ is the corresponding class label. Each cell $\mathbf{x}_{i,k} \in \mathbb{R}^{m \times n}$ in the module is represented as a grayscale image of shape $m \times n$. The cell-level class label $y_{i,k}$ is not provided. The module-level class label Y_i takes values of 0 and 1 to indicate that the module belongs to the normal and defective classes, respectively.

Although the module-level annotated training dataset $\mathcal{D}^{\text{module}}$ is provided in the training phase, the trained CNN f is required to make cell-level predictions during the inference phase, as illustrated in Figure 2b. The supervised learning approach shown in Figure 2a, which has been used in existing studies, is not applicable in this situation because it requires cell-level annotations in the training dataset.

In this study, we introduce a weakly supervised loss function to bridge the gap in granularity between the training and inference phases. The loss function leverages information from the module-level annotations in the training dataset $\mathcal{D}^{\text{module}}$. The model f is trained in a weakly supervised manner so that all cells in a normal module are predicted to be normal, and at least one cell in a defective module is predicted to be defective. The details of the loss function are described in the following subsection. The notations used in this paper are listed in Table 1.

B. TRAINING WITH MODULE-LEVEL ANNOTATIONS

A PV module is considered normal if all associated cells are normal, whereas it is considered defective if at least one cell is defective. Accordingly, all the cells in a normal module should be classified as normal, whereas at least one cell in a defective module should be classified as defective. Owing to the absence of explicit cell-level annotations, the weakly supervised loss function is designed to identify the cell in a defective module that differs the most from normal cells. Using the loss function, the CNN f is trained to classify all cells in a normal module as normal and the identified cell in a defective module as defective.

For a training instance (\mathbf{X}_i, Y_i) , the CNN f predicts the defect scores of the individual cells $\mathbf{x}_{i,1}, \dots, \mathbf{x}_{i,K}$ as $\hat{y}_{i,1}, \dots, \hat{y}_{i,K}$. The K defect scores are compared with the module-level label Y_i . The loss function is computed as follows:

$$\mathcal{L}(\mathbf{X}_i, Y_i) = -(1 - Y_i) \sum_{k=1}^K \log(1 - \hat{y}_{i,k}) - \gamma \cdot Y_i \log(\max_l \hat{y}_{i,l}), \tag{2}$$

where γ is a hyperparameter that balances these two terms. The computational path of the loss function is illustrated in Figure 3. If the module is normal ($Y_i = 0$), then the first term is used to decrease the defect scores of all cells in the module. If the module is defective ($Y_i = 1$), then the second term is used to increase the maximum defect score among the cells in the module. As training progresses, the CNN f is encouraged to exclusively output higher defect scores for defective cells within defective modules, while suppressing the defect scores for normal cells in both normal and defective modules.

For each training iteration with a mini-batch $\mathcal{S} \subset \mathcal{D}^{\text{module}}$, the parameters of the CNN f are updated by minimizing the following objective function \mathcal{J} :

$$\mathcal{J} = \frac{1}{|\mathcal{S}|} \sum_{(\mathbf{X}_i, Y_i) \in \mathcal{S}} \mathcal{L}(\mathbf{X}_i, Y_i), \tag{3}$$

where $|\mathcal{S}|$ is the number of modules in the mini-batch.

In terms of computational cost during the training phase, the proposed method is comparable to existing methods that rely on fully supervised learning with cell-level annotations. In each training iteration, the existing methods require K forward and backward passes for each module. The proposed method requires the same for each normal module. However, it requires K forward passes and only a single backward pass for each defective module.

C. INFERENCE FOR CELL-LEVEL AND MODULE-LEVEL DEFECT DETECTION

For inference, the trained CNN f is used to detect defective PV cells. For cell-level defect detection, the inference process of the proposed method is identical to that of existing methods that rely on fully supervised learning with cell-level annotations. Given a query cell image $\mathbf{x}_{*,k}$, the defect score

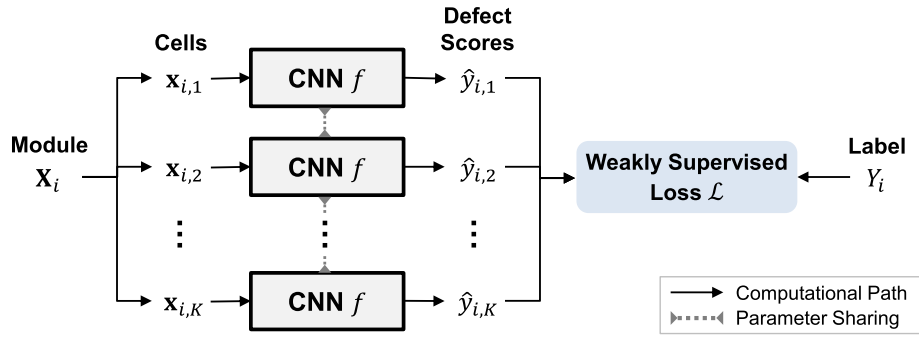


FIGURE 3. Weakly supervised loss function.

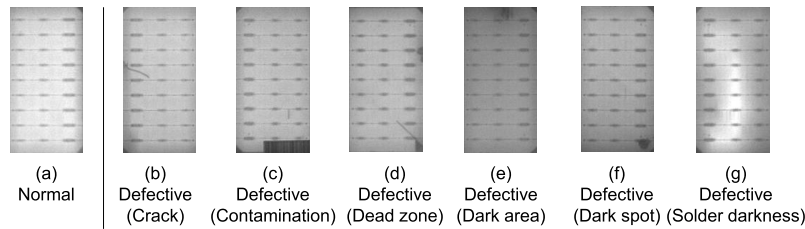


FIGURE 4. Examples of EL images for normal and defective cells.

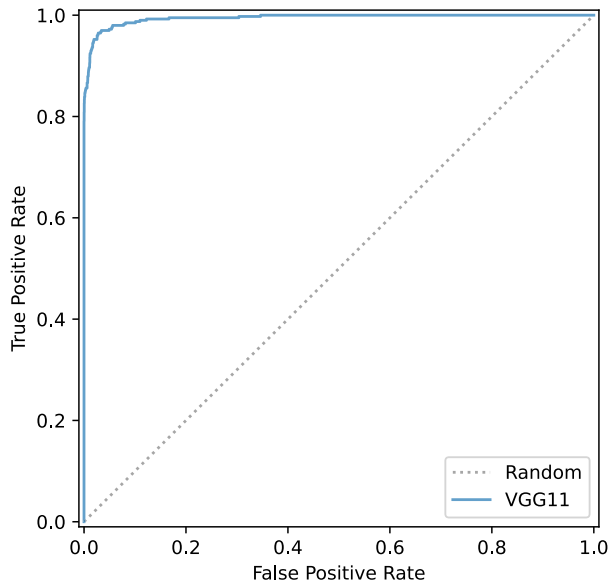


FIGURE 5. Receiver operating characteristic (ROC) curve (VGG11).

$\hat{y}_{*,k}$ can be obtained as follows:

$$\hat{y}_{*,k} = f(\mathbf{x}_{*,k}). \quad (4)$$

A cell is classified as defective if the defect score $\hat{y}_{*,k}$ is greater than a threshold θ , *i.e.*, $\hat{y}_{*,k} > \theta$.

For module-level defect detection, the defect score of the query module $\mathbf{X}_* = (\mathbf{x}_{*,1}, \dots, \mathbf{x}_{*,K})$ is calculated as the

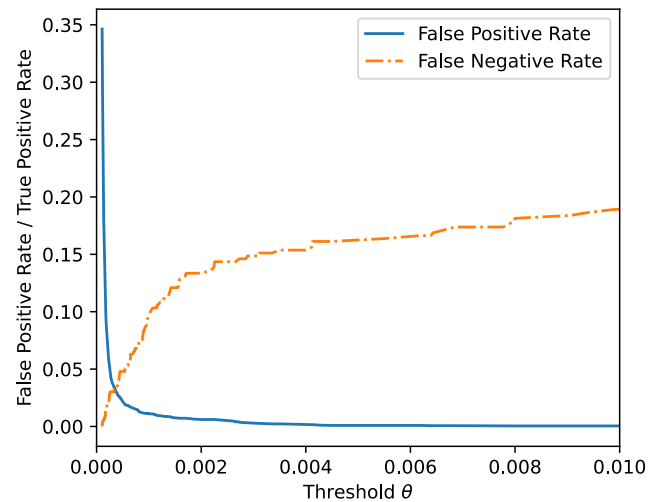


FIGURE 6. False positive rate and false negative rate at different thresholds (VGG11).

maximum defect score among the cells in the module:

$$\hat{Y}_* = \max_k f(\mathbf{x}_{*,k}). \quad (5)$$

IV. EXPERIMENTS

A. DATA ACQUISITION AND PREPROCESSING

The effectiveness of the proposed method was investigated using a real-world dataset collected from Shinsung E&G, a PV module manufacturer in South Korea, by EL imaging of the manufactured PV modules. This dataset consists of two parts with different levels of annotations by domain experts,

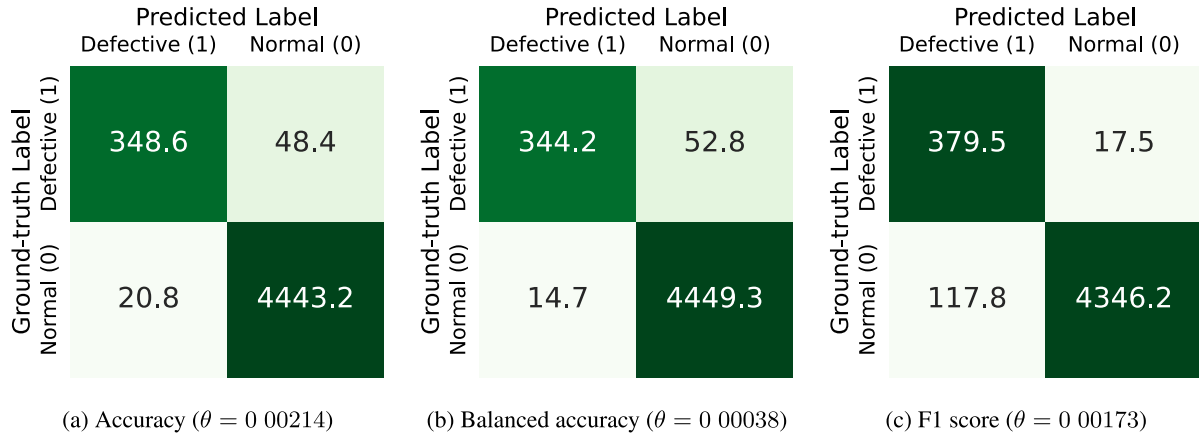


FIGURE 7. Confusion matrices with thresholds optimized for different performance measures (VGG11, average over 10 replications).

the first part with module-level annotations $\mathcal{D}^{\text{module}}$ and the second part with cell-level annotations $\mathcal{D}^{\text{cell}}$.

The first part $\mathcal{D}^{\text{module}}$ contained 1,418 module EL images with module-level annotations. The resolution of the images ranged from 5243×2676 to 5392×2702 . Each image consisted of 144 cell images (*i.e.*, $K = 144$) in 6 rows and 24 columns. Of the 1,418 images, 604 (42.6%) and 814 (57.4%) images were annotated as normal and defective, respectively. We split the EL image of each raw module into 144 individual cell images using border lines. As the borders between the cells were relatively dark, the positions of the border lines were automatically detected by finding the peaks of the horizontal and vertical means of the pixel values in the module image [6].

The second part $\mathcal{D}^{\text{cell}}$ contained 4,861 cell EL images that were separately examined by domain experts. Each image was annotated at the cell level. Among the 4,861 images, 4,464 (91.8%) and 397 (8.2%) images were annotated as normal and defective, respectively, indicating that the class distribution was imbalanced. The types of defects included crack, contamination, dead zone, dark area, dark spot, and solder darkness. The resolution of the images ranged from 84×201 to 105×213 . Figure 4 shows examples of EL images for normal and defective cells.

In the experiments, the first part of the dataset was used as the training dataset to train the classification model. The second part was used as the test dataset to evaluate the performance of the model.

Each cell image was preprocessed as follows. We resized the image to 50×100 and increased the contrast by a factor of two to emphasize the defective region. The pixel values were standardized using the mean and standard deviation of ImageNet [23].

B. EXPERIMENTAL SETTINGS

The performance of the proposed method was evaluated using different settings for the classification model. It is important to note that the existing methods are not applicable

to the problem situation addressed in this study, where a cell-level defect detection model needs to be trained using a module-level annotated training dataset. We used two common CNN architectures with different numbers of layers: VGG11, VGG13, VGG16, and VGG19 for the VGGNet family [24] and ResNet18, ResNet34, ResNet50, and ResNet101 for the ResNet family [25]. We initialized each model with ImageNet-pretrained parameters [23]. For the VGGNet models, we removed all fully-connected layers and the flatten operation, and then added a new global average pooling and a new fully-connected layer with a single output unit. For the ResNet models, the existing fully-connected layer was replaced by a new fully-connected layer with a single output unit.

In the training phase, we trained the classification model with the module-level annotated training dataset, among which 90% of the dataset was used to update the parameters and the remaining 10% was used to monitor the validation loss. The hyperparameter γ in the objective function \mathcal{J} was set to 0.01. Data augmentation was applied by randomly transforming the cell images at each training iteration based on vertical/horizontal flip ($p = 0.5$), rotation ($[-1, 1^\circ]$), vertical/horizontal shift ($[-1, 1\%]$), brightness adjustment ($[-10, 10\%]$), and contrast adjustment ($[0, 50\%]$). To update the parameters, we used the Adam optimizer with an initial learning rate of 10^{-5} and a weight decay factor of 10^{-8} . The mini-batch size was set to eight modules, with 144 cells each (*i.e.*, $144 \times 8 = 1,152$ cells). The learning rate was halved if the validation loss did not decrease over 5 consecutive epochs. The training was stopped if the validation loss did not decrease for 20 consecutive epochs or if the number of epochs reached 1,000.

In the inference phase, we evaluated the performance of the model by using the cell-level annotated test dataset. We applied test-time augmentation with the same transformation operations as in the training phase. For a query image, inference was performed 30 times with different random image transformations. The individual predicted

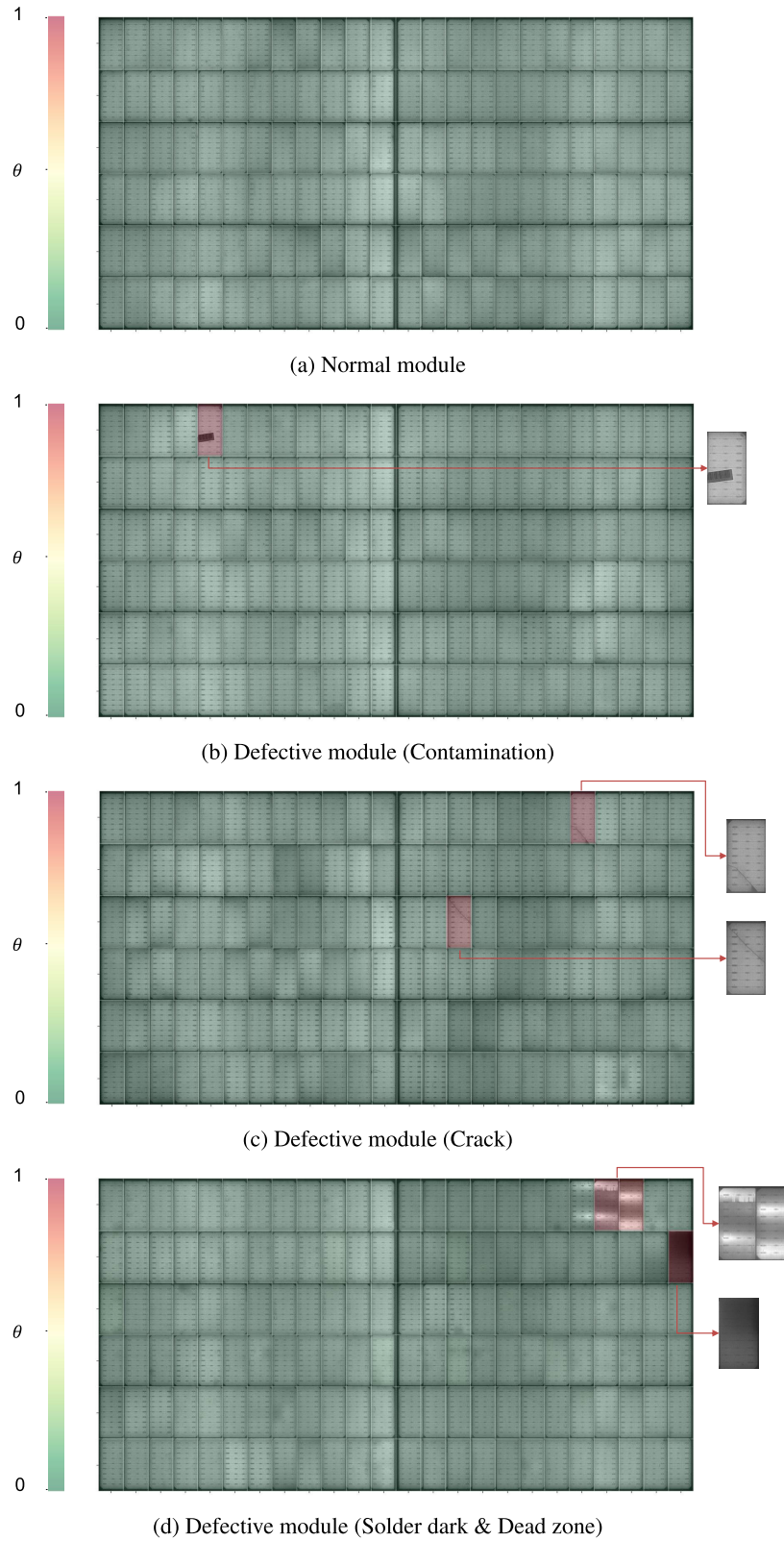


FIGURE 8. Example of heatmap visualization for cell-level predictions (VGG11, $\theta = 0.00038$).

defect scores were then averaged to obtain the final defect score.

We calculated four performance measures on the test dataset: accuracy, balanced accuracy, F1 score, and area under

TABLE 2. Comparison of classification performance (average \pm standard deviation over 10 replications).

Model	Accuracy	Balanced Accuracy	F1 Score	AUROC
VGG11	0.9861 \pm 0.0043	0.9648 \pm 0.0069	0.9109 \pm 0.0288	0.9926 \pm 0.0023
VGG13	0.9831 \pm 0.0090	0.9508 \pm 0.0269	0.8891 \pm 0.0622	0.9824 \pm 0.0123
VGG16	0.9853 \pm 0.0046	0.9583 \pm 0.0087	0.9043 \pm 0.0316	0.9856 \pm 0.0037
VGG19	0.9885 \pm 0.0020	0.9602 \pm 0.0093	0.9261 \pm 0.0136	0.9880 \pm 0.0056
ResNet18	0.9809 \pm 0.0064	0.9412 \pm 0.0176	0.8757 \pm 0.0406	0.9788 \pm 0.0109
ResNet34	0.9853 \pm 0.0099	0.9458 \pm 0.0427	0.9005 \pm 0.0810	0.9790 \pm 0.0293
ResNet50	0.9618 \pm 0.0141	0.8766 \pm 0.0419	0.6985 \pm 0.1359	0.9338 \pm 0.0350
ResNet101	0.9592 \pm 0.0248	0.8303 \pm 0.1160	0.6741 \pm 0.2211	0.8518 \pm 0.1359

the receiver operating characteristic curve (AUROC). The first three measures are dependent on the threshold θ ; thus, we individually optimized the threshold θ for each measure. The AUROC is a threshold-independent measure.

The proposed method was implemented using PyTorch in Python. The experiments were conducted with 10 independent replications using different random seeds. We report the mean and standard deviation of each result over 10 replications.

C. RESULTS AND DISCUSSION

Table 2 presents an overall comparison of the classification performance of the classification models trained using the proposed method. Most of the models achieved a considerably high classification performance with an AUROC of over 0.97, especially when the CNN architectures used were relatively small. This indicates that the proposed method successfully trained the models in a weakly supervised manner using only module-level annotations. Among the compared models, the smallest model, VGG11, yielded the highest AUROC value. On the other hand, when the architecture was larger (*e.g.*, ResNet101), we observed a tendency towards instability in training and a deterioration in classification performance, thereby resulting in a lower average and a higher standard deviation of the performance measures over the 10 replications.

We further analyzed the classification results for VGG11, which yielded the highest AUROC value. Figure 5 presents the ROC curve of VGG11, where the true positive rate against the false positive rate is plotted at different threshold settings. Figure 6 illustrates the trade-off between false positive rate (FPR) and false negative rate (FNR) at various thresholds. Increasing the threshold θ leads to a lower FPR but a higher FNR, whereas decreasing θ leads to a lower FNR but a higher FPR. This suggests that the threshold θ can be optimized by taking into account the different costs associated with different types of misclassification errors. For example, we can decrease the threshold θ if false negatives (*i.e.*, classifying defective cells as normal) should be avoided. The threshold θ can also be optimized based on a specific performance criterion. Figure 7 shows the confusion matrices of VGG11 for various threshold values, each optimized based on different performance measures. The results showed that the optimal thresholds differed depending on the specified performance measure. The use of accuracy and balanced

accuracy resulted in the highest and lowest thresholds, respectively.

Figure 8 presents the heatmap visualization of the cell-level predictions of the PV module examples for VGG11. One was a normal module and the other three were defective modules containing different types of defective cells. The cells predicted as normal and defective were shaded green and red, respectively. When a module contained more than one defective cell, the proposed method successfully detected these cells by assigning high defect scores, as shown in Figure 8c and Figure 8d. In these examples, all cells in the normal module had low defect scores, whereas the cells with high defect scores in the defective modules were identified as defective by domain experts.

V. CONCLUSION

In this study, we proposed a weakly supervised learning method to build a CNN to detect PV cell defects without cell-level annotations. By introducing a weakly supervised loss function, the CNN was trained using a module-level annotated training dataset such that the most defective cell in a defective module was predicted as defective and all cells in a normal module were predicted as normal. The CNN successfully detected defective cell images in a real-world dataset.

While previous studies relied on supervised learning with costly cell-level annotations, this study pioneers the application of weakly supervised learning with module-level annotations to build a CNN for cell-level prediction. Although a CNN trained with cell-level annotations can provide better classification performance, the proposed method is useful in practical scenarios where reducing the annotation cost for constructing the training dataset is crucial. We expect that the proposed method will lead to cost-efficient applications in the automatic PV cell defect detection. In the future, we plan to extend the proposed method to multi-class or multi-label classifications, where each cell should be classified according to its specific defect categories.

REFERENCES

- [1] M. W. Akram, G. Li, Y. Jin, X. Chen, C. Zhu, X. Zhao, A. Khaliq, M. Faheem, and A. Ahmad, "CNN based automatic detection of photovoltaic cell defects in electroluminescence images," *Energy*, vol. 189, Dec. 2019, Art. no. 116319.
- [2] S. Deitsch, V. Christlein, S. Berger, C. Buerhop-Lutz, A. Maier, F. Gallwitz, and C. Riess, "Automatic classification of defective photovoltaic module cells in electroluminescence images," *Sol. Energy*, vol. 185, pp. 455–468, Jun. 2019.

- [3] A. M. Karimi, J. S. Fada, M. A. Hossain, S. Yang, T. J. Peshek, J. L. Braid, and R. H. French, "Automated pipeline for photovoltaic module electroluminescence image processing and degradation feature classification," *IEEE J. Photovolt.*, vol. 9, no. 5, pp. 1324–1335, Sep. 2019.
- [4] A. Ahmad, Y. Jin, C. Zhu, I. Javed, A. Maqsood, and M. W. Akram, "Photovoltaic cell defect classification using convolutional neural network and support vector machine," *IET Renew. Power Gener.*, vol. 14, no. 14, pp. 2693–2702, Oct. 2020.
- [5] S.-D. Lu, M.-H. Wang, S.-E. Wei, H.-D. Liu, and C.-C. Wu, "Photovoltaic module fault detection based on a convolutional neural network," *Processes*, vol. 9, no. 9, p. 1635, Sep. 2021.
- [6] X. Chen, T. Karin, and A. Jain, "Automated defect identification in electroluminescence images of solar modules," *Sol. Energy*, vol. 242, pp. 20–29, Aug. 2022.
- [7] M. Demant, T. Welschehold, M. Oswald, S. Bartsch, T. Brox, S. Schoenfelder, and S. Rein, "Microcracks in silicon wafers I: Inline detection and implications of crack morphology on wafer strength," *IEEE J. Photovolt.*, vol. 6, no. 1, pp. 126–135, Jan. 2016.
- [8] U. Otamendi, I. Martinez, M. Quartulli, I. G. Olaizola, E. Viles, and W. Cambarau, "Segmentation of cell-level anomalies in electroluminescence images of photovoltaic modules," *Sol. Energy*, vol. 220, pp. 914–926, May 2021.
- [9] M. Mayr, M. Hoffmann, A. Maier, and V. Christlein, "Weakly supervised segmentation of cracks on solar cells using normalized L_p norm," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 1885–1889.
- [10] D.-M. Tsai, S.-C. Wu, and W.-C. Li, "Defect detection of solar cells in electroluminescence images using Fourier image reconstruction," *Sol. Energy Mater. Sol. Cells*, vol. 99, pp. 250–262, Apr. 2012.
- [11] D.-M. Tsai, S.-C. Wu, and W.-Y. Chiu, "Defect detection in solar modules using ICA basis images," *IEEE Trans. Ind. Informat.*, vol. 9, no. 1, pp. 122–131, Feb. 2013.
- [12] C. Shou, L. Hong, W. Ding, Q. Shen, W. Zhou, Y. Jiang, and C. Zhao, "Defect detection with generative adversarial networks for electroluminescence images of solar cells," in *Proc. 35th Youth Academic Annu. Conf. Chin. Assoc. Autom. (YAC)*, Oct. 2020, pp. 312–317.
- [13] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *Nat. Sci. Rev.*, vol. 5, no. 1, pp. 44–53, Jan. 2018.
- [14] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B. B. Gupta, X. Chen, and X. Wang, "A survey of deep active learning," *ACM Comput. Surv.*, vol. 54, no. 9, p. 180, 2021.
- [15] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Mach. Learn.*, vol. 109, no. 2, pp. 373–440, 2020.
- [16] M.-A. Carbonneau, V. Cheplygina, E. Granger, and G. Gagnon, "Multiple instance learning: A survey of problem characteristics and applications," *Pattern Recognit.*, vol. 77, pp. 329–353, May 2018.
- [17] C. Mera, M. Orozco-Alzate, J. Branch, and D. Mery, "Automatic visual inspection: An approach with multi-instance learning," *Comput. Ind.*, vol. 83, pp. 46–54, Dec. 2016.
- [18] G. Algan and I. Ulusoy, "Image classification with deep learning in the presence of noisy labels: A survey," *Knowl.-Based Syst.*, vol. 215, Mar. 2021, Art. no. 106771.
- [19] S. Gupta and A. Gupta, "Dealing with noise problem in machine learning data-sets: A systematic review," *Procedia Comput. Sci.*, vol. 161, pp. 466–474, Jan. 2019.
- [20] C. H. Lin, M. Mausam, and D. S. Weld, "Re-active learning: Active learning with relabeling," in *Proc. AAAI Conf. Artif. Intell.*, 2016.
- [21] M. Bernhardt, D. C. Castro, R. Tanno, A. Schwaighofer, K. C. Tezcan, M. Monteiro, S. Bannur, M. P. Lungren, A. Nori, B. Glocker, J. Alvarez-Valle, and O. Oktay, "Active label cleaning for improved dataset quality under resource constraints," *Nature Commun.*, vol. 13, no. 1, p. 1161, Mar. 2022.
- [22] H. Song, M. Kim, D. Park, Y. Shin, and J.-G. Lee, "Learning from noisy labels with deep neural networks: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 11, pp. 8135–8153, Nov. 2023.
- [23] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
- [24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015.
- [25] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 630–645.



HYUNGU KANG received the B.S. degree in systems management engineering and the M.S. degree in industrial engineering from Sungkyunkwan University, in 2020 and 2022, respectively, where he is currently pursuing the Ph.D. degree. His research interests include deep learning-based machine vision, uncertainty-aware learning, and industrial artificial intelligence.



JEONGMIN HONG received the B.S. degree in systems management engineering and the M.S. degree in industrial engineering from Sungkyunkwan University, in 2021 and 2023, respectively. She is currently a Research Engineer with SK Hynix Inc. Her research interests include anomaly detection, representation learning, and their application to industrial artificial intelligence.



JIWON LEE received the B.S. degree in computer engineering from Dongseo University, in 2007. He is currently pursuing the M.S. degree in industrial engineering with Sungkyunkwan University. He is a Software Engineer with Shinsung CS. His research interests include deep learning and image classification.



SEOKHO KANG received the B.S. and Ph.D. degrees in industrial engineering from Seoul National University, in 2011 and 2015, respectively. He is currently an Associate Professor of industrial engineering with Sungkyunkwan University. His research interests include bridging the gap between theory and practice in machine learning through practical methodologies and data mining applications in various industrial fields, including manufacturing, materials, and healthcare.

...