INTRODUCTION

Solar power is currently one of the most important forms of new energy power generation. In the photovoltaic power station, photovoltaic modules undertake the critical energy conversion function, which is the core component. If the quality of photovoltaic modules is not guaranteed, the manufacturer’s reputation will be affected, and the power station will also be under tremendous pressure on operation and maintenance.

Some relevant studies\(^1,2\) show that 41\% of the failures in photovoltaic power plants are caused by panels. According to the quality report of photovoltaic modules given by German Rhine and Hamelin Solar Energy Institute, the failure rate of photovoltaic modules is increasing year by year.

In this paper, an object detection algorithm for electroluminescence image defects of photovoltaic modules based on deep learning is proposed. The proposed algorithm can achieve high precision and real-time detection of defects in the photovoltaic module production line.
Research Institute, about 6% of photovoltaic modules were damaged during production and transportation before installation, which may be related to impact and vibration during transportation. Module defects will seriously affect the stable and efficient operation of the power plant, so it is necessary to carry out quality inspection during the production and power generation.3

Methods as current–voltage curve scanning, electroluminescence imaging, infrared radiation test, ultraviolet fluorescence, etc., are widely used methods for PV module failure detection.4 Among them, EL image has high resolution, and can detect cracks and other defects that cannot be found by naked eyes, so it has become the usual detection method at present in both production line and onsite of plant. The EL images inspection manually is time-consuming and laborious, besides that subjective factor affects the whole inspection process, which makes it difficult to achieve standardized defect detection. Therefore, a fast and accurate detection algorithm is needed to automatically detect defects in the PV modules.

The object detection algorithm requires not only classifying the object category but also using one bounding box to mark the location. Nowadays, object detection algorithms based on deep learning have become the mainstream. In many cases, it has replaced the traditional object detection algorithm. Object detection with artificial neural network (ANN) has been applied to many fields including EL image.

In this paper, we propose an object detection YOLO-PV based on YOLO. The YOLO-PV network structure is optimized specially for PV modules, to rapidly and accurately detect defects. The contributions of this paper are as follows:

1. To propose a standard for detecting defects in EL images of PV modules and establish a complete PV module defect detection data set.
2. The YOLO-PV network structure is proposed combined with the actual situation of the photovoltaic module defect detection task. Through experiments on the PV module data set, we verify the effectiveness of the network.
3. We use various data augment methods to solve the overfitting problem and give an effective data augment suggestion.

The remainder of this work is organized as follows: Section 2 introduces related work. Section 3 introduces the EL image and object detection data set. Section 4 includes data preprocessing, network structure design, network loss function, and evaluation criteria. In Section 5, the performance of the proposed model is evaluated by experiments, and the experimental results are given and discussed. Finally, concluding observations and the direction of the future work are provided in Section 6.

2 RELATED WORK

As far as we know, many researches on EL image defect detection revolve around image classification: Deitsch et al.5 uses an improved VGG-19 model to predict the defect probability of cell EL images. By rounding the predicted continuous probability to the nearest neighbor of the four original classes, they directly compare convolution neural network (CNN) decisions against the ground truth labels. The results in public data set of solar cells show that the method using CNN is more accurate than the support vector machine (SVM), reaching 88.42%. Akram et al.6 regard the identification of cell defects as a two-category problem. They propose a neural network classification model with only four layers of convolution. Also, they compress the input image size to save processing time and use data enhancement to improve the robustness of the network. In public solar cell data set, the process speed of each picture is 8.07 ms per picture, and the accuracy is 93.02%.

Tang et al.7 divide the cell into defect-free, microcrack, finger-interruption, break in private data set, regarding the defect identification problem as a four-category problem. Given the lack of data sets, they use one generative adversarial networks (GAN) to fake additional images for data enhancement, which greatly improves the accuracy of classification. GAN enhancement is also proved to be one good method for solar cell classification in our previous work.8

For the research of object detection tasks in EL images, Liu et al.9 improved the feature extraction part and RPN part in Faster-RCNN and proposed GA-Faster-RCNN. The model can identify the defects on the cell and mark its location. This work uses GA-RPN to greatly reduce the number of candidate frames. The detection speed is improved in self-made data set. Zhang et al.10 designed a detection algorithm of surface defects on solar cells, which combines the results of Faster R-CNN and R-FCN to improve detection precision and position accuracy. This method creatively integrates the detection results of two mature network models, and the mAP of three defects on the self-made data set reaches 85.7%. Otamendi et al.11 perfectly combine three deep learning technologies: Faster-RNN, EfficientNet, and autoencoder to build an end-to-end deep learning pipeline that detects, locates, and segments cell-level anomalies from the entire photovoltaic modules via EL images. In the object detection part, this work improves the Faster-RNN to better perform the solar cell detection task. This work not only does the object detection of PV module defects, but
also uses autoencoder to complete the task of anomaly segmentation module.

While briefly summarizing the results of the existing research, we find that the current research has the following problems: low detection accuracy, slow detection speed, non-uniform defect detection standards. Although the existing methods have greatly saved the inspection time, it still cannot meet the real-time processing of the EL images. It is still challenging to apply in some environments with sensitive hardware resources and high real-time requirements. At the same time, a large number of false detections and missed detections caused by low accuracy seriously affect subsequent analysis and the reliability of algorithms; hence, further research is needed.

3 | DEFECT AND DATA SET

Electroluminescence, abbreviated as EL, refers to the luminescence phenomenon in which materials directly convert electrical energy into light energy under external electric field. Injection electroluminescence is commonly used in the photovoltaic industry. This method was first proposed by Y. Takahashi. The specific step is to apply a forward bias to the solar cell. The forward current will inject many unbalanced carriers into the solar cell, which will produce injection electroluminescence. Since the intrinsic silicon band gap is about 1.12 eV, we can calculate that the peak of infrared light generated by the electroluminescence should be around 1150 nm, which is beyond the strong response range of the CMOS sensor. So, when using the SLR camera to shoot EL image, it needs in a dark room to avoid the adverse effects of ambient light. In other way, the SWIR camera with InGaAs sensor will be good at imaging the EL of solar cell and make it possible to do fast imaging which leads to lower cost in PV plant mapping when cooperated with drone.

3.1 | Failure detected in electroluminescence images

Takashi Fuyuki et al. point out that for the detailed inspection of cell performance, the most essential material parameter is the minority carrier diffusion length (or lifetime), which determines the collection efficiency. They also point out that the brightness intensity of the EL image has a one-to-one correspondence with the minority carrier diffusion length. Würfel et al. also consider that the minority carrier diffusion length can be uniquely determined by the brightness of the EL image. This is because the minority carrier diffusion will be low in the defective area of the cell, and the corresponding area in the image will be dark. The defects of solar cells can be classified into two categories: the first one is called the intrinsic deficiencies which arise due to material properties, such as black core and broken grid. The second one is the process-induced extrinsic defects, such as microcracks and breaks. According to the International Energy Agency standards, these defects will have a certain impact on the safety and power generation capacity.

As shown in Figure 1, we can observe the following defects through EL images:

1. **Black area**: An irregular black area in the EL image, often caused by material defects or poor contact caused by inappropriate welding.
2. **Cracks**: Potential cracks cannot be found directly by naked eyes. It can only be detected by EL technology. This type of cracks may cause damage to the cell or not. They are generally caused by the weak external stress from manufacturing, transportation, and installation.
3. **Break**: Severe cracks cause part of the cell completely be electrically isolated from the others, represents as a black area in the EL image. Severe external thermal stress and mechanical stress may cause break cell.
4. **Finger failure**: Finger failure is a black line perpendicular to the bus bar, which may be caused by too low printing height, too low printing pressure, or problems with the bus bar welding.
5. **Low cell**: Low cell is significantly darker than other cells in the module’s EL image. Cells with different efficiencies are welded on the same modules, and the lifetime of the cell’s minority carrier is quite different, which is the main reason for the low cell.
6. **Scratch**: Scratch is the appearance of single or multiple smooth and uneven black lines on the cell. The reason of scratch may be that sharp objects cause nicks on the cells during transportation on the production line.
7. **Black cell**: Black cell appears on the EL image as one or more cells are completely black. The reasons for the black cell include short-circuit of the cell, low-quality silicon chip material, or cracks caused by improper welding process.

Besides, the cell may also have defects, such as broken corner, black edges. Some abnormalities may also occur during the EL image shooting.

3.2 | EL image object detection data set

For supervised learning, a complete training process requires labeled data set. However, the detection standards for defects in production lines and power stations are often inconsistent. For power station operators, they hope the defective cells that impact power generation efficiency and
safety can be detected in time. As for production line, not only defects as mentioned above must be found, but also finger failure, black corner, etc., because these defects reflect the deficiencies in raw materials or manufacturing processes. Through defect detection, these deficiencies can be found in time to avoid affecting module quality. It is worth mentioning that different types of defects are defined in this work in order to better mark the defective cell. However, in the following work, these defects are uniformly marked and detected as the same type, because the strategy we adopt on the production line is to use neural network to identify defects first, and then use manual classification if necessary. Based on the demands of manufacturers, in this paper, we establish a set of defect detection standards that can be applied on the production line. The main points are as follows:

1. For defects that may reflect material defects and welding errors, such as black area, finger failure, we must mark them as defects.
2. Cracks include cross-cracks, parallel cracks, reticular cracks, etc. As long as the crack causes the failure area of the cell, we need to mark the crack as a defect. For the slight crack, we ignore the V-shaped crack which is <6 mm and the cross-crack which is <14 mm.
3. For the scratch, we ignore the single scratch <8 mm. For the single scratch >8 mm and multiple scratches, we mark them as defects.
4. For all other defects, such as break, low cell, and broken corner, they are marked as defects in this work.

These pictures are taken by different types of equipment, including a professional EL camera (NXL-100) and a modified SLR camera (Nikon D5300), which also makes the collected images more diverse. The parameters for shooting are focal length 26 mm, aperture f/4 and exposure time 4–10 s. An inspection expert and two graduate students mark these pictures. We use labeling software to make labels. Mark the defects in the EL image with a rectangular box and save the marking information (including the size and position of the box and the object type) in an XML file. We take 360 of 2144 pictures as the test set, and the remaining 1784 pictures are divided into training set and verification set according to 8:2.

The collected images have a resolution of 7380 × 3838. To save computing resources and remove redundant information, we first resize the image to a suitable size. In this work, the image is scaled to 960 × 512 pixels, and the aspect ratio is retained to reduce image distortion. In this process, the choice of image scaling algorithm is very important, because object detection tasks often require fine-grained information of the image. But some resize algorithms may lose some defects such as slight cracks. Therefore, we choose the pixel area relationship interpolation algorithm, as shown in Figure 2.

![Photovoltaic cell defects observed in EL images.](image-url)
4 | METHODOLOGY

4.1 | YOLO-PV

Object detection algorithm based on deep learning can be divided into two categories: one-step and two-step.\(^1\) A typical two-step object detector is R-cnn\(^2\) series. The algorithm first obtains candidate regions through region selection methods, then resizes the candidate regions to a fixed size and sends them to the CNN network to extract image features. Finally, SVM classifier is used to classify the extracted features, then get the classification results of each region. The two-step method has a very complex workflow. Model training is divided into several stages, and the steps are cumbersome: fine tuning CNN Network, training SVM, and training frame regressor. The Faster-RCNN\(^3\) proposed later solves the above problems to a certain extent, but there are still many redundant calculations.

Common one-stage detectors include YOLO,\(^4\) SSD,\(^5\) etc. In large production lines, processing speed is often as important as accuracy. In addition, with the application of UAV patrol inspection in power station, if the detection video can be processed in real time, the detection work will be more efficient. To match the production rate of production line and meet the needs of real-time detection, we give priority to speed and choose the YOLO algorithm. YOLO algorithm is the first one-stage detector in the field of deep learning. It reframes the object detection as a regression problem. It uses a single convolutional neural network to predict the position and category of the object directly from the image. This makes the YOLO algorithm extremely fast and can process streaming video in real time. After continuous iteration, the accuracy and speed of the YOLO algorithm have made great progress.

As mentioned above, the YOLO algorithm implements end-to-end detection, and the process is shown in Figure 3. The idea of the algorithm is to divide the image into \(S\times S\) grids. If the center of an object is located in this grid, then the grid is responsible for predicting that object. Each grid predicts \(B\) boxes and a confidence score. The confidence score means whether there is an object in the grid. So, the predicted bounding box has five parameters \((x, y, w, h, c)\), where \((x, y)\) represent the coordinates of the center point of the bounding box, and \((w, h)\) are the relative values of the width and height of the bounding box for the whole picture, and \(c\) is the confidence scores of each bounding box. At the same time, each grid also has a category score \(C\), and \(C\) represents the conditional class probabilities. Using one-hot code, if the data set has 20 categories, then \(C\) is a 20-dimensional vector. The output of the network is a vector of \(S\times S\times (B\times 5 + C)\).

4.1.1 | YOLO-PV structure

YOLO-PV is based on the YOLO v4\(^6\) algorithm, an object detection algorithm designed to detect defects in PV modules. Before designing the network, we first analyze the characteristics of the PV module defect detection task, which is summarized as follows:
1. Since the PV module is composed of cells arranged in an orderly manner, the defective cells to be detected will not overlap and block each other. Each cell is independent of the other, and the defective cell is often not related to the surrounding cells. So, there is no need for complicated semantic understanding and excessive global information.

2. In reality, the size of the cell is constant, so the size of the bounding box we predict is relatively fixed, which greatly reduces the error caused by the bounding box in the interference.

3. EL images are grayscale images, and the defects in the EL images are often low-level features (such as color, shape, and texture) without semantic information. This can reduce the acquisition of high-level semantic when designing the network structure.

On the basis of analysis of the above characteristics and application requirements, we present the YOLO-PV network structure, which may also be suitable for many industrial inspection tasks with relatively fixed targets (such as crack detection of ceramic tiles). Our network framework is the same as other mainstream object detectors and is divided into three parts: Backbone, Neck, and Head.

**Backbone**

In the object detection algorithm, the backbone part is used to extract the input features, which is the cornerstone of subsequent tasks. The processing time of backbone part accounts for more than half of the whole algorithm process to achieve high accuracy and fast detection. The backbone part must be redesigned and simplified. The major components of the backbone include CBM, Res_unit and CSPn modules.

**CBM:** This module contains a convolutional layer, a batch normalization layer, and a Mish activation function layer. The batch standardization layer is to alleviate overfitting during the training process.

**Res_Unit:** This module is a residual block, which is composed of two CBM and an addition module. It can maintain low complexity in deeper network layers and improve feature extraction ability.

**CSPn:** CY Wang et al. point out that CSPNet can reduce the amount of calculation by 20% while enhancing the learning ability of CNN, effectively reducing the memory usage. Therefore, referring to this study, we apply CSPNet to ResNet to form a CSPn module, where $n$ represents $n$ Res_units in the module.

**Neck**

The Neck part, as a connecting link between the Backbone and the Head plays an important role. It integrates the important features extracted by the backbone, which is conducive to the specific learning of the Head in the next step.

In the Neck part, we use the method of path aggregation. Common methods of path fusion are: FPN, PANet, and Bi-FPN. Compared with FPN, PANet and Bi-FPN have better performance, but the computational complexity increases. The Bi-FPN model is more complex than PANet and is difficult to model simplification. Considering that only one size of the feature map output is retained in the following work, we use PANet to fuse the semantic information in the multilayer feature map. The structure of PANet is shown in Figure 4. It takes level 3–5 input features, where $P_i$ represents a feature level with resolution of $1/2^i$ of the input images. First, different feature map inputs are obtained from backbone, and then the information of the feature maps is fused with each other. Finally, there are feature map outputs of different sizes in PANet, in which a small feature map size is used to detect large objects, and a large feature map size is used to detect small objects. Multiscale feature map output is very effective when the size of the detection target is not constant. However, the size and aspect ratio of the defective cell in this task are fixed. Therefore, we simplify the PAN structure and propose SPAN, which only retains one-size feature map output. It greatly reduces the amount of calculation of the PAN network.

The Neck of YOLO-PV includes CBM, CBL (including a convolution layer, a BN layer and a Leaky Relu activation function layer), and SPAN. In addition, to improve the model’s attention to spatial, we introduce the attention mechanism SAM layer.

**Head**

The head part uses the same head layer as YOLO V3. We use this layer to predict both class probabilities and bounding box coordinates.

Figure 5 shows the structure of YOLO-PV. Since the defect detection of PV modules does not involve complex...
semantic understanding and the complex relationship between target and background, we use four CSPN modules to extract image feature information in the backbone. The input image size is 960 × 512. After each CSPN module, the size of the feature map will be reduced by half. In the last three CSPN modules, 8, 16, and 32 times of downsampling are extracted. Three feature maps of different sizes represent the feature information of different levels. Through the Neck part, the understanding and learning of feature information are strengthened. The learned feature information is a single-size feature map. Then, the output can be obtained through the YOLO layer.

To verify the effectiveness of the YOLO-PV network structure on the PV module data set, in Section 5, we compare the effect of using different backbones to extract feature information and different Neck to fuse the feature information on the network performance.

4.1.2 | Loss function

In this work, we use the loss function of YOLO v4, which includes three parts of positioning loss, object confidence loss, and classification loss. The loss function calculation formula is given by Equation (1), where \( w, h, B \) are the width and height of the feature map and the number of boxes predicted by each point, \( C \) represents the confidence level, and \( p(c) \) represents the category prediction result. \( \text{1}_{\text{obj}} \) means that if the box at \( i,j \) has an object, its value is 1, otherwise, it is 0. \( \text{1}_{\text{noobj}} \) means that if the box at \( i \) and \( j \) has no object, its value is 1, otherwise it is 0. \( \lambda_{\text{iou}} \) is iou_normalizer, \( \lambda_{\text{cls}} \) is cls_normalizer, and \( \lambda_{c} \) is classes_multipliers. They are all adjustable hyperparameters. The confidence loss adopts MSE, and the classification loss part adopts the cross-entropy loss function. Positioning loss uses CIOU (Complete IoU) loss instead of MSE loss for bounding box regression. CIOU considers three more factors than IOU: overlap area, center point distance, and aspect ratio. Using Lciou makes the bounding box prediction more accurate.

\[
L = \lambda_{\text{iou}}L_1 + \lambda_{\text{cls}}L_2 - \sum_{i=0}^{w} \sum_{j=0}^{h} \text{1}_{\text{obj}} \sum_{c\in\text{class}} \lambda_{c}L_3 \quad (1)
\]

\[
L_1 = \sum_{i=0}^{w} \sum_{j=0}^{h} \text{1}_{\text{obj}} L_{\text{ciou}} \quad (2)
\]

\[
L_2 = \sum_{i=0}^{w} \sum_{j=0}^{h} \text{1}_{\text{obj}} \lambda_{c}(C_i - \hat{C}_i)^2 + \sum_{i=0}^{w} \sum_{j=0}^{h} \text{1}_{\text{noobj}} \lambda_{c}(C_i - \hat{C}_i)^2 \quad (3)
\]
4.1.3 Evaluation criteria

The most important evaluation index in the field of target detection is AP.\(^{30}\) The calculation of AP requires IOU, precision, recall rate, and other indicators. How to calculate these indicators will be introduced one by one. IOU is a standard to measure the accuracy between the predicted box and the ground truth, and the calculation of IOU is given by Equation (5). The larger the IOU value, the closer the prediction box and the ground truth are. When the IOUGP (IOU between the predicted and the ground truth) is greater than a certain threshold, we believe that the network has correctly detected the object. The calculation method of Precision and Recall is given by Equations (7) and (8). Among them, TP represents a correct detection of a ground-truth bounding, which is detection IOUGP ≥ threshold. FP represents a wrong detection (an incorrect detection of a nonexistent object, or a misplaced detection of an existing object), which is the detection IOUGP < threshold. FN represents the number of targets that have not been detected by the network. NTP, NFP, and NFN represent TP, FP, FN respective statistics.

\[
L_3 = \hat{p}(c)\log(p(c)) + (1 - \hat{p}(c))\log(1 - p(c)) \tag{4}
\]

Another important evaluation criterion called AP is calculated on the basis of Precision and Recall rate. This work uses F11-point interpolated average precision to calculate the AP. The specific steps are given as follows:

1. First, the algorithm outputs all the defect prediction boxes and sorts them according to the confidence level.
2. Set different values of \(K\), select the top \(k\) prediction boxes, calculate FP and TP, so that recall rate \(R\) is equal to 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0 respectively, and calculate the precision rate under different recall rates.
3. Take the average of the obtained 11 precision rates to get the value of AP.

4.2 Solving overfitting

When analyzing the actual experimental results, we find that the validation set loss will increase slightly during the later training. The performance AP of the model will decrease (see Section 5.1). We think that the model may have overfitting. It may be due to the difference of distribution between training set and validation set, which makes the model better in training set, but poor in validation set.

The methods to solve the problem include adding BN layer\(^{31}\) and Dropout layer,\(^{32}\) early stopping, and data augmentations. For the first two methods, we add BN layer and Dropout layer to the network during network design. We save the model weight in time during the training process, check the AP of the model in the training set, and use the best weight to interference. But these two methods change the symptoms instead of addressing the core of the problem. The essence of the overfitting problem is that the gap between the training set and the validation set. Therefore, collecting more data is the most effective solution, but we cannot obtain unlimited data due to condition. Increasing data diversity through data enhancement has been an effective way to make limited data generate greater value. Commonly used data enhancement methods include flip, rotate, adjust color temperature, exposure, etc. Unlike the image classification task, the object detection data set may cause image displacement during data enhancement, so the corresponding label must be adjusted. We choose to rotate the picture randomly and adjust the exposure randomly to enhance the data set, and we also use the mosaic method. Figure 6 shows the original image and the image after data enhancement. The impact of different data enhancement methods on network performance will be discussed in Section 5.

5 EXPERIMENTAL RESULTS AND DISCUSSION

This part introduces the environment, some hyperparameters and the data enhancement methods used in the experiment. This work uses deep learning frameworks TensorFlow and Darknet. Experimental operating system Ubuntu includes 18.04, CPU model i7-4790, GPU model GeForce RTX 2080 Ti. We need to set the learning rate, epochs, etc., at the start of the experiment. The values of these hyperparameters are given in Table 1.
5.1 Comparison of different backbones

In Section 4.1.1, we propose CSP-PV as the backbone to extract the feature information of EL image. In order to verify the effectiveness of the proposed model, we compare the performance of the object detection algorithm using VGG-16, Resnet50, Resnet101, Darknet53, CSPDarknet, and CSP-PV as backbone. To avoid the interference of other parts on the experimental results, neck and head in this experiment use the corresponding parts in YOLO v4. The experimental results are shown in Figure 7. The first six lines in Table 2 show the precision, recall, and the AP when using different backbones.

It can be seen from the experimental results that the best AP on the validation set is 90.7% when using CSP-PV as Backbone, which is an increase of 1.88% compared with the AP on the validation set by CSPDarknet. This may be because CSPDark is in-depth, and the feature extraction ability is stronger, so it can learn more deep information and global information. This information may be redundant for this task and may eventually affect the accuracy. Therefore, the network structure of deep learning is not as complicated as possible. It should be designed reasonably according to practical requirements. We also find that the model converges faster when CSP-PV is used as the backbone, and achieves the best AP at about the 2700th epoch. This shows that using CSP-PV as the backbone can reduce the depth and complexity of the model. So, it improves the performance and accelerates the convergence, thus saving the training time.

When VGG-16 is used as a feature extraction network, the loss is the largest, and the value of loss appears to get unstable the change with sharp fluctuations. This demonstrates that using VGG-16 may not be able to extract enough feature information to fulfill the requirements for object detection, and the robustness of the network is insufficient. However, the remaining five groups of experiments will have the problem that the loss on the validation set will increase in the later stage of the training. The AP will drop significantly. We suggest that this is probably because the model has been overfitted. The overfitting phenomenon may be due to too little data to learn enough information, or the network model is too complicated to learn some useless information (such as noise) in the training set. Comparing the experimental results of Resnet50 and Resnet101 shows that the more complex the model, the more serious the overfitting phenomenon. To overcome the problem of overfitting, we use different data enhancement methods in Section 5.3.

5.2 Comparison of different necks

The PAN has multiple sizes of feature map output, responsible for predicting objects of different sizes. In SPAN, we only keep one-size feature map output, which greatly reduces the computation cost. In addition, we introduce spatial attention layer in the Neck part of YOLO-PV to enhance the model’s acquisition of spatial information. In this section, we use YOLO-PV as the object detection algorithm to experiment on the data set, and the experimental results are shown in Figure 8. We compare it with the loss and AP of CSP-PV as the backbone in Section 5.1.
(the two differ only in the Neck part). Through contrastive analysis, we can conclude that the AP using the YOLO-PV structure reaches 91.34%, which is an increase of 0.64% compared with the CSP-PV group in Section 5.1. This result shows that the effect of multiscale prediction will not be qualitatively improved when the object size is relatively fixed. Using a single-size feature map can slightly improve AP, and at the same time, greatly reduce the number of output feature maps to avoid the waste of computing resources. YOLO-PV makes the model faster by reducing the complexity of the feature extraction network and removing the redundant feature map output. So, it takes only 28 ms to process an EL image, which reduces the processing speed by 36.36% compared with 44 ms for YOLO v4.

5.3 | Comparison of different data augmentation

In the above experiment, we find that the model has an overfitting problem. In response to this problem, three different data enhancement methods are used to solve the overfitting phenomenon in the training process in this section. The three methods are random rotation, random adjustment of exposure, and mosaic method. To verify whether the three methods are effective, we first apply the
three methods separately, and the experimental results are shown in Figure 8. After using the data enhancement method, the overfitting phenomenon of the model is alleviated, but not all data enhancement methods will bring positive effects.

As shown in Table 3, when we randomly rotate the image, the AP of the model drops by 1.6%. This may be because there are no rotated module images in the validation set. Adding this situation to the training set will make the gap between the training set and the validation set larger. For the model, although more features are learned, these features do not belong to the validation set, thus affecting the model's acquisition of effective information. However, the method of rotating images may be helpful to some data sets that contain pictures from various shooting angles. The AP of the model will increase by 1.52% when the exposure is adjusted randomly. This may be because, when the exposure is increased, some unobvious defects, such as cracks and broken grids may be more obvious, which is conducive to the defect feature extraction of the model. What is more, adjusting the exposure can also simulate the situation of inconsistent picture brightness caused by the unfixed input current and exposure time. It greatly enriches the diversity of data sets and makes the model more stable. Using the mosaic data enhancement method can increase the AP of the model the most, thus increasing the AP by 2.26%. The mosaic data enhancement method is to splice four pictures together to generate a new picture. This method can greatly enrich the background of the target in the object detection task. But in this experiment, the EL image's background is the same, so we think this effect can be produced because the mosaic method synthesizes the new EL images. These new EL images are equivalent to expanding the number of EL images in the training set. More images are involved in the training, which can improve the performance of the model.

Therefore, we apply random exposure adjustment and mosaic to the training process at the same time. Through the results of Figure 8, we can see that after using the random adjustment of exposure and the mosaic method, the convergence speed of the model becomes slower, but there is no obvious overfitting phenomenon in the training process. The value of the loss function decreases with the increase of training times and finally stabilizes. AP increases by 3.42%–94.76%.

5.4 Model performance on test set

To verify the performance of the model, we test the performance of the model on the test set. Table 4 shows the AP of the model on the training set, verification set, and test set. The effect of YOLO-PV on the test set decreased slightly within an acceptable range to 94.55% compared with the verification set. The model still maintains its characteristics of high accuracy. Figure 9 shows the test set results of object detection after using YOLO-PV and data enhancement methods. The model not only accurately identify the defects, such as break cell and black cell, but also identify some very subtle defects such as cracks less
than 10mm. These defects are very difficult to be detected with the naked eyes. However, the model is not robust in identifying some subtle finger failure and slight low cell. In practice, missed inspections may occur. For finger failure, some of finger failure features are not obvious and are easily mixed with the background (such as regular black lines). For low cells, some slight low cells are not significantly different from normal cells in the modules, causing missed inspection.

6 | CONCLUSION

Fast and high-accuracy object detection algorithm is an important work because detection speed and detection accuracy are equally important on PV module production line. Therefore, in this paper, a fast object detection algorithm for the EL image of PV modules is proposed based on YOLO v4 algorithm, which can quickly and accurately identify the defective cells in PV modules. After the completion of the network design, we have carried out experiments on EL image data sets. The experimental results show that compared with the YOLO v4 algorithm, the Precision, Recall rate and AP are improved. The processing time reduces 36.36%, which meets the requirements of industrial detection speed and accuracy. The algorithm can be better applied to the working scene with only CPU and mobile terminal. In addition, the structure of the algorithm can be used for other industrial detection tasks with relatively fixed size and independent objects.

This paper uses three different data enhancement methods, analyzes the actual effect on the EL image, and explains the performance improvement and possible application scenarios. In the next, we can extend this work in several directions. A more concise backbone structure and more efficient feature fusion method to further improve the speed and efficiency must be used. In addition,
the defect classification task is not involved in this work. We can use a more novel backbone extraction network to extract deeper network information without increasing the complexity of the model or use the currently extracted feature information to complete a more complex object detection task. At the same time, for the data set, it can be seen from the experimental results that data enhancement can well improve the performance of the network model in the case of limited data. However, this work only explores three data enhancement methods, which can explore more effective data enhancement methods in future work.

There are still many problems in the research of EL image, such as the secondary use of the results in this work. In addition, there is still no clear relationship between the defect problems reflected in EL images and the actual performance. The algorithm for quantitative analysis of EL defects has not been deeply studied. Future research should focus on more accurate and faster identification of defective solar cells. As well as the algorithm detection results should be effectively used to quantify the performance of solar cells.

ORCID
Ziyao Meng © https://orcid.org/0000-0002-0541-5868

REFERENCES
1. Lillo-Bravo I, González-Martínez P, Larrañeta M, Guasumbacodena J. Impact of energy losses due to failures on photovoltaic plant energy balance. Energies. 2018;11(2):363. doi:10.3390/en11020363
2. Köntges M, Kajari-Schröder S, Kunze I, Jahn U. Crack statistic of crystalline silicon photovoltaic modules. European Photovoltaic Solar Energy Conference; 2011.
3. Paggi M, Berardone I, Infuso A, Corrado M. Fatigue degradation and electric recovery in Silicon solar cells embedded in photovoltaic modules. Sci Rep. 2015;4(1):1-7. doi:10.1038/srep04506
4. Köntges M, Kurtz S, Packard C, Jahn U, Friesen G. Review of failures of photovoltaic modules; 2014.
5. Deitsch S, Christlein V, Berger S, et al. Automatic classification of defective photovoltaic module cells in electroluminescence images. Sol Energy. 2019;185:455-468. doi:10.1016/j.solener.2019.02.067
6. Akram MW, Li G, Jin Y, et al. CNN based automatic detection of photovoltaic cell defects in electroluminescence images. Energy. 2019;189:116319. doi:10.1016/j.energy.2019.116319
7. Tang W, Yang Q, Xiong K, Yan W. Deep learning based automatic defect identification of photovoltaic module using electroluminescence images. Sol Energy. 2020;210:453-460. doi:10.1016/j.solener.2020.03.049
8. Yiping Z. EL image recognition of solar cell module defects based on deep learning; 2020.
9. Liu L, Zhu Y, Ur Rahman MR, Zhao P, Chen H. Surface defect detection of solar cells based on feature pyramid network and GA-faster-RCNN. 2019 2nd China Symposium on Cognitive Computing and Hybrid Intelligence (CCHI), Cognitive Computing and Hybrid Intelligence (CCHI); 2019.
10. Zhang X, Hao Y, Shangguan H, et al. Detection of surface defects on solar cells by fusing multi-channel convolution neural networks. Infrared Phys Technol. 2020;108:103334.
11. Otamendi U, Martínez I, Quartulli M, et al. Segmentation of cell-level anomalies in electroluminescence images of photovoltaic modules. Sol Energy. 2021;220:914-926.
12. Takahashi Y, Kaji Y, Ogane A, Uraoka Y, Fuyuki T. “Luminoscopy”- novel tool for the diagnosis of crystalline silicon solar cells and modules utilizing electroluminescence. 2006 IEEE 4th World Conference on Photovoltaic Energy Conversion, Photovoltaic Energy Conversion, Conference Record of the 2006 IEEE 4th World Conference On, Vol. 1; 2006:924-927. doi:10.1109/WCPEC.2006.279607
13. Fuyuki T, Kaji Y, Ogane A, Takahashi Y. Analytic findings in the photographic characterization of crystalline silicon solar cells using electroluminescence. 2006 IEEE 4th World Conference on Photovoltaic Energy Conversion, Photovoltaic Energy Conversion, Conference Record of the 2006 IEEE 4th World Conference On, Vol. 1; 2006:905-907. doi:10.1109/WCPEC.2006.279602
14. Würfel P, Trupke T, Rüdiger M, et al. Diffusion lengths of silicon solar cells from luminescence images. J Appl Phys. 2007;101(12):1650. doi:10.1063/1.2749201
15. Köntges M, Kurtz S, Packard C, et al. Review of failures of photovoltaic modules; 2014.
16. Zou Z, Shi Z, Guo Y, Ye J. Object detection in 20 years: a survey; 2019.
17. Girshick R, Donahue J, Darrell T, Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation. 2014 IEEE Conference on Computer Vision and Pattern Recognition, Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference On; 2014:580-587. doi:10.1109/CVPR.2014.81
18. Girshick R. Fast R-CNN. Proceedings of the IEEE International Conference on Computer Vision, 2015 International Conference on Computer Vision, ICCV 2015; n.d.:1440-1448. doi:10.1109/ICCV.2015.169
19. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: unified, real-time object detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference On; 2016:779-788. doi:10.1109/CVPR.2016.91
20. Liu W, Anguelov D, Erhan D, et al. SSD: single shot multibox detector[J]. European Conference on Computer Vision; 2016.
21. Bochkovskiy A, Wang CY, Liao HY. Yolov4: optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934; 2020.
22. Wang C-Y, Mark Liao H-Y, Wu Y-H, Chen P-Y, Hsieh J-W, Yeh I-H. CSPNet: a new backbone that can enhance learning capability of CNN. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Computer Vision and Pattern Recognition Workshops (CVPRW), 2020 IEEE/CVF Conference On; 2020:1571-1580. doi:10.1109/CVPRW.50498.2020.0020323
23. Misra D. Mish:a self regularized non-monotonic activation function. arXiv preprint arXiv:1908.08681; 2019.
24. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference On; 2016:770-778. doi:10.1109/CVPR.2016.90

25. Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2017:2117-2125.

26. Liu S, Qi L, Qin H, Shi J, Jia J. Path aggregation network for instance segmentation. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Computer Vision and Pattern Recognition (CVPR), 2018 IEEE/CVF Conference on, CVPR; 2018:8759-8768. doi:10.1109/CVPR.2018.00913

27. Tan M, Pang R, Le QV. EfficientDet: scalable and efficient object detection. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Computer Vision and Pattern Recognition (CVPR), 2020 IEEE/CVF Conference on, CVPR; 2020:10778-10787. doi:10.1109/CVPR42600.2020.01079

28. Woo S, Park J, Lee J-Y, Kweon IS. CBAM: convolutional block attention module. Computer Vision – ECCV 2018 : 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part VII, 3; 2018. doi:10.1007/978-3-030-01234-2_1

29. Zheng Z, Wang P, Liu W, Li J, Ye R, Ren D. Distance-iou loss: faster and better learning for bounding box regression. arXiv; 2019. doi:10.1609/aaai.v34i07.6999

30. Padilla R, Netto SL, da Silva EAB. A survey on performance metrics for object-detection algorithms. 2020 International Conference on Systems, Signals and Image Processing (IWSSIP); 2020. ISBN 9781728175393. doi:10.1109/iwssip48289.2020.9145130

31. Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. Proceedings of the 32nd International Conference on Machine Learning, in Proceedings of Machine Learning Research, Vol. 37; 2015:448-456. http://proceedings.mlr.press/v37/ioffe15.html

32. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res. 2014;15(6):1929-1958.

33. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556; 2014.

34. Redmon J, Farhadi A. Yolov3: an incremental improvement. arXiv preprint arXiv:1804.02767; 2018.

How to cite this article: Meng Z, Xu S, Wang L, Gong Y, Zhang X, Zhao Y. Defect object detection algorithm for electroluminescence image defects of photovoltaic modules based on deep learning. Energy Sci Eng. 2022;10:800–813. doi:10.1002/ese3.1056