Journal of Network Intelligence
Taiwan Ubiquitous Information

Volume **9**, Number **4**, November **2024**

Domain-adaptive Solar Cell Surface Defect Detection Algorithm

Xiong Zhang*, Ran-Ran Li, Ting Hou, Hong Shangguan,
Xiao-Jia Wu, Ai-Ping Ning, An-Hong Wang

School of electronic information engineering
Taiyuan University of Science and Technology, Taiyuan, 030024, China
zx@tyust.edu.cn, s202115110164@stu.tyust.edu.cn, 896456637@qq.com,
shangguan_hong@tyust.edu.cn, 2006052@tyust.edu.cn,
2005074@tyust.edu.cn, ahwang@tyust.edu.cn

*Corresponding author: Xiong Zhang

Received December 11, 2023, revised March 9, 2024, accepted July 25, 2024.

Abstract. *Domain-adaptive networks have demonstrated outstanding performance in bridging interdomain disparities, emerging as a burgeoning research direction in the field of object detection in recent years. However, the substantial disparities between the distribution of solar cell defect detection data in real-world scenarios and that of training datasets significantly impact the accuracy and robustness of detection networks. To address this issue, this paper first constructs three sets of datasets, including "source domain-exposure domain dataset," " source domain-night domain dataset," and "source domain dataset-noise domain dataset." Secondly, an uncertainty-aware module is introduced into the feature extraction module of the baseline domain-adaptive multi-level entropy attention alignment (MEAA) model to enhance the network's efficiency in utilizing features during interdomain image-level alignment. Lastly, an instance-level efficient region proposal network (RPN) alignment module is designed to improve the network's instance-level prediction accuracy. Experimental results demonstrate that the proposed method significantly enhances domain adaptive detection accuracy across different domains and exhibits notable superiority compared to state-of-the-art domain adaptive detection networks.*

Keywords: Defect detection, Interdomain differences, Domain-adaptive networks,Image-level alignment, Instance-level alignment

1. Introduction.

Errors during the production process or manual operation often lead to cracks, broken cells, unsoldered cells, and other defects on the surface of solar cells, which are the main factors affecting their efficiency and lifetime. Therefore, it is of great theoretical significance and practical value to study the surface defect detection of solar cell modules in depth, particularly automatic classification and detection technology [1,2]. Early research on solar cell surface defect detection algorithms primarily included manual visual [3–5] and machine vision methods [6–10]. The class of manual visual methods is susceptible to various uncertainty factors arising from problems, such as low detection efficiency, high error rate, and high labor intensity [11]. Machine vision based methods achieve automatic detection of solar cell surface defects using video image processing technology [12], including traditional image domain analysis methods (e.g., gradient feature [13, 14], matrix decomposition [15] and clustering [16]), and transform domain analysis methods (e.g.,

the Fourier transform [17] and the wavelet transform [18,19]). These methods extract features by manually designed filters and are thus applicable to simple scenarios; however, their limited ability to capture high-level semantic features leads to poor performance when the defect features are not obvious or the defect appearance is very similar to the background. Recently, with the development of deep learning, convolutional neural networks (CNNs) [20] have been widely used for defect detection. In 2021, Zhang et al. [21] proposed a fused multichannel CNN for solar cell surface defect detection and achieved good detection performance and position regression accuracy. In 2022, Sharma et al. [22] proposed a two-stage approach for the classification and detection of steel plate surface defects. In 2024, Cao [23] improved YOLOv8-GD deep learning model for defect detection in electroluminescence images of solar photovoltaic modules. These deep learning-based defect detection methods dominantly use fully supervised learning, which offers the advantages of good real-time detection, high detection accuracy, and convenient operation. Because a significant amount of labeled data must be trained to improve the network performance, the prerequisite for adopting these algorithms is to produce several labeled defect datasets, which is extremely labor-intensive, material-intensive, and time-consuming and is thus difficult to achieve. Inspired by domain adaptive class methods, scholars have begun to investigate feature extraction from labeled source domain data to transfer them to unlabeled target domain data. The idea of domain adaptive object detection (DA-OD) first appeared in relation to the DA-OD-Faster R-CNN [24] network, which uses Faster R-CNN [25] as the backbone network to migrate the inter-domain invariant information learned from the source domain to the target domain by performing adversarial learning between the supervised source domain and unsupervised target domain datasets, aligning the inter-domain image-level and instance-level features to accomplish adaptive detection of the target domain datasets. In 2020, Chen et al. [26] proposed a hierarchical transferable calibration network HTCN to improve the transferability and discriminability of domain adaptive networks. Subsequently, methods, such as strong-weak distribution alignment (SWDA) [27], multi-adversarial faster-R-CNN (MAF) [28] and selective cross-domain alignment (SCDA) [29] networks, were developed to explore feature alignment schemes from different perspectives. A typical representative method is the DA-OD-multi-level entropy attention alignment-(MEAA) [30] algorithm, which exhibits good performance in target detection through an image-level feature alignment module based on local uncertainty attentional alignment (LUAA) and multi-level uncertainty-aware contextual alignment (MUCA). The research goal of the "Solar Cell Surface Defect Detection" project is to detect three types of defects on the surface of solar cell images and to accurately classify and locate the different defects within bounding boxes. The aforementioned adaptive detection algorithm model is bound to encounter some challenges if directly applied to this study. On the one hand, the actual solar cell image acquisition scene is subject to overexposure at noon, darkness at night, dust and gravel in the shooting environment, and so on, whereas the existing dataset is almost entirely composed of solar cell surface defect images with the same parameters for the same scene, causing the data distributions in the real scene and training dataset to have certain differences. In addition, there is a large domain gap between the source-domain data from the training dataset and the target-domain data of the real scene, necessitating the algorithm to perform interdomain image-level feature alignment characterized by low feature-utilization efficiency, poor detection accuracy, and poor algorithm robustness. On the other hand, the surface defects of solar cells reflect the characteristics of different morphologies with similarity to the background, exacerbating recognition and making it difficult to retain the original features intact when transferring them between different domains. In this context, using the DA-OD-MEAA model as the baseline network, this study proposes a solar

cell surface defect detection algorithm for domain adaptation. The specific contributions are summarized as follows:

(1) In feature extraction, information entropy is used to measure the uncertainty of different local regions of the solar cell surface defect image, whereby the calculated probability image is used as the uncertainty-aware attention feature map and fused with the feature map obtained by applying the basic feature extraction module, to obtain the multiscale weighted feature map, which can improve the feature utilization efficiency of the network.

(2) An instance-level efficient channel-attention RPN alignment module is designed to improve the instance-level prediction accuracy of the network. Specifically, local cross-channel interaction without dimensionality reduction and a one-dimensional convolutional kernel adaptive selection strategy are used in RPN. Local cross-channel interaction can realize weight sharing without increasing the number of network parameters and operations, whereas, one-dimensional convolutional kernel adaptive selection can preserve the integrity of the original features by avoiding data dimensionality reduction during computation.

2. **Methods.**

The overall framework of the proposed domain-adaptive solar-cell surface-defect-detection network, shown in Figure 1, mainly consists of an image-level feature alignment (ILFA) module and an instance-level efficient channel attention RPN (ILECA-RPN) alignment module. The working principle can be briefly described as follows. A solar cell image randomly selected from the source domain and another selected from target domain are fed into the backbone network. The input image pair passes through three feature extractors: G, F_1, and F_2, where F_1, F_2 and their lower paths jointly form the ILFA module. The feature extractor G contains two $3 * 3$ convolutional layers and a maxpooling layer.The feature map output of F_1 is fed into the ILFA module, whereby the feature maps obtained after F_1 and F_2 processing are sent to the discriminators D_1 and D_2. A pixel-by-pixel information entropy calculation is then performed on the output of the discriminators to obtain uncertainty-aware attention feature maps, which are then multiplied by the feature map outputs from F_1 and F_2; the result is further subjected to cross-layer cascading to improve the network utilization of local features. Because RPN networks have a strong dependency on channel mechanisms, an efficient channel attention (ECA) [31] module was introduced to help the classification and regression branches of the RPN network make more accurate prejudgments and achieve feature alignment at the instance level.

2.1. **Image-level feature alignment (ILFA) module.**

In the solar-cell surface defect dataset, the local areas of each image may contain different target defects or parts of the same target defect, which usually consist of a certain number of pixels with different intensities. In different domains, the local areas where the target defects are distributed in the solar-cell images display uncertainties, exhibiting differences in discriminability and transferability in these areas. The structure and texture features of the same class of target defects in two solar cell images in different domains are similar, and the structure and texture features of different classes of target defects are different. Thus, it can be concluded that the structure and texture features of local regions of solar cell images can have an important impact on the effectiveness of domain-invariant feature extraction in unsupervised domain adaptive tasks. As the morphological defects of solar cell surfaces are subtle and difficult to detect, the ability to extract high-level semantic information while preserving low-level structural and textural

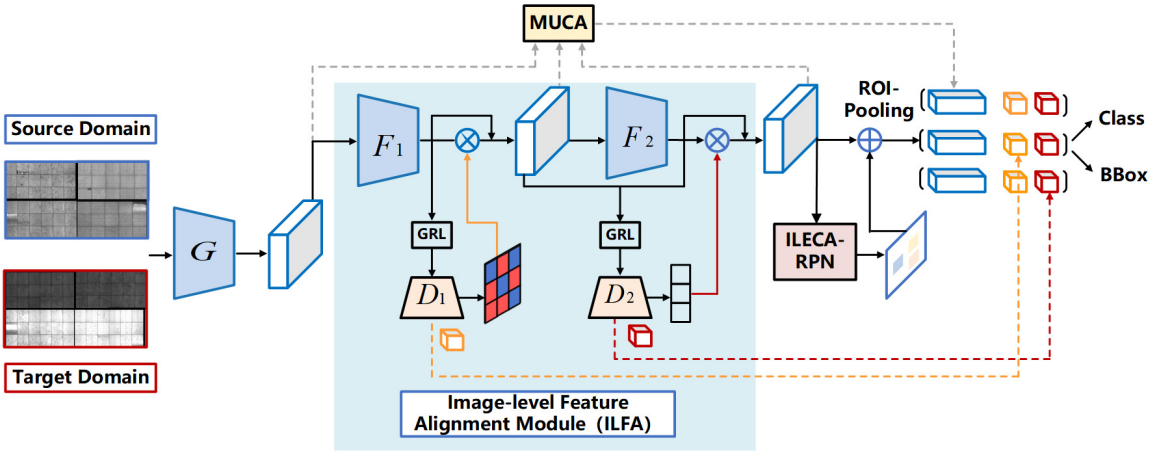


FIGURE 1. An overview of our framework composed of two proposed modules, an image-level feature alignment module (ILFA), and an instance-level efficient channel attention RPN alignment module (ILECA-RPN).

information is crucial. Considering information entropy as a measure of uncertainty, this method adopts information entropy to determine the uncertainty of target defects in solar cell images in different local regions and uses the generated uncertainty-aware attention feature maps to weight the original feature maps, thereby extracting the domain-invariant features from different domains to perform feature alignment between source and target domains considering pixel semantics. The solar cell surface defect image is fed into the domain-adaptive detection network, and the primary feature map, processed by feature extraction module G, is the input into F_1. The output f_1 of F_1 is then fed into the domain discriminator D_1 which outputs d_1.

$$d_1 = D_1(f_1) \tag{1}$$

If the uncertainty-aware attention feature map is represented by s_i^1, then it can be calculated by introducing d_1 into the information entropy function in Equation (2),

$$\begin{aligned} s_i^1 &= H(d_1) \\ &= -\left[d_1 \log(d_1) + (1-d_1)\log(1-d_1)\right] \end{aligned} \tag{2}$$

where $H(\cdot)$ denotes the entropy function. The larger the s_i^1, the higher is the uncertainty of the sample, and a higher weight is assigned to images with higher uncertainty. By contrast, a relatively lower weight should be assigned to images with lower uncertainty. The uncertainty obtained is used to update the weight features, and the updated weight $(1+s_i^1)$ is multiplied by the Hadamard product corresponding to the original feature map f_1 to obtain the final uncertainty-aware feature map h_1.

$$h_1 = f_1 \odot \left(1 + s_i^1\right) \tag{3}$$

At this point, the domain discriminator D_1 corresponds to the adversarial loss, as shown in Equation (4).

$$\begin{aligned} L_1 = &E\left[\log\left(D_1\left(F_1\left(f_1^s\right)\right)\right)\right] \\ &+ E\left[\log\left(1 - D_1\left(F_1\left(f_1^t\right)\right)\right)\right] \end{aligned} \tag{4}$$

The feature map f_2, obtained after applying the feature extraction module F_2, is cascaded with h_1^* to obtain the fused features e_2.

$$f_2 = F_2(h_1) \tag{5}$$

$$h_1^* = \text{Shuffle}(h_1) \tag{6}$$

$$e_2 = \text{concatenate}(h_1^*, f_2) \tag{7}$$

Further, the corresponding uncertainty-aware attention feature map s_i^2 can be expressed as follows:

$$\begin{aligned} s_i^2 &= H(d_2) \\ &= -D_2(e_2)\log(D_2(e_2)) \\ &\quad -[1-D_2(e_2)]\log[1-D_2(e_2)] \end{aligned} \tag{8}$$

Similar to the previous step, the related feature maps here are fused to obtain the fused feature map h_2 containing multi-scale image-level semantic information. The domain discriminator D_2 corresponds to the adversarial loss, as shown in Equation (10).

$$h_2 = f_2 \odot \left(1 + s_i^2\right) \tag{9}$$

$$\begin{aligned} L_2 &= E\left[\log\left(D_2\left(F_2\left(h_2^s\right)\right)\right)\right] \\ &\quad + E\left[\log\left(1 - D_2\left(F_2\left(h_2^t\right)\right)\right)\right] \end{aligned} \tag{10}$$

2.2. **Instance level efficient channel attention RPN alignment module (ILECA-RPN).**

In general, RPN networks are used to generate a region proposal for the target to be detected, and the IoU threshold can be set to further classify positive and negative samples and determine the initial target location. In RPN networks, "Distinction between foreground and background" and "generation of prediction coordinates of regression box position" are performed in the two branches of classification and regression, respectively, and the proportion of anchor points of the regression box in traditional RPN networks is often set to 1 : 1, 1 : 2, and 2 : 1, and the sizes are set to 88, 1616, and 3232; therefore, the classification branches of foreground and background include $32 = 18$ channels where "2" is with reference to positive and negative samples, and "3" refers to the three proportions and three sizes, respectively), while the regression branches include $34 = 36$ channels (where "4" denotes the left corner coordinates of the prediction box (x, y) and the width and height of the prediction box (w, h)). Considering that the number of feature channels is an important factor affecting the classification and regression performance of an RPN, this method proposes an adaptive enhancement for each channel to further improve the classification and regression prediction accuracy of the network.

The weighted feature map output from the ILFA module is fed into the ILECA-RPN alignment module to obtain the classification and regression results of the network. As shown in Figure 2, in ILECA-RPN, first, the weighted feature map is input into the RPN network and sent to the global average pooling layer after feature extraction, to obtain a $1 \times 1 \times C$ dimensional vector. Next, the input vector is processed in the adaptive enhancement layer of the proposed design, and an adaptive convolution kernel of size k is used to perform a fast one-dimensional convolution of the input vector to capture the correlation between each channel and the neighboring channels. Subsequently, the weight of each channel is predicted by the cross-channel information interaction between different channels, and this direct correspondence between channels and their weights, makes is possible to achieve adaptive enhancement of different channel features.

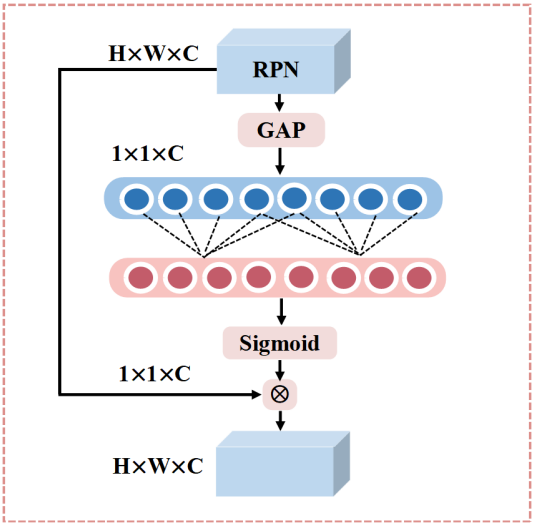


FIGURE 2. ILECA-RPN structure diagram.

Letting $y \in R^C$ denote the aggregated features without downscaling, the weights of the different channels in ILECA-RPN are obtained using Equation (11).

$$\omega = \sigma(W_y) \tag{11}$$

$$L_{\text{pixel}} = E_{X,Y}\left[(\|Y - G(X,\theta)\|_1)\right] \tag{12}$$

where Wdenotes the parameter matrix, as shown in Equation (12).

$$\begin{bmatrix} \omega^{1,1} & \cdots & \omega^{1,k} & 0 & 0 & \cdots & \cdots & 0 \\ 0 & \omega^{2,2} & \cdots & \omega^{2,k+1} & 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & \omega^{C,C-k+1} & \cdots & \omega^{C,C} \end{bmatrix} \tag{13}$$

In this experiment, considering that too many parameters increase the memory reading and writing costs, to avoid the computational inefficiency caused by the excessive complexity of the model and too many feature channels, the method of local cross-channel interaction was adopted to learn the attention weights between channels, as shown in Equation (13), such that for weight ω_i, local cross-channel interaction considers only the information interaction between y_i and its most adjacent k channels. In addition, sharing the learning parameters for all the feature channels, as shown in Equation (14), and applying a fast one-dimensional convolution with a kernel size of k, help achieve parameter sharing for all channels.

$$\omega_i = \sigma\left(\sum_{j=1}^{k} \omega_i^j y_i^j\right), y_i^j \in \Omega_i^k \tag{14}$$

$$\omega = \sigma\left(C1D_k(y)\right) \tag{15}$$

where Ω_i^k denotes the set of k adjacent channels of y_i and $C1D_k$ denotes one-dimensional convolution. It should be noted that when performing local cross-channel information interaction, the coverage of information interaction (e.g., the size of the convolution kernel k of the one-dimensional convolution) needs to be further considered. Obtaining the optimal coverage by manual adjustment, leads to a large amount of resource consumption; however, the channel dimension C is determined by a mapping relationship, as shown in Equation (15):

$$C = \Phi(k) \tag{16}$$

Here, $\Phi(\cdot)$ denotes the mapping relationship. Equation (16) shows a common type of linear mapping relationship, and this linear function is limited by its characterization ability. Considering that in our problem, the channel dimension C is usually an exponent of two, the linear mapping relation of Equation (16) can be extended to a nonlinear function, as shown in Equation (17) to improve the characterization ability of the function.

$$\Phi(k) = \gamma \times k - b \tag{17}$$

$$C = \Phi(k) = 2^{\gamma \times k - b} \tag{18}$$

Further, based on a given channel dimension C, the size of the convolution kernel k can be determined adaptively using Equation (18).

$$k = \varphi(C) = \left| \frac{\log_2(C)}{\gamma} + \frac{b}{\gamma} \right|_{\text{odd}} \tag{19}$$

where, $|x|_{odd}$ denotes the selection of the nearest odd-numbered channels, $\gamma = 2, b = 1$. The use of mapping gradients allows for different interactions between channels of different dimensions, with higher-dimensional channels having longer interactions through nonlinear mapping, and lower-dimensional channels having shorter interactions.

The MUCA module is a multilevel uncertainty-aware context module, the main purpose of which is to automatically capture target-specific and source-specific features between domain-invariant structures. It mainly consists of three auxiliary feature extractors, F_3, F_4, F_5 and three domain discriminators D_3, D_4, D_5, the details of which can be found in the literature [30].

Therefore, the adversarial loss of the entire network is expressed by Equation (19). The total network loss consists of classification and regression losses, as shown in Equation (20). The overall target loss is expressed by Equation (21).

$$L_{adv} = L_1 + L_2 + L_3 + L_4 + L_5 + L_{ins} \tag{20}$$

$$L_{\text{det}} = L_{cls} + L_{reg} \tag{21}$$

$$L_{\text{loss}} = \max \min_{D_i} L_{\text{det}} + \lambda L_{adv} \tag{22}$$

where D_i represents the multi-level domain classifiers $D_1, D_2, D_3, D_4, D_5, D_{ins}$; F_i represents the multi-level feature extractors F_1, F_2, F_3, F_4, F_5; L_i denotes the loss functions of different domain discriminators, where $D_3, D_4, D_5, F_3, F_4, F_5, L_3, L_4, L_5$ have the same form as in [27]. λ is a controllable hyperparameter that adjusts the degree of influence of the adversarial loss on the detection loss.

3. **Experimental results and analysis.**

3.1. **Datasets and Experimental Settings.**

To evaluate the performance of the proposed domain-based adaptive solar cell surface defect detection network shown in Figure 3, we generated three paired datasets: "source domain-exposure domain," "source domain-night domain," and "source domain-noise domain." The source domain dataset consists of labeled solar cell surface defect images collected under standard conditions, and the three target domain datasets consist of contaminated unlabeled solar cell surface defect images. In the experiment, solar cell surface defect images were simulated in overexposed daytime, low-light nighttime, and in the presence of dust and grit in the scene by applying luminance conversion and adding simulation noise to the images. The source domain dataset contained 2462 training images and 300 test images. All three target domain datasets contained 2012 training images and 675 test images.

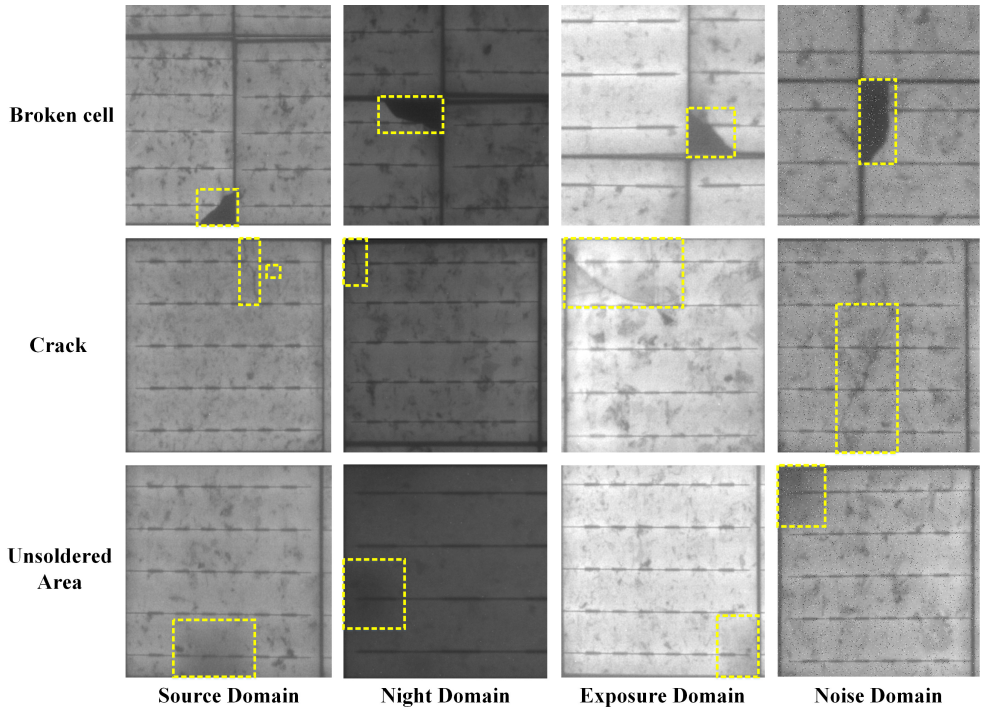


FIGURE 3. Enlarged display of broken cells, cracks, and unsoldered areas in solar cell surface images in source domain and target domain datasets. The first, second, third, and fourth columns are the local enlarged view of the three defects in the source domain, night domain, exposure domain, and noise domain datasets, respectively.

In this study mean average precision (mAP), precision, and recall were used as metrics to quantitatively evaluate the performance of defect detection.Precision indicates the accuracy of the correct positive sample, and Recall represents the coverage of the correctly predicted positive sample, and they can be expressed as:

$$\text{Prec} = \frac{TP}{TP + FP} \tag{23}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{24}$$

Recall and precision influence and restrict each other, as they are inversely proportional. Thus, the higher the recall, the lower is the precision, and the lower the recall, the higher is the precision. The PR curve describes the relationship between two, with recall as the horizontal axis and precision as the vertical axis. By comparing the size of the area under the PR curve, the training performance of the target detection network can be determined. In general, the larger the recall and the larger the precision, the closer is the PR curve to the upper right, and the better is the performance of the network. As an indicator of object detection accuracy, the mean average precision mAP, is the average of AP values of all categories of sample data, which represents the average precision performance of the detector for multiple object categories. The higher the mAP value, the better the performance of the detector.

The experimental environment in this study consisted of a 64-bit Linux system with CPU of i7-9700KF, with GTX2080 (11 Gb video memory), GPU acceleration, and the PyTorch deep learning platform. The pretrained VGG16 and Resnet101 models on ImageNet were used as the backbone network for feature extraction; DA-OD-MEAA was

the baseline model; stochastic gradient descent was used to optimize the network; the hyperparameterwas set to 1, and the IOU thresholds were all 0.5. After training, the test images in the dataset were input into the model, whereby three types of defects–broken cells, cracks, and unsoldered areas– were identified and localized. The classification and localization boxes of the defects in the test images were then output.

3.2. **Visual effect analysis.**

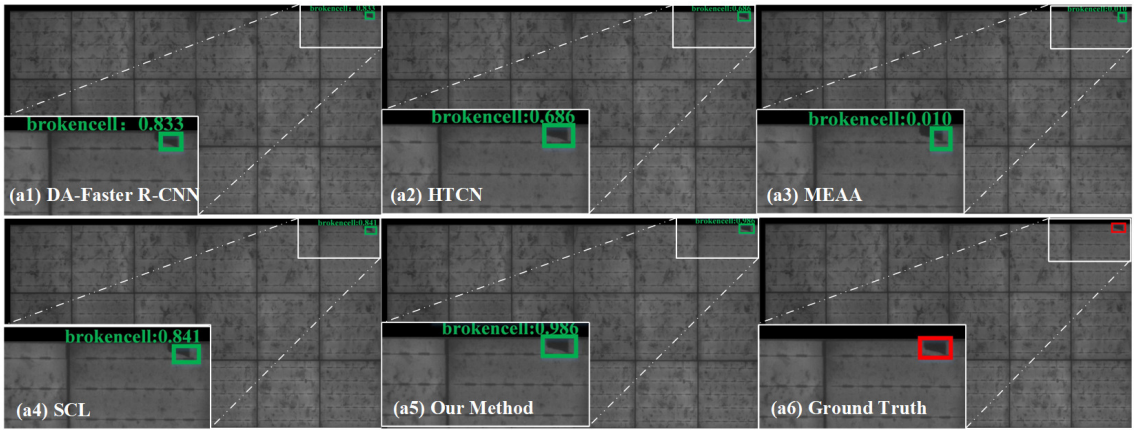
Figure 4 shows the detection results for three defects: broken cells, cracks, and unsoldered areas in the night domain dataset evaluated by five different algorithms, namely, the proposed method, DA-Faster R-CNN, HTCN, MEAA, and soft competitive learning (SCL). The detection results vary depending on the method. Figure 4(a) displays the results for broken cell defect detection. The accuracy of the regression boxes in the different experimental results decreased in the following order: the proposed method > SCL > DA-Faster R-CNN > HTCN > MEAA. Except for our experimental results, the regression boxes of the other four experimental results do not fully contain defects; the SCL experimental results have a confidence level of 0.841, and the performance of the regression box is also poor. In Figure 4(b), the detection performance of different detection frameworks decreases significantly when the defect is a crack with an elongated shape and is very similar to the background such that DA-Faster R-CNN, HTCN, and MEAA all generate wrong regression boxes. Although the SCL regression box does not deviate completely, the deviation is still large, whereas the regression box of the proposed method encloses the crack with high accuracy. In Figure 4(c), the regression boxes of the other four comparison methods also deviate to different degrees for the unsoldered area defects, but the detection performance is significantly improved compared to that of the crack defects, which is primarily attributed to the fact that the defect type of unsoldered area is relatively large and easy to detect. In summary, the detection performance of our method in the night domain for the three defects of broken cells, cracks, and unsoldered areas are all highly accurate, and the detection regression box is closer to that of the ground truth than those of the comparison algorithms. The proposed method is more accurate in enclosing small defects, such as cracks.

As shown in Figure 5, the overall detection of broken cell and unsoldered area defects of different detection models is more accurate in the exposure domain compared to the night domain. The detection results of cracks show that MEAA and SCL have missed some of the small, inconspicuous crack defects; HTCN performance is also poor in detecting small cracks with the regression box characterized by low accuracy and false detection.

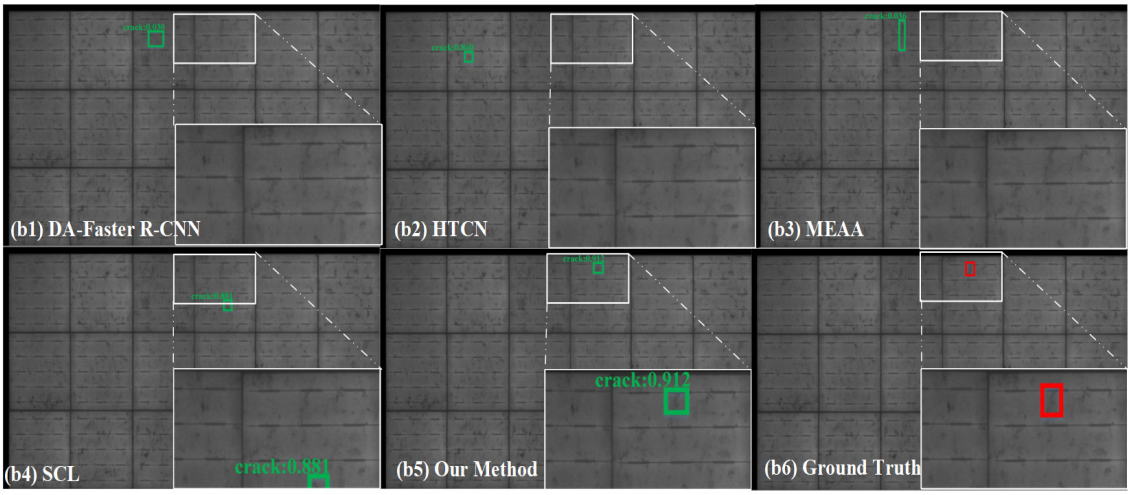
Figure 6 shows the detection results of the five algorithms for three defects: broken cells, cracks, and unsoldered areas, in the noise domain dataset. In Figure 6, HTCN falsely detects broken cell defects. Although the other models can detect broken cell defects, the results show a large deviation between the regression box and ground truth, indicating inaccurate defect localization. MEAA and SCL can misjudge crack defects. In the detection results of DA-Faster R-CNN and HTCN for unsoldered area defects, there are missed defects; however, unsoldered area defects can be detected by MEAA, SCL, and the proposed method, and the detection results of the proposed method are closer to ground truth. In summary, in the night, noise, and exposure domains, the proposed method performed well in detecting three types of defects: broken cells, cracks, and unsoldered areas, and there were no false or missed detections.

3.3. **Quantitative Analysis.**

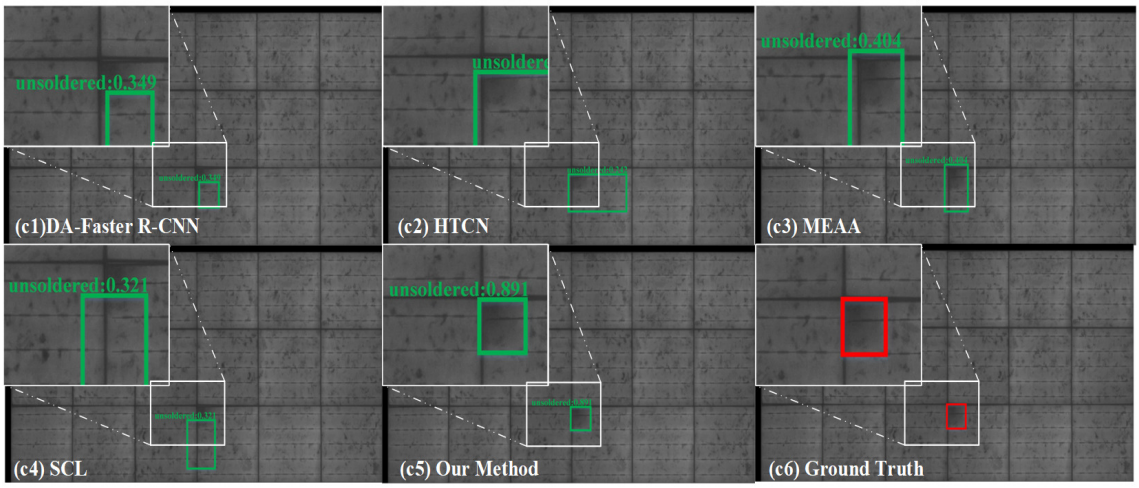
Figure 7, 8, and 9 show the precision-recall curves of the different detection models for



(a) Schematic diagram of the detection results of different detection algorithms for broken cells in the night domain.

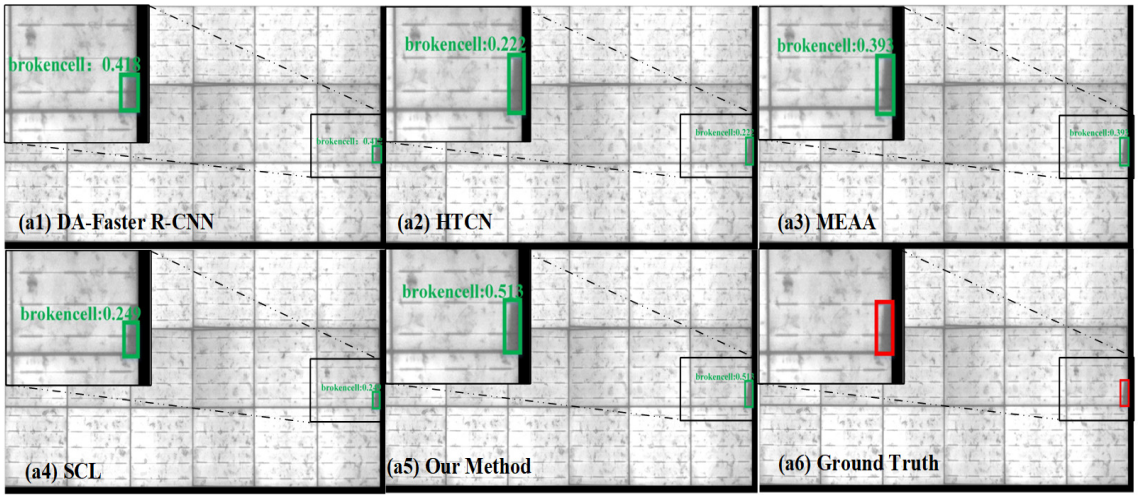


(b) Schematic diagram of the detection results of different detection algorithms for cracks in the night domain.

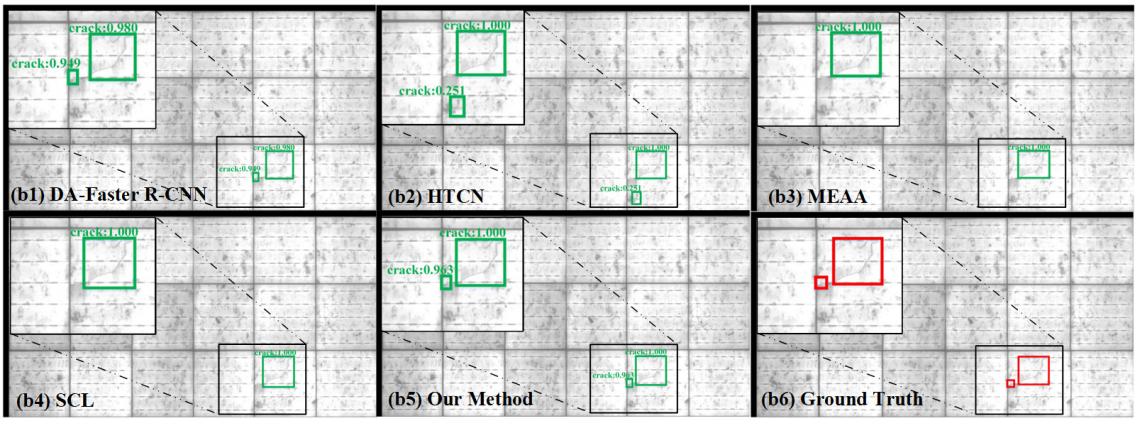


(c) Schematic diagram of the detection results of different detection algorithms for unsoldered areas in the night domain.

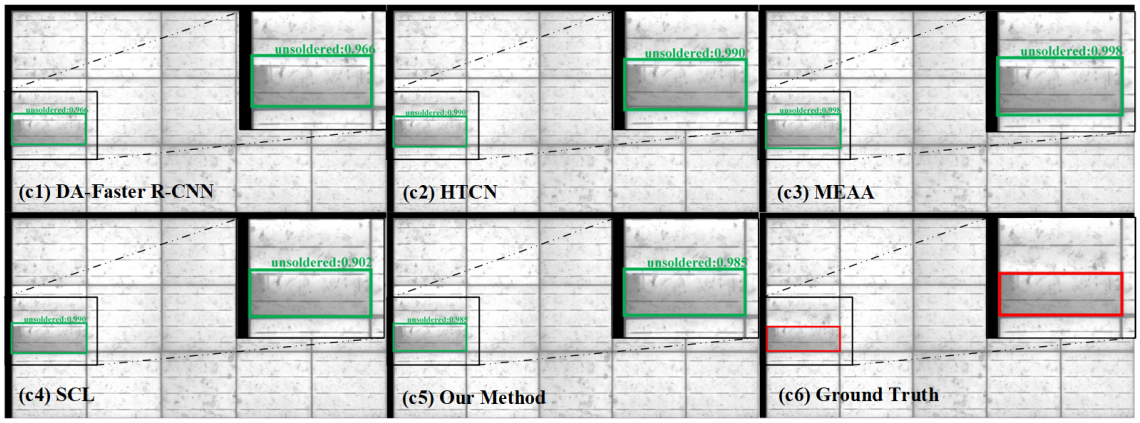
FIGURE 4. Schematic of the detection results of three kinds of defects of different detection models for the night domain scenario.



(a) Schematic diagram of the detection results of different detection algorithms for broken cells in the exposure domain.

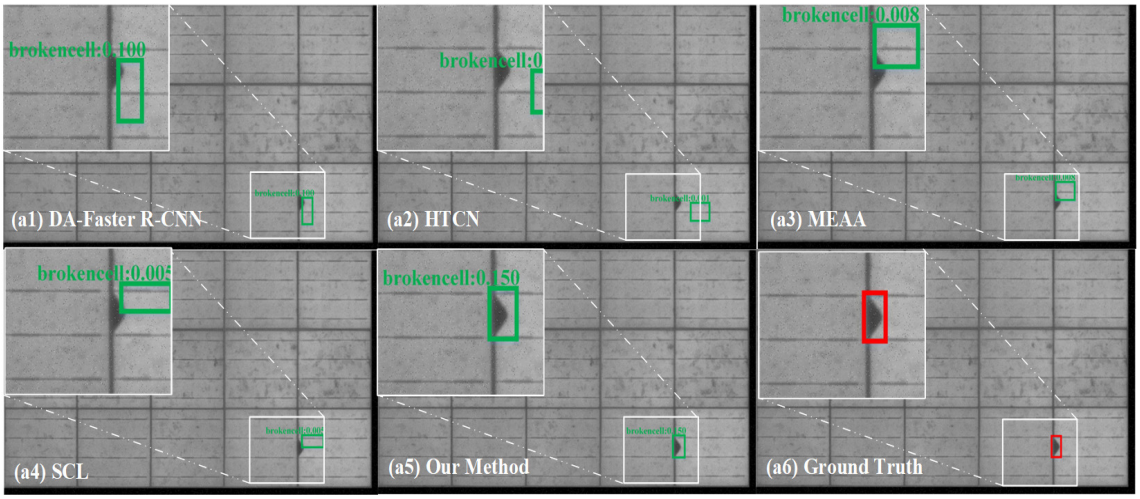


(b) Schematic diagram of the detection results of different detection algorithms for cracks in the exposure domain.

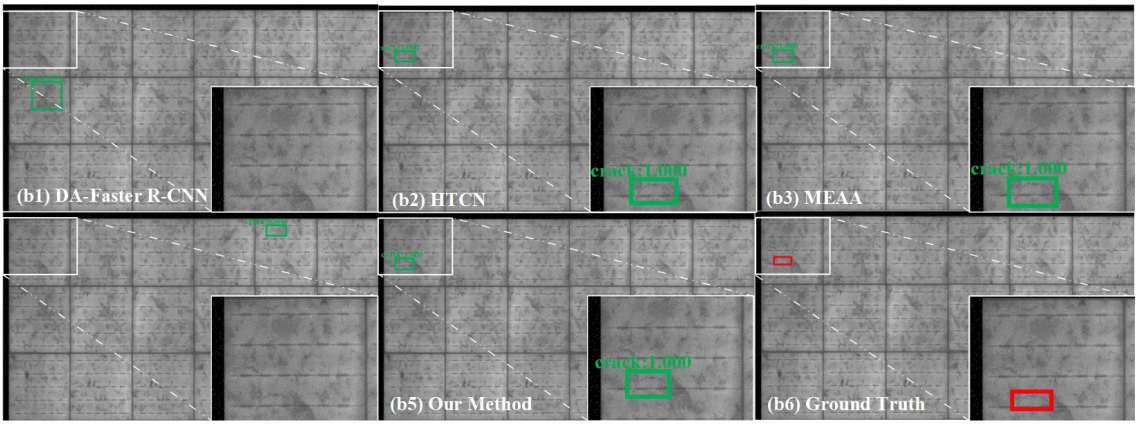


(c) Schematic diagram of the detection results of different detection algorithms for unsoldered areas in the exposure domain.

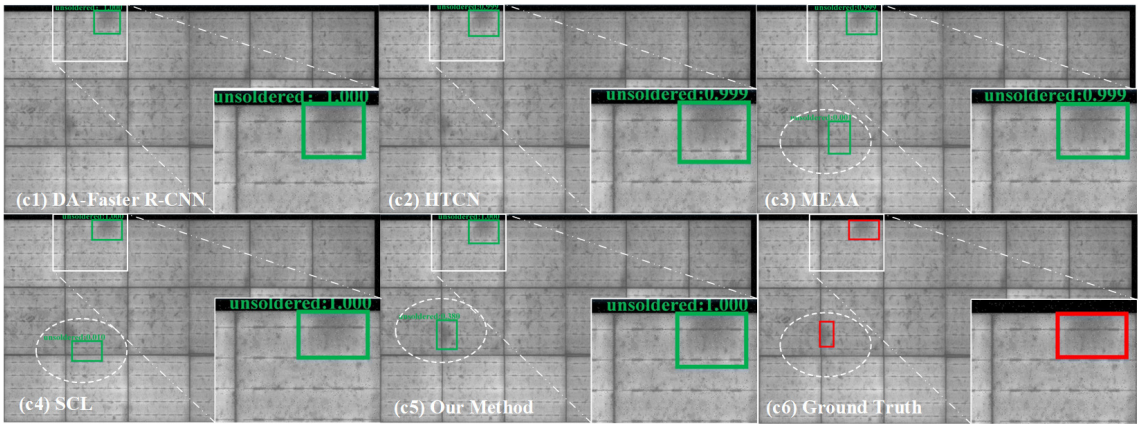
FIGURE 5. Schematic of the detection results of three kinds of defects of different detection models for the exposure domain scenario.



(a) Schematic diagram of the detection results of different detection algorithms for broken cells in the noise domain.



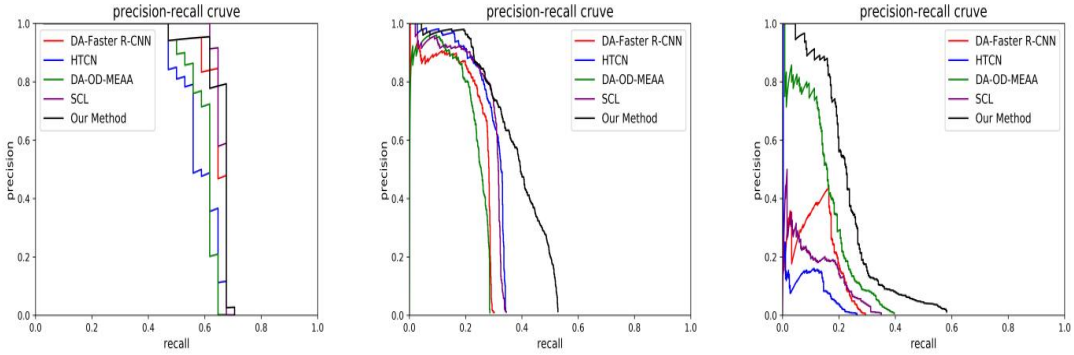
(b) Schematic diagram of the detection results of different detection algorithms for cracks in the noise domain.



(c) Schematic diagram of the detection results of different detection algorithms for unsoldered areas in the noise domain.

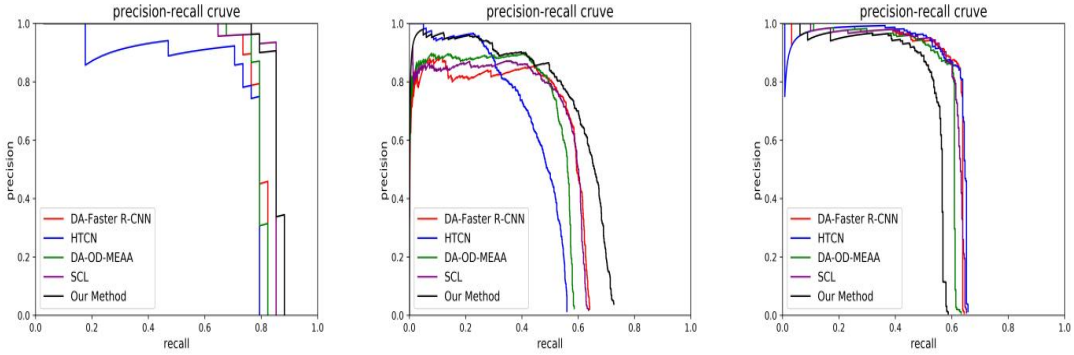
FIGURE 6. Schematic of the detection results of three kinds of defects of different detection models for the noise domain scenario.

the three target domain scenarios. The PR curves of the three defects of the proposed detection model, illustrated by the black solid line, are higher than the PR curves of the other detection models in the night-domain scenario, exhibiting the best detection effect on crack defects. The performance of the proposed method is also more stable in the exposure and noise domains, and the experimental results are generally better than those of other algorithms.



(a) brokencell (b) cracks (c) unsoldered areas

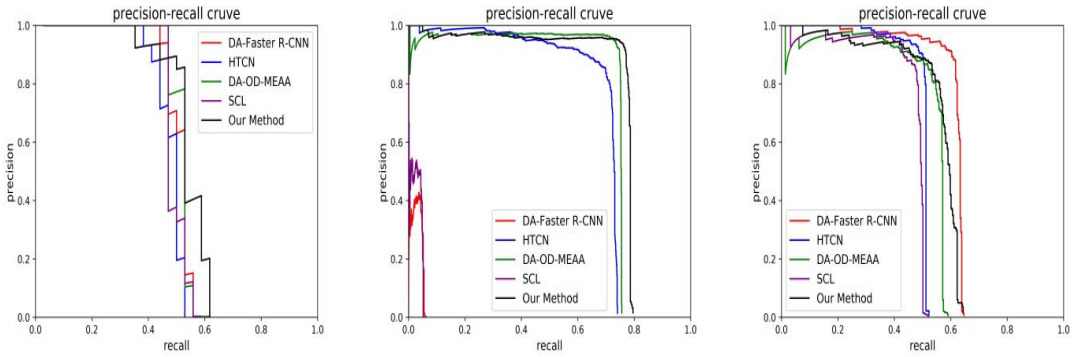
FIGURE 7. Schematic diagram of the prescision-recall curves of different detection models in the night domain scenario.



(a) brokencell (b) cracks (c) unsoldered areas

FIGURE 8. Schematic diagram of the prescision-recall curves of different detection models in the exposure domain scenario.

To visually analyze the cross-domain detection capability of the proposed method, a statistical table (Table 1) was created for the detection accuracy results of different models for the three defects in different target domains. Even for the same defect, the detection performances of different models in different target domains vary significantly, as reflected by the differences in the detection accuracy values obtained, indicating that the environment of the background domain where the target is located affects the performance of the detection model. Overall, the night domain scenario increased the difficulty of detecting unsoldered area defects with great similarity to the background and crack defects with insignificant gray level changes such that the detection accuracy values obtained by the five models on the night domain dataset for the three defects are lower than those



(a) brokencell (b) cracks (c) unsoldered areas

FIGURE 9. Schematic diagram of the prescision-recall curves of different detection models in the noise domain scenario.

obtained on the exposure and noise domain datasets. In comparison, the proposed method outperforms other domain adaptive models in the detection accuracy values obtained for cracks in unsoldered areas. In the exposure and noise domains, the proposed method achieved the best average detection results for the three defects and the highest mAP values.

TABLE 1. Statistics of the detection accuracy of three types of defects in different target domains

Different method		Defect	The Night Domain		The Exposure Domain		The Noise Domain	
			AP	mAP(%)	AP	mAP(%)	AP	mAP(%)
DA-Faster R-CNN		brokencell	64.65	32.68	76.46	63.58	51.39	55.25
		crack	24.61		54.49		10.60	
		unsoldered	8.78		59.78		61.32	
MEAA		brokencell	60.62	34.98	75.25	61.39	52.59	41.10
		crack	24.72		47.93		60.96	
		unsoldered	19.60		60.98		52.21	
HTCN		brokencell	58.62	30.42	74.11	60.59	48.17	50.77
		crack	29.61		45.04		69.62	
		unsoldered	3.03		62.61		34.52	
SCL		brokencell	63.65	34.69	72.03	62.63	48.89	36.26
		crack	32.41		54.57		13.10	
		unsoldered	8.01		61.30		47.78	
Our Method	Source Domain	broken cell	63.64	63.56	63.65	56.43	63.64	59.14
		crack	59.14		52.39		52.48	
		unsoldered	67.89		53.25		61.31	
	Target Domain	broken cell	63.06	42.66	80.91	64.89	54.86	60.47
		crack	39.84		58.60		70.56	
		unsoldered	34.69		55.17		55.98	

3.4. **Ablation study.**

The proposed method uses two different strategies: the image-level feature alignment module ILFA and instance-level efficient channel attention RPN alignment module ILECA-RPN. Ablation experiments were performed to analyze the feasibility of both strategies. Table 2 shows that both of the listed strategies lead to a significant improvement in the mAP value. ILFA alone leads to a 5.6 % improvement in mAP, whereas ILECA-RPN

alone leads to a 2.54 % improvement in mAP, and combined a 7.68 % improvement in mAP is obtained.

TABLE 2. Baseline detection effect obtained by training the network using different strategies for the exposure domain scenario.

Ablation study	Submodules		mAP(%)
	ILFA(M_1)	ILECA-RPN(M_2)	
Baseline			61.39
Baseline+M_1	√		63.21
Baseline+M_2		√	62.78
Baseline+M_1 + M_2	√	√	64.89

4. Conclusion.

In this study, aiming at the differences in the data distributions of real scenarios and standard datasets, which result in a significant decrease in the accuracy and robustness of domain adaptive solar cell surface defect detection, a solar cell surface defect detection algorithm was proposed based on domain adaptation. For image-level alignment, an image-level semantic context alignment module was used to reduce the image-level differences in adaptive detection between different domains through adversarial learning and adaptive weight enhancement based on pixel information entropy. For instance-level alignment, an instance-level efficient channel attention RPN module was proposed considering the strong dependency of the classification and regression branches of the RPN module on the channel mechanism, employing the idea of local cross-channel interaction strategy without dimensionality reduction and group convolution, to adaptively determine the size of the convolution kernel by the given number of channel dimensions. This ensures that the high-dimensional channels have strong interactions, whereas low-dimensional channels have weaker interactions through nonlinear mapping, thus concurrently improving the pre-determined results of instance-level classification and regression as well as the accuracy of instance-level prediction. The experimental results show that the domain-adaptive detection accuracy values of the proposed method are significantly improved for the different domains, and the method in this study displays considerable superiority in solar cell surface defect detection compared to the latest domain-adaptive detection networks.

Acknowledgment.

This work was supported by the Natural Science for Youth Foundation of China under Grant 62001321 and Fundamental Research Program of Shanxi Province under Grants 202103021224265.

REFERENCES

[1] S. Kumar, A. Damaraju, A. Kumar, S. Kumari, and C. M. Chen, "Lstm network for transportation mode detection," *Journal of Internet Technology*, vol. 22, no. 4, pp. 891–902, 2021.

[2] K. Wang, C. M. Chen, M. S. Hossain, G. Muhammad, and S. Kumar, "Transfer reinforcement learning-based road object detection in next generation iot domain," *Computer Networks*, vol. 193, p. 108078, 2021.

[3] Z. Fu and Z. Wang, "Research on vision-based intelligent detection method of circle features," in *Journal of Physics: Conference Series*, vol. 1684, no. 1. IOP Publishing, 2020, p. 012097.

[4] W. Luo, N. An, Y. Sun, and J. Xu, "Research progress of digital image cracks recognition," in *IOP Conference Series: Earth and Environmental Science*, vol. 170, no. 2. IOP Publishing, 2018, p. 022072.

[5] Y. Deng, W. Li, S. Lin, Q. Lu, and X. Liu, "Detection method of water channel defects in engine block based on deep learning," in *Journal of Physics: Conference Series*, vol. 2366, no. 1. IOP Publishing, 2022, p. 012020.

[6] S. K. Mishra, K. Nandini, S. H. Ahammad, S. Inthiyaz, B. R. Altahan, L. K. Smirani, M. A. Hossain, A. N. Z. Rashed *et al.*, "Machine learning based models for defect detection in composites inspected by barker coded thermography: a qualitative analysis," *Advances in Engineering Software*, vol. 178, p. 103425, 2023.

[7] K. Zhang and S. Zhu, "Defect detection image processing technology based on swarm intelligence optimization algorithm," in *Journal of Physics: Conference Series*, vol. 2400, no. 1. IOP Publishing, 2022, p. 012031.

[8] Z. Xing, Z. Zhang, X. Yao, Y. Qin, and L. Jia, "Rail wheel tread defect detection using improved yolov3," *Measurement*, vol. 203, p. 111959, 2022.

[9] J. Zhang, J. Jing, P. Lu, and S. Song, "Improved mobilenetv2-ssdlite for automatic fabric defect detection system based on cloud-edge computing," *Measurement*, vol. 201, p. 111665, 2022.

[10] J. Lamb, M. Echlin, A. Polonsky, R. Geurts, K. Pusch, E. Raeker, A. Botman, C. Torbet, and T. Pollock, "Merging machine learning and tribeam tomography for 3d defect detection in an am coni-based superalloy," *Microscopy and Microanalysis*, vol. 28, no. S1, pp. 862–863, 2022.

[11] S. Xiaoyu, Y. Liu, X. Xinghua, and C. Zhili, "Defect detection method for solar cells based on human visual characteristics," in *2020 5th International Conference on Mechanical, Control and Computer Engineering (ICMCCE)*. IEEE, 2020, pp. 515–518.

[12] F. Zhang, T. Y. Wu, J. S. Pan, G. Ding, and Z. Li, "Human motion recognition based on svm in vr art media interaction environment," *Human-centric Computing and Information Sciences*, vol. 9, no. 1, p. 40, 2019.

[13] Y.-C. Chiou, J.-Z. Liu, and Y.-T. Liang, "Micro crack detection of multi-crystalline silicon solar wafer using machine vision techniques," *Sensor Review*, vol. 31, no. 2, pp. 154–165, 2011.

[14] D.-M. Tsai, C.-C. Chang, and S.-M. Chao, "Micro-crack inspection in heterogeneously textured solar wafers using anisotropic diffusion," *Image and vision computing*, vol. 28, no. 3, pp. 491–501, 2010.

[15] X. Qian, H. Zhang, C. Yang, Y. Wu, Z. He, Q.-E. Wu, and H. Zhang, "Micro-cracks detection of multicrystalline solar cell surface based on self-learning features and low-rank matrix recovery," *Sensor Review*, vol. 38, no. 3, pp. 360–368, 2018.

[16] D.-M. Tsai, G.-N. Li, W.-C. Li, and W.-Y. Chiu, "Defect detection in multi-crystal solar cells using clustering with uniformity measures," *Advanced Engineering Informatics*, vol. 29, no. 3, pp. 419–430, 2015.

[17] J. Guan, X.-L. Chen, Y. Huang, and Y. He, "Adaptive fractional fourier transform-based detection algorithm for moving target in heavy sea clutter," *IET Radar, Sonar & Navigation*, vol. 6, no. 5, pp. 389–401, 2012.

[18] W.-C. Li and D.-M. Tsai, "Wavelet-based defect detection in solar wafer images with inhomogeneous texture," *Pattern Recognition*, vol. 45, no. 2, pp. 742–756, 2012.

[19] F. Zhang, T. Y. Wu, and G. Zheng, "Video salient region detection model based on wavelet transform and feature comparison," *EURASIP Journal on Image and Video Processing*, vol. 2019, no. 1, 2019.

[20] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[21] X. Zhang, Y. Hao, H. Shangguan, P. Zhang, and A. Wang, "Detection of surface defects on solar cells by fusing multi-channel convolution neural networks," *Infrared Physics & Technology*, vol. 108, p. 103334, 2020.

[22] M. Sharma, J. Lim, and H. Lee, "The amalgamation of the object detection and semantic segmentation for steel surface defect detection," *Applied Sciences*, vol. 12, no. 12, p. 6004, 2022.

[23] Y. Cao, D. Pang, Q. Zhao, Y. Yi, Y. Jiang, C. Tian, F. Wang, and J. Li, "Improved yolov8-gd deep learning model for defect detection in electroluminescence images of solar photovoltaic modules," *Engineering Applications of Artificial Intelligence*, vol. 131, no. 107866-, 2024.

[24] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. Van Gool, "Domain adaptive faster r-cnn for object detection in the wild," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[25] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[26] C. Chen, Z. Zheng, X. Ding, Y. Huang, and Q. Dou, "Harmonizing transferability and discriminability for adapting object detectors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8869–8878.

[27] K. Saito, Y. Ushiku, T. Harada, and K. Saenko, "Strong-weak distribution alignment for adaptive object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6956–6965.

[28] Z. He and L. Zhang, "Multi-adversarial faster-rcnn for unrestricted object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6668–6677.

[29] X. Zhu, J. Pang, C. Yang, J. Shi, and D. Lin, "Adapting object detectors via selective cross-domain alignment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 687–696.

[30] D.-K. Nguyen, W.-L. Tseng, and H.-H. Shuai, "Domain-adaptive object detection via uncertainty-aware distribution alignment," in *Proceedings of the 28th ACM international conference on multimedia*, 2020, pp. 2499–2507.

[31] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: Efficient channel attention for deep convolutional neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 534–11 542.