CCA-YOLO: Channel and Coordinate Aware-Based YOLO for Photovoltaic Cell Defect Detection in Electroluminescence Images

Junqi Bao, *Graduate Student Member, IEEE*, Xiaochen Yuan, *Senior Member, IEEE*,
Qingying Wu, *Graduate Student Member, IEEE*, Chan-Tong Lam, *Senior Member, IEEE*,
Wei Ke, *Member, IEEE*, and Ping Li, *Member, IEEE*

***Abstract*— Solar energy is a renewable energy used for urban power generation, contributing to sustainable cities. In solar energy generation, it is important to inspect the health of photovoltaic (PV) cells for safety and power transformation efficiency. Defects in PV cells are usually irregular with different scales, challenging automated defect detection for PV cells. Therefore, this article presents a channel and coordinate aware-based YOLO (CCA-YOLO) for efficient PV cell defect detection. Specifically, to provide accurate backbone features from the complex background defect images, the residual coordinate convolution-based ECA (RCC-ECA) enhances the backbone feature representation by learning from channel and coordinate information. To learn the intraclass/interclass variations and interclass similarity and convey coordinate information among different scales, the multiscale defect feature localization module (MDFLM) incorporates a larger backbone feature to improve the robustness of multiscale defects. The RCC-Up/Down optimizes the sampled features to minimize the inaccurate representation of the features caused by the sampling process. In addition, RCC-Up/Down conveys the coordinate information during the up/down sampling process to maintain coordinate awareness, which allows the network to learn from the coordinate information efficiently. Furthermore, the residual feature fusion with coordinate convolution-based CBAM (RFC-CBAM) is introduced to maintain the channel and coordinate awareness for efficient learning from fused features. The proposed CCA-YOLO outperforms state-of-the-art (SOTA) methods in PVEL-AD on precision (71.71%), recall (76.91%), $F1$-Scores (74.19%), mAP$_{50}$ (98.57%), AP$_S$ (26.80%), AP$_M$ (64.78%), and AP$_L$ (74.93%).**

***Index Terms*— Convolutional neural networks, defect detection, electroluminescence images, photovoltaic (PV) cell.**

I. INTRODUCTION

SOLAR power is a renewable energy source, and presently, the use of solar power for electricity generation is one of the most extensively employed methods. Multicrystalline photovoltaic (PV) cells play a significant role in solar power systems. However, defects in PV cells can occur during the manufacturing process or due to environmental factors, which may impede the conversion efficiency of solar power and reduce the lifespan of the PV cell. Thereafter, it is valuable to detect and classify the defects in PV cells. In industry, this process can provide higher safety and more efficient operation of PV cell power stations.

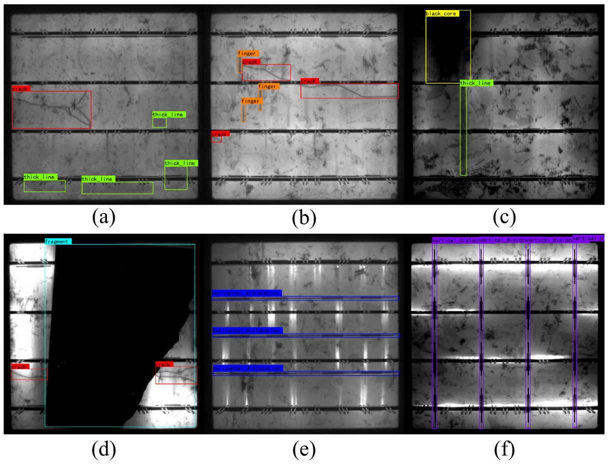


Fig. 1. Example of defects from PVEL-AD dataset. (a) *Crack* (red box) and *Thick line* (green box). (b) *Finger* (orange box) and *Crack*. (c) *Black core* (yellow box) and *Thick line*. (d) *Crack* and *Fragment* (dark slate gray box). (e) *Horizontal dislocation* (blue box). (f) *Vertical dislocation* (purple box).

Electroluminescence imaging is a nondestructive, near-infrared technology that can image PV cells at a high resolution. Fig. 1 demonstrates an example of defects in PV cells' EL images from the PVEL-AD dataset [1]. Although the EL image presents a clear vision of each defect, various interferences between defects bring severe challenges to defect detection on PV cell EL images.

1) *Interclass Difference in Scales and Shapes:* As demonstrated in Fig. 1(d), there are multiple-class defects (*Crack* and *Fragment*) that exist in a single EL image. *Crack* usually presents an irregular linear structure with darker colors while *Fragment* usually is a large black area with a clear border. The scales and the shapes of *Crack* and *Fragment* have significant differences, which makes it difficult to learn and distinguish them.

Received 13 September 2024; revised 3 December 2024; accepted 1 January 2025. Date of publication 13 February 2025; date of current version 26 February 2025. This work was supported in part by the Science and Technology Development Fund of Macau SAR under Grant 0045/2022/A and in part by the Macao Polytechnic University under Grant RP/FCA-12/2022. The Associate Editor coordinating the review process was Dr. Shiqian Chen. *(Corresponding author: Xiaochen Yuan.)*

Junqi Bao, Xiaochen Yuan, Qingying Wu, Chan-Tong Lam, and Wei Ke are with the Faculty of Applied Sciences, Macao Polytechnic University, Macau, SAR, China (e-mail: junqi.bao@mpu.edu.mo; xcyuan@mpu.edu.mo; qingying.wu@mpu.edu.mo; ctlam@mpu.edu.mo; wke@mpu.edu.mo).

Ping Li is with the Department of Computing and the School of Design, The Hong Kong Polytechnic University, Hong Kong, SAR, China (e-mail: p.li@polyu.edu.hk).

Digital Object Identifier 10.1109/TIM.2025.3541805

2) *Intraclass Variations in Scales and Shapes: Thick Line* defects usually consist of multiple lighter-colored columns as demonstrated in Fig. 1(a) and (c). Although the structures of different *Thick Line* are similar, there are variations in shapes and scales between different defects. In addition, the *Crack* as illustrated in Fig. 1(a), (b), and (d) also present variations in scales and shapes.
3) *Interclass Similarity:* As demonstrated in Fig. 1(c) and (d), *Fragment* and *Black Core* gain high similarity since the difference between them is *Fragment* has more clear and regular borders. The interclass similarity makes the network harder to classify similar defects precisely.
4) *Bounding Box Overlapping:* As demonstrated in Fig. 1(d), the overlapping exists between the bounding box of *Fragment* and *Crack*, which brings challenges for the network to learn efficient and accurate features of these overlapped defects.
5) *Location-Sensitive:* Horizontal Dislocation and *Vertical Dislocation* usually present as a linear-like horizontal or vertical-distributed defects. However, the only difference between these two kinds of defects is the location distribution. Therefore, it is important for the network to learn the distribution of the defects.

For the automated detection of defects in PV cell EL images, many state-of-the-art (SOTA) methods have been proposed for defect detection. The handcrafted feature-based methods usually extract features to represent the texture, color, and shape of the EL images and use the data-based classifier to detect defects. Su et al. [2] introduce a novel feature descriptor that can obtain stronger discriminatory local defect information in EL images. In addition, they calculate the descriptors in EL images to train and test the discriminant classifiers, such as support vector machine (SVM) and nearest neighbor classifier (NNC). Xu et al. [3] propose a sigmod-Hough-transform-based geometric segmentation method to extract the defect region in EL images. Also, they propose a self-comparison method to detect defects in the background with nonuniform luminance and complicated texture.

In recent years, using convolutional neural networks to detect defects has received lots of interest in the industry. Chen and Jahanshahi [4] incorporate naive Bayes data fusion and CNN for crack detection of nuclear power plant components. Su et al. [5] proposed a BAF-Detector that introduces a bidirectional attention feature pyramid network (BAFPN) into Faster R-CNN [6]. Chen and Jahanshahi [7] proposed a deep fully convolutional network for real-time crack detection in video. They introduce a naive Bayes-based data fusion into the fully convolutional network, which significantly improves the efficiency and capability of video crack detection. Bao and Yuan [8] proposed a YOLO-based PV cell defect detection method in five categories, which incorporate an improved CBAM for YOLO. Qin et al. [9] proposed an efficient defect detection network (EDDNet) that provides spatial information on defects efficiently by a spatial attention mechanism. In addition to CNN-based approaches, Carion et al. [10] proposed a detection transformer (DETR), which presents an end-to-end method for object detection using transformer architecture. Su et al. [11] proposed a prior-modulated and semantic-aligned dynamic transformer (PMSA-DyTr) which introduces a long short-term self-attention mechanism to eliminate noise for the defect images and prior-modulated cross attention to make the transformer suit the small-scale defects datasets. Zheng et al. [12] proposed the Focus-DETR for efficient object detection. They focus attention on more informative tokens, which balance accuracy and efficiency. Du et al. [13] proposed an AFF-Net based on YOLO architecture and Swin Transformer [14] for strip steel surface defect detection. Zhang et al. [15] proposed an LDD-Net based on YOLO architecture for lightweight printed circuit board defect detection.

Across these defect detection methods, the attention mechanism plays an important role in allowing the network to learn efficient information from the features. Efficient channel attention (ECA) [16] reallocates the channel weights to achieve significant performance improvement with only a small number of parameter increments. However, ECA lacks the ability to handle global context dependencies and spatial relationships. Furthermore, the convolution block attention module (CBAM) [17] is a lightweight attention mechanism that refines the features from both channel and spatial aspects. Due to the additional parameters caused by the CBAM, it runs the risk of overfitting, especially when dealing with small or imbalanced datasets in the industry. In addition, as the number of defect categories increases, the difficulty of defect detection using deep learning techniques also increases. Existing methods such as [18] only detect three categories of defects on PV cell EL images. Therefore, [1] discussed the eight-categories defect detection performance of several SOTA methods on PVEL-AD [1] dataset where the experimental results indicated that these methods lack generalization ability in multicategories defect detection for PV cells EL images. Additionally, some researchers have achieved 12-categories defect detection in the PVEL-AD dataset recently such as [19] and [20], but these methods still struggle with the identification and localization of few-shot categories. Moreover, the generalization ability of their network in other defect detection datasets has not been validated.

Consequently, we propose a channel and coordinate aware-based YOLO (CCA-YOLO) for 12-categories defect detection for PV cells EL images. Specifically, we adopt YOLOv4 as the baseline network and propose four novel modules: residual coordinate convolution-based ECA (RCC-ECA), residual feature fusion with coordinate convolution-based CBAM (RFC-CBAM), multiscale defect feature localization module (MDFLM), and RCC-Up/Down for multiscale defect detection. The RCC-ECA efficiently improves the channel and coordinates awareness by making the network learn from both channel and coordinate aspects. Therefore, we adopt RCC-ECA to enhance the backbone feature representation, making the backbone feature more sensitive to the structure and distribution of the defects. Moreover, we propose an RCC-Up/Down based on RCC-ECA and MDFLM for efficient feature fusion. The RCC-Up/Down optimizes the sampled features to minimize the inaccurate representation of the features caused by the sampling process. In addition, RCC-Up/Down conveys the

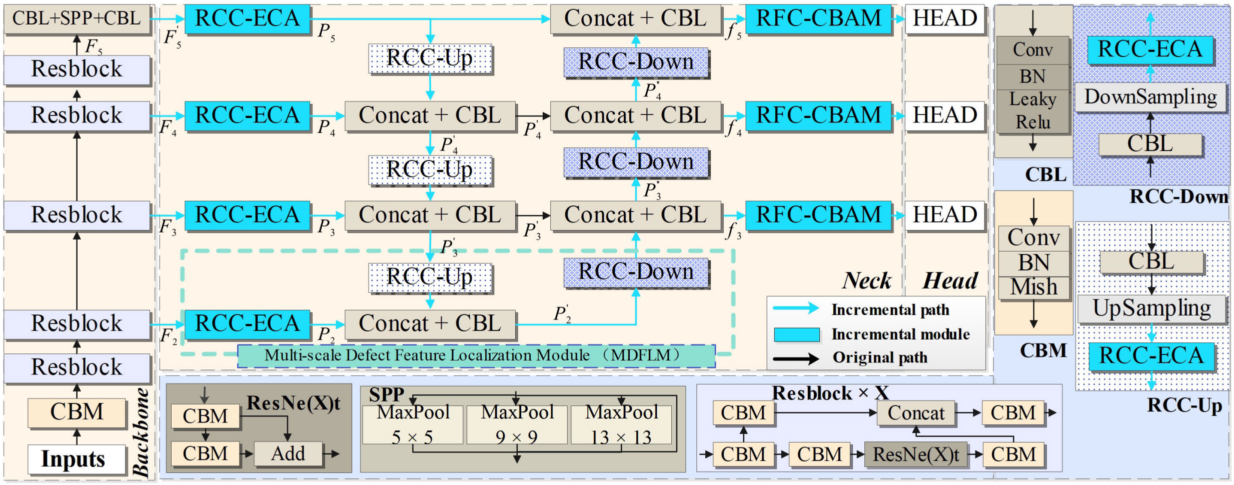


Fig. 2. Architecture of CCA-YOLO. The proposed CCA-YOLO is based on conventional YOLOv4, but we further propose RCC-ECA, MDFLM, and RFC-CBAM to enhance the feature fusion process, thereby enhancing the feature representation in feature fusion to improve the multiscale defect detection performance. In addition, we incorporate an RCC-Up/Down in Neck based on our RCC-ECA for up/down sampling. We use blue blocks to represent our proposed modules and a blue arrow to highlight the new path incorporating these modules, distinguishing it from the conventional YOLOv4 architecture. The RCC-ECA is first adopted for backbone feature refinement. Moreover, RCC-Up/Down conveys the coordinate information during the feature fusion by incorporating RCC-ECA into the sampling process. Furthermore, MDFLM is proposed to introduce a large-scale feature map for fusion, which enhances the robustness of the network to multiscale defects. In addition, the RFC-CBAM is utilized to refine the output feature of feature fusion to achieve better performance for the bounding box regression and defect classification.

coordinate information during the up/down sampling process to maintain coordinate awareness, which allows the network to learn from the coordinate information efficiently. The MDFLM incorporates a larger backbone feature for feature fusion, which improves the robustness of multiscale defects by providing more detailed information such as the structure and texture of the defects. Furthermore, RFC-CBAM is adopted to maintain the channel and coordinate information and to learn from the refined features for accurate prediction by the feature fusion-based channel attention module (FF-CAM) and the coordinate convolution-based spatial attention module (CC-SAM). The contributions of this article are summarized as follows.

1) We propose a CCA-YOLO for defect detection on PV cell EL images, allowing the network to efficiently learn defect features from the channel and coordinate information of the features. Data augmentation is applied to the PVEL-AD dataset to address the data imbalance among 12 categories, thereby enhancing the generalization ability of the network. Experimental results demonstrate that CCA-YOLO can handle multiscale defects of PV cells in 12 categories and outperform the SOTA methods. In addition, performance on the NEU-DET and PCB datasets verifies and validates the generalization ability of CCA-YOLO.
2) A RCC-ECA is adopted to enhance the network awareness of channel and coordinate information, which is beneficial for accurate feature representation. Due to the complex background of the defect images in the industry, we adopt RCC-ECA to enhance the backbone feature representation at four scales, making the backbone features more sensitive to the structure and distribution of the defect features. In addition, an RCC-Up/Down is introduced to convey the coordinate information and optimize the sampled features to minimize the inaccurate feature representation caused by the sampling process.
3) A RFC-CBAM is proposed to refine the fuse features for efficient learning from the fused features. Specifically, FF-CAM fuses the output features of average and max pooling layers to make full use of the channel information and avoid information loss. Meanwhile, CC-SAM provides additional coordinate information, which is beneficial for locating defect features from the fused features.
4) Since low-dimensional features usually contain more detailed information, we incorporate an MDFLM to enhance the ability of the network to locate the defect features. In CCA-YOLO, MDFLM provides larger features to the feature fusion path, thus enhancing the multiscale feature representation.

This article is organized as follows. Section II gives a detailed explanation of the proposed architecture and modules. Section III presents the experimental results and ablation studies. Finally, Section IV gives a conclusion to this article.

II. PROPOSED CCA-YOLO

A. Architecture of CCA-YOLO

Fig. 2 demonstrates the architecture of the proposed CCA-YOLO. Our CCA-YOLO is based on YOLOv4, but we further propose an RCC-ECA, MDFLM, and RFC-CBAM to enhance the feature representation ability, thereby meeting the requirement for multiscale defect detection in PV cell EL images. To demonstrate the difference between CCA-YOLO and YOLOv4, we use blue blocks to represent our proposed modules and a blue arrow to highlight the new path incorporating these modules, distinguishing it from the conventional YOLOv4 architecture. First, given an input image, the backbone extracts multiscale backbone features. Before the feature fusion, the proposed RCC-ECA is adopted to refine these backbone features of different scales through channel and spatial information. Then, the coordinate information is further conveyed from the backbone features to enhance coordinate awareness by the proposed RCC-ECA during the feature fusion process. Thereafter, the proposed RFC-CBAM is adopted to further refine the fused features of different scales and three YOLO Head predict the categories and bounding box of the defects using the refined features.

1) Backbone: Given an input image $I \in \mathbb{R}^{C\times H\times W}$, the CSPDarkNet [21] backbone extracts multiscale feature maps $F \in \mathbb{R}^{C_i\times(H/2^i)\times(W/2^i)}(i = 1, 2, 3, 4, 5)$, where the channel dimension are 64, 128, 256, 512, 1024. Then, a spatial pyramid pooling (SPP) is adopted to the feature $F_5 \in \mathbb{R}^{1024\times H\times W}$ to aggregate different scales of feature maps, thereby improving the detection capabilities for defects of different scales.

2) Neck: The Neck focuses on making full use of the backbone features for defect detection by aggregating the features of different scales. We follow the conventional architecture of the Neck in YOLOv4 [22] and we further propose three novel modules RCC-ECA, RFC-CBAM, and MDFLM to enhance the feature fusion. We first adopt RCC-ECA to enhance the backbone feature representation by enhancing the channel and coordinate awareness at four different scales of features: F_2, F_3, F_4, and F_5', respectively. In addition, we propose an RCC-Up and RCC-Down which incorporate the RCC-ECA into the up/down sampling process. Moreover, we proposed an MDFLM to introduce a new path of feature fusion into the Neck. Last, an RFC-CBAM is adopted to refine the fused features for detection at three different scales of features: f_3, f_4, and f_5. Therefore, during the feature fusion in Neck, the backbone features $F_i(i = 2, 3, 4)$ and F_5' will be further refined by RCC-ECA. Thereafter, the RCC-ECA refined feature P_5 will be upsampled by RCC-Up and then concatenated with the refined feature P_4. The fused features P_4' will further be upsampled by RCC-Up and then be concatenated with P_3. Furthermore, compared to the conventional YOLOv4, the proposed MDFLM further introduces a new path of feature fusion by aggregating the fused features P_3' with a large feature P_2. In addition, P_2' will go through an RCC-Down to be aggregated with P_3' to obtain the fused features f_3. Similarly, the fused features f_4 and f_5 will be obtained by aggregating P_3'' with P_4' and P_4'' with P_5. Finally, the RFC-CBAM integrates the channel and spatial information of the fused features to ensure that the network can accurately capture the categories and locations of defects.

3) Head: Three YOLO Heads are used to detect the defect. Furthermore, the loss function of the proposed CCA-YOLO is defined as (1), where IoU represents the intersection over union between ground truth (GT) and the predicted bounding box, d is the distance between GT and the predicted bounding box, c is the diagonal length of the smallest enclosing box covering the two boxes, and v is used to measure the relationship between the width-to-height ratio $W_{\mathrm{GT}}/H_{\mathrm{GT}}$ of GT and the width-to-height ratio W/H of the predicted bounding box as defined in (2)

$$L_{\mathrm{cIoU}} = 1 - \mathrm{IoU} + \frac{d^2}{c^2} + \frac{v^2}{(1-\mathrm{IoU}) + v} \tag{1}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{W_{\mathrm{GT}}}{H_{\mathrm{GT}}} - \arctan\frac{W}{H}\right)^2. \tag{2}$$

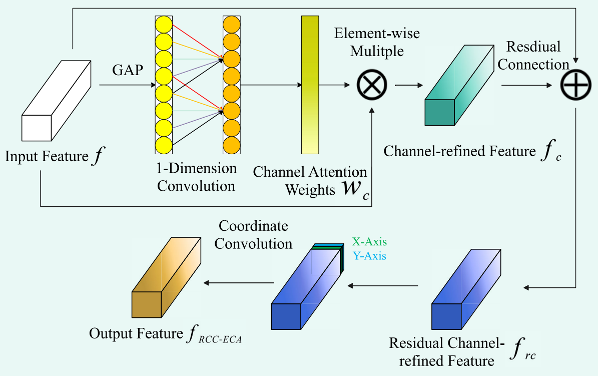


Fig. 3. Architecture of RCC-ECA. The channel attention weights are obtained by a GAP and a 1-D convolution. Subsequently, a residual connection is incorporated to maintain efficiency. Thereafter, the spatial awareness is further enhanced by coordinate convolution.

B. Residual Coordinate Convolution-Based ECA (RCC-ECA)

In this article, RCC-ECA is proposed to efficiently enhance the channel and coordinate awareness of the network. Specifically, we adopt RCC-ECA to enhance the backbone feature representation, making the backbone features more sensitive to the structure and distribution of the defects. In addition, RCC-ECA is introduced into up/down sampling in feature fusion by RCC-Up/Down to convey the coordinate information and minimize the inaccurate feature representation caused by the sampling process.

The architecture of the proposed RCC-ECA is demonstrated in Fig. 3. Specifically, RCC-ECA starts by applying global average pooling (GAP) on the input feature f, adapting the dimensionality of the input features to $1 \times 1 \times C$. Then, a 1×1 convolution and a sigmoid activation function are used to generate the channel attention weights. Consequently, the channel-refined feature f_c is generated by an element-wise multiple between the input features and the channel attention weights. Additionally, to avoid gradient vanishing, we further introduce a residual connection between the input feature and the channel-reallocated feature f_c. Furthermore, since the types of defects on PV cells are related to their location on the PV cells, we incorporate coordinate convolution operations [23] to provide additional coordinate information for the extracted backbone features. In summary, the reallocated features $f_{\text{RCC-ECA}}$ by RCC-ECA are generated as follows:

$$\begin{cases} \omega_c = \mathrm{GAP}(\mathrm{Conv}_{1D}(f)) \\ f_c = f \otimes \omega_c \\ f_{\mathrm{rc}} = f + f_c \\ f_{\text{RCC-ECA}} = \mathrm{CoordConv}(f_{\mathrm{rc}}) \end{cases} \tag{3}$$

where ω_c is the channel attention weights, GAP represents the GAP layer, Conv_{1D} represents the 1-dimensional convolution, and $\otimes$ is the element-wise multiplication. In addition, f is the input feature, f_c is the channel-reallocated feature, f_{rc} is the channel-reallocated feature after residual connection, and $f_{\text{RCC-ECA}}$ is the final output feature after coordinate convolution.

Consequently, RCC-ECA improves the channel and coordinates awareness of the network, allowing a better feature representation for efficient learning. Introducing RCC-ECA to the backbone features significantly reduces the noise of the complex background of the defect images and enhances the backbone feature representation to capture the structure and distribution of the defects. Moreover, RCC-Up/Down conveys the coordinate information during the feature fusion process, which enhances the network to learn from the coordinate information.

C. Multiscale Defect Feature Localization Module

Considering the significant variances in shape, size, and location of defects across different types of PV cells, it is imperative for the network to focus on accurately identifying and distinguishing these diverse defects. Since conventional YOLOv4 only uses F_3, F_4, and F_5 for feature fusion and defect detection, the ability of the conventional YOLOv4 to locate multiscale defects has been limited. Therefore, we propose an MDFLM to further incorporate a large feature F_2 into the conventional feature fusion process.

Specifically, the feature F_2 will be refined by RCC-ECA first. Simultaneously, the up-bottom path aggregated feature P_3' will be upsampled by RCC-Up and then be concatenated with P_2. Subsequently, the concatenated feature P_2' will be further downsampled by RCC-Down and then concatenated with P_3' to obtain the fused feature f_3. Since a large size of feature maps usually contains more detailed image information, incorporating a larger feature F_2 with other three smaller sizes can significantly help the network to locate the multiscale defects, which can enhance the robustness of the network to multiscale defects.

D. Residual Feature Fusion With Coordinate Convolution-Based CBAM

In this article, we propose RFC-CBAM to refine the fused features by maintaining the channel and coordinate awareness from the fused features for efficient learning and prediction.

Fig. 4(a) demonstrates the architecture of the proposed RFC-CBAM. Specifically, given an input feature map $f_{\text{in}} \in \mathbb{R}^{W \times H \times C}$, we first calculate 1-dimension channel attention $\omega_c \in \mathbb{R}^{1\times 1\times C}$ by FF-CAM. By an element-wise multiple between f_{in} and ω_c, the channel-refined feature f_c is obtained as denoted in (4). Subsequently, a CC-SAM is applied to f' to calculate a 2-D spatial attention map $\omega_s \in \mathbb{R}^{W\times H\times 1}$ by CC-SAM. Thereafter, the spatial-refined feature f_s is generated by multiplying f_c with ω_s as denoted in (5). Furthermore, a residual connection is incorporated to generate the output feature $f_{\text{RFC-CBAM}}$ as denoted in (6)

$$f_c = \omega_c \otimes f_{\text{in}} \tag{4}$$

$$f_s = \omega_s \otimes f_c \tag{5}$$

$$f_{\text{RFC-CBAM}} = f_s + f_{\text{in}} \tag{6}$$

where $\otimes$ denotes element-wise multiplication, f_c is the channel-refined feature, ω_c is the channel attention weights by FF-CAM, f_s is the spatial-refined feature, ω_s is the spatial attention weights by CC-SAM, and f_{in} is the input feature.

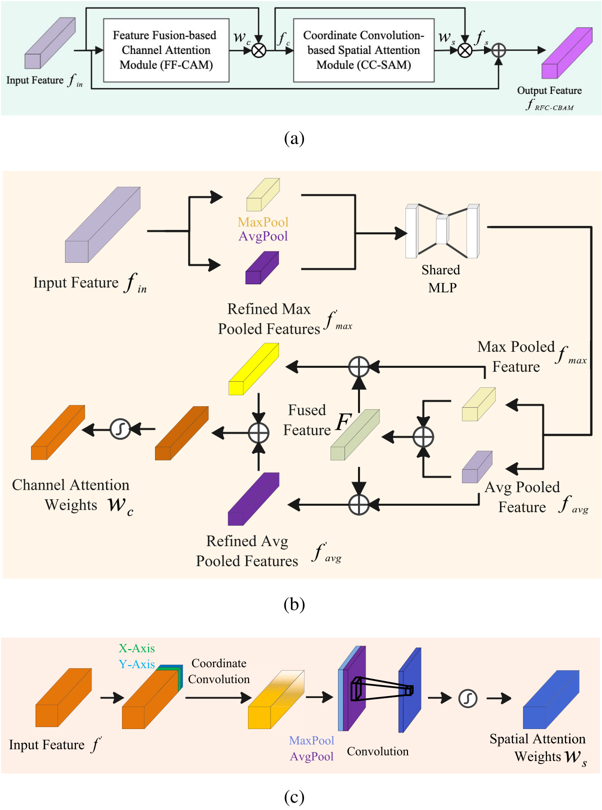


Fig. 4. Architecture of RFC-CBAM. (a) Demonstrates the flow of the proposed RFC-CBAM. The input feature will first be refined by FF-CAM. Then, a CC-SAM is utilized to provide spatial information. (b) Demonstrates the architecture of the proposed FF-CAM. (c) Demonstrates the architecture of the proposed CC-SAM.

Fig. 4(b) demonstrates the proposed FF-CAM. Similar to CBAM [17], we first aggregate spatial information of the input feature map f_{in} by using both average pooling and max pooling operations. Then, the generating feature maps will be forwarded to a shared multilayer perceptron (MLP) where the output feature maps are average-pooled feature f_{avg} and max-pooled feature f_{max} as denoted as follows:

$$\begin{aligned} f_{\max} &= \text{MLP}(\text{MP}(f_{\text{in}})) \\ f_{\text{avg}} &= \text{MLP}(\text{AP}(f_{\text{in}})) \end{aligned} \tag{7}$$

where f_{in} is the input feature, MLP represents the MLP layer, AP represents the average pooling, and MP represents the max pooling.

Furthermore, we first fuse $f_{\max}$ and f_{avg} by an element-wise summation to enhance the generalization ability of the network. Moreover, we further fuse the F with $f_{\max}$ and f_{avg}, respectively, as denoted as follows:

$$\begin{aligned} F &= f_{\max} + f_{\text{avg}} \\ f'_{\max} &= F + f_{\max} \\ f'_{\text{avg}} &= F + f_{\text{avg}}. \end{aligned} \tag{8}$$

Consequently, the channel attention weights are calculated according as follows:

$$\omega_c = \text{Sigmoid}\left(f'_{\max} + f'_{\text{avg}}\right). \tag{9}$$

Fig. 4(c) demonstrates the proposed CC-SAM. Specifically, we first utilize a coordinate convolution so that the network can better capture spatial attention. Thereafter, the features

TABLE I

DISTRIBUTION OF THE PVEL-AD DATASET BEFORE AND AFTER DATA AUGMENTATION

Category	Before Data Augmentation	After Data Augmentation			
		Training Set	Validation Set	Test Set	Total
Finger (Fr)	2958	2975	448	936	4359
Crack (Ck)	1260	2453	348	709	3510
Black Core (BC)	1028	758	101	229	1088
Thick Line (TL)	981	737	116	218	1071
Horizontal Dislocation (HD)	798	570	51	177	798
Short Circuit (SCt)	492	767	117	208	1092
Vertical Dislocation (VD)	137	2884	447	918	4249
Star Crack (SCk)	135	2905	407	878	4190
Printing Error (PE)	32	688	128	192	1008
Corner (CR)	9	205	18	57	280
Fragment (Ft)	7	148	24	45	217
Scratch (Sh)	5	97	23	35	155

after the coordinate convolution are flattened and stacked using the average pooling layer and the max pooling layer, respectively. Furthermore, a one-dimensional convolution is adopted to generate the spatial attention weights ω_s.

Consequently, RFC-CBAM maintains the channel and coordinate information from the fused features at 13 × 13, 26 × 26, and 52 × 52, which is beneficial for the network to learn the multiscale defects.

III. EXPERIMENTAL RESULTS

A. Implementation Details

The proposed CCA-YOLO is trained on a 64-bit Ubuntu OS running with i9-10980XE, 256 GB of RAM, and a single Nvidia RTX A4000 with 16 GB of RAM.

1) Datasets: We use a large-scale publicly available dataset PVEL-AD [1] for defect detection in PV cells EL images, which consists of 7842 images of 12 categories of defects for PV cell defects. To achieve diversity and balance in the dataset, we conduct several data augmentation methods such as up-down flip, left-right flip, zoomed-in view, rotation, and random noise. Specifically, we mainly increase these few-shot categories such as star vertical dislocation, star crack, printing error, corner, fragment, and scratch. Therefore, the distribution of the training validation and test sets is shown in Table I. In addition, the NEU-DET [24] and PCB [25] dataset are adopted to evaluate the generalization ability of the proposed CCA-YOLO. Specifically, the NEU-DET dataset contains 1770 images with six categories of defects in hot-rolled steel strips, and the PCB dataset contains 1386 images with six categories of defects in the printed circuit board.

2) Parameters: In the PVEL-AD dataset, we apply a stochastic gradient descent (SGD) with a weight decay of $5\mathrm{e}^{-4}$, and a momentum of 0.9 as the optimizer. In addition, the learning rate starts from $1\mathrm{e}^{-2}$ and decays to $1\mathrm{e}^{-4}$ gradually. Furthermore, the batch size is set to 8 and trained for 300 epochs on 640 × 640 images. In the NEU-DET [24] dataset, CCA-YOLO is trained on 192 × 192 images with 300 epochs while other parameters are the same. In addition, we train our CCA-YOLO on 416 × 416 images with 300 epochs while other parameters are the same.

3) Evaluation Metrics: In this article, five evaluation metrics are used to evaluate the performance of defect detection, which are listed as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{10}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{11}$$

$$F1\text{-Score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{12}$$

$$\text{AP} = \int_0^1 \text{precision}(\text{recall}) d(\text{recall}) \tag{13}$$

$$\text{mAP} = \frac{1}{N} \sum_0^N \text{AP} \tag{14}$$

where TP refers to true positives, FP refers to false positives, and FN refers to false negatives. precision (P), recall (R), and $F1$-Score (F) measure the classification performance. The average precision (AP) is evaluated with different intersections over union (IoU). Specifically, it can be reported as AP_{50} to evaluate the single-category object detection with an IoU of 0.5. Similarly, AP_{75} is used to evaluate the detection performance with an IoU of 0.75. Therefore, AP_{50}, AP_{75}, and $\text{AP}_{50:95}$ are used to evaluate the defect detection performance among all categories. In addition, multiscale detection metrics AP_S, AP_M, and AP_L are used to evaluate the AP for small-, medium-, and large-scale objects, respectively. Furthermore, mean AP (mAP) is used to evaluate the AP of all categories.

B. Ablation Study

In this section, we conduct an ablation study to evaluate the performance of each proposed module: RCC-ECA, RFC-CBAM, and MDFLM. Table II demonstrates the comparison of the detection performance between ECA [16], CBAM [17], RCC-ECA, RFC-CBAM, and MDFLM on the baseline (YOLOv4 [22]) in terms of precision (P), recall (R), $F1$-Score (F), AP_S, AP_M, AP_L, and mAP_{50} on the PVEL-AD [1] dataset, where the best results have been highlighted in bold, and the second-best results have been labeled in underline.

1) Performance of RCC-ECA: As shown in Table II, row 1 (baseline), row 2 (with ECA [16]), and row 3 (with RCC-ECA), we evaluate the performance of using RCC-ECA for backbone feature and sampling. The backbone feature plays an important role in downstream tasks such as object detection; however, the complex background of the PV cell EL images and the diversity of the defects bring challenges to efficient defect detection. The proposed RCC-ECA enhances the backbone feature representation by applying channel and spatial attention. Specifically, an efficient one-dimensional convolution is applied to make the backbone feature aware of the defect and nondefect features. In addition, we further enhance the defect feature coordinate information awareness of the network by incorporating a coordinate convolution after channel attention. To overcome the shortcomings of information loss and lack of spatial information during feature fusion, RCC-ECA is further applied to the up/down sampling process to convey coordinate information from the backbone features, which helps the network track defects during the feature fusion. Consequently, by focusing on the channel information,

TABLE II

Ablation Study on RCC-ECA, RFC-CBAM, and MDFLM in Terms of P, R, F, AP_S, AP_M, AP_L, and mAP_{50} on YOLOv4 Architecture on the PVEL-AD Dataset

ECA	RCC-ECA	CBAM	RFC-CBAM	MDFLM	Evaluation Metrices						
					P (%)	R (%)	F (%)	AP_S	AP_M	AP_L	mAP_{50}
					44.10	54.42	48.42	11.01	27.68	46.18	81.77
✓					49.79	58.61	53.61	16.36	32.93	51.88	83.80
	✓				52.70	61.15	56.37	17.38	33.18	56.66	84.81
		✓			48.30	57.58	52.25	16.79	31.77	48.68	83.34
			✓		49.48	58.69	53.37	16.81	31.09	52.58	83.77
				✓	49.67	58.75	53.83	11.08	32.44	48.56	90.79
✓		✓			50.30	59.70	54.35	19.41	32.89	51.98	85.39
	✓		✓		59.26	67.61	62.99	23.91	46.57	62.12	91.84
	✓		✓	✓	**71.71**	**76.91**	**74.19**	**26.80**	**64.78**	**74.93**	**98.57**

Notes: The 1^{st} row is the results of the baseline (YOLOv4).

RCC-ECA demonstrates its ability to identify and classify the defects, which achieves an improvement in classification as P (+2.91%), R (+2.54%), and F (+2.76%) compared to ECA and P (+8.6%), R (+6.73%), and F (+7.95%) compared to the baseline. On the other hand, providing and conveying coordinate information also brings improvement in locating the defect areas as an increment in AP_L (+4.78%) compared to ECA and AP_L (+10.48%) compared to the baseline.

2) Performance of RFC-CBAM: As illustrated in Table II, row 3 (with CBAM [17]) and row 4 (with RFC-CBAM), we further evaluate the performance of using the proposed RFC-CBAM for fused feature refinement. As discussed in the last paragraph, information loss may be caused during the feature fusion process. Therefore, we further introduce the RFC-CBAM to integrate the fused features to ensure that the network can accurately capture the categories and location of defects by FF-CAM and CC-SAM. The proposed FF-CAM helps to reallocate the channel information from different scales of features in the fused features to enhance the awareness of defects categories, while CC-SAM maintains the coordinate information from the feature fusion to ensure the effective multiscale defect localization ability. As a result, applying RFC-CBAM brings a balanced improvement in both classification and detection as P (+5.38%), R (+4.27%), F (+4.95%), AP_S (+5.8%), AP_M (+3.41%), AP_L (+6.4%), and mAP_{50} (+2%) compared to the baseline. In addition, compared to the conventional CBAM, RFC-CBAM achieves a significant improvement in AP_L (+3.9%).

3) Performance of MDFLM: During the convolution process, shallow feature maps (corresponding to large-scale feature maps) have a small receptive field and are more suitable for detecting small targets while deep feature maps (corresponding to small-scale feature maps) have a large receptive field and are suitable for detecting large targets. Since conventional YOLOv4 only uses 13 × 13, 26 × 26, and 52 × 52 feature maps for detection, it lacks the ability to localize small-scale defects. Therefore, our proposed MDFLM incorporates a high-resolution feature map that contains more detailed information to allow the network to learn more information, which significantly improves the awareness of the network to defect in medium scales as a AP_M (+4.76%). In addition, the excellent mAP_{50} (+8.96%) compared to the baseline indicated that MDFLM brings overall improvement in defect detection for all categories.

4) Performance of RCC-ECA and RFC-CBAM: The performance of RFC-CBAM for fused feature refinement is associated with the fused features, which explains the reason why RFC-CBAM only achieves balanced but not significant improvement in classification and detection. Therefore, we further discuss the situation of using RCC-ECA and RFC-CBAM simultaneously. The RCC-ECA enhances the feature representation before and during the feature fusion process, and RFC-CBAM further refines the fused feature, which allows the network to capture the defect feature and its location accurately during the complete learning process. Consequently, the combination of RCC-ECA and RFC-CBAM achieves impressive improvement in both classification and detection as reflected in Table II row 8 where P (+15.16%), R (+13.19%), F (+14.57%) AP_S (+12.9%), AP_M (+18.89%), AP_L (+15.94%), and mAP_{50} (+10.07%) compared to the baseline. In addition, we have noticed that the combination of ECA and CBAM only achieves similar performance compared to using ECA individually, suggesting that our RCC-ECA and RFC-CBAM are more generalizable.

5) Performance of RCC-ECA and RFC-CBAM and MDFLM: Due to the significant improvement of using both RCC-ECA and RFC-CBAM and the individual use of MDFLM, we further evaluate the performance of using both three modules. As listed in row 9 of Table II, incorporating MDFLM improves the performance of classification as reflected in P (+12.45%), R (+17.21%), and F (+11.2%) while the ability to localize medium- and large-scale defects has also been enhanced as a better AP_M (+18.21%), AP_L (+12.81%) compared to using both RCC-ECA and RFC-CBAM.

C. Comparison With SOTA Methods

In this section, we conduct an experiment to compare the proposed CCA-YOLO with SOTA methods such as Faster R-CNN [6], YOLOv4 [22], EfficientDet [31], YOLOv5 [26], YOLOv7 [27], YOLOv8 [28], YOLOv11 [29], RetinaNet [30], DETR [10], and SQR-DETR [32] on PVEL-AD [1] dataset.

TABLE III

Comparison of the Proposed CCA-YOLO With SOTA Methods in Terms of AP_{50}, mAP_{50}, mAP_{75}, and $mAP_{50:95}$ on the PVEL-AD Dataset

Model	AP_{50}												mAP_{50}	mAP_{75}	$mAP_{50:95}$
	Ck	Fr	BC	TL	SCk	Cr	Ft	Sh	HD	VD	PE	SCt			
YOLOv4 [22], 2020	72.86	91.80	96.95	73.28	95.88	49.94	70.14	39.15	96.46	98.93	96.68	99.21	81.77	38.05	44.10
YOLOv5-m [26], 2021	82.73	90.22	29.36	81.77	98.06	34.66	20.07	51.54	63.97	96.00	98.87	0.00	62.27	30.21	31.40
YOLOv5-l [26], 2021	81.20	91.20	87.19	81.84	98.57	59.96	61.83	67.81	84.66	97.49	99.94	0.06	75.98	32.38	35.94
YOLOv5-x [26], 2021	81.45	90.87	93.52	80.39	98.01	51.88	67.40	73.54	78.17	90.50	**100.00**	0.01	75.48	30.81	35.53
YOLOv7-l [27], 2022	86.31	93.78	98.21	85.27	99.47	90.02	99.31	84.75	0.00	98.36	**100.00**	98.84	86.19	64.54	58.79
YOLOv7-x [27], 2022	90.21	93.16	98.74	87.48	99.41	87.12	91.58	87.63	0.00	98.91	**100.00**	98.96	86.10	65.05	59.18
YOLOv8-l [28], 2023	94.72	93.81	**99.79**	84.00	98.98	96.79	95.77	**100.00**	64.78	99.92	**100.00**	99.01	93.96	73.73	67.89
YOLOv8-x [28], 2023	92.61	93.79	98.78	86.94	99.83	91.76	97.22	**100.00**	60.71	99.87	99.99	98.76	93.35	74.12	67.26
YOLOv11-x [29], 2024	95.09	95.06	97.82	86.72	99.01	97.03	**100.00**	**100.00**	95.98	94.06	**100.00**	99.01	96.64	75.81	**71.75**
RetinaNet [30], 2017	87.90	54.39	97.91	84.24	99.19	92.85	99.94	83.56	0.00	66.10	98.41	99.01	80.29	61.68	55.40
Faster R-CNN [6] 2015	85.18	36.44	99.47	58.83	95.92	92.98	96.29	84.95	77.18	99.81	81.29	99.98	84.03	49.60	51.03
EfficientDet-D3 [31], 2016	91.75	59.64	99.59	84.86	98.97	95.40	99.94	93.11	0.00	69.93	99.97	**100.00**	82.76	62.83	56.76
DETR [10], 2020	**98.72**	96.65	94.09	90.97	**100.00**	98.73	98.99	97.99	99.23	98.88	85.75	**100.00**	96.67	72.91	67.60
SQR-DETR [32], 2023	97.77	**97.03**	94.12	93.27	**100.00**	**100.00**	99.99	**100.00**	99.24	98.99	87.27	**100.00**	97.31	75.48	67.67
Focus-DETR [12], 2023	97.90	90.16	95.10	**94.48**	99.61	99.97	98.73	88.86	99.99	99.93	85.40	99.98	95.84	79.91	70.37
Proposed CCA-YOLO	97.59	96.79	99.12	92.32	99.98	97.03	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**98.57**	**84.36**	71.71

1) Detection on 12 Categories: Table III demonstrates the comparison between the proposed CCA-YOLO with SOTA methods in terms of AP_{50}, mAP_{50}, mAP_{75}, and $mAP_{50:95}$ on PVEL-AD, where the best results have been highlighted in bold and the second-best results have been labeled using underline. As demonstrated in Table III, CCA-YOLO achieves the best AP_{50} on six categories (Ft, Sh, HD, VD, PE, and SCt) and the second-best AP_{50} on two categories (Fr and SCk). Although we only achieve better AP_{50} on eight categories, CCA-YOLO achieves a more balanced performance in multicategory defect detection as reflected in all 12 categories of AP_{50} beyond 90% while the second-best method (SQR-DETR) only achieves 87.27% in PE. In addition, as for those failure cases of CCA-YOLO (Ck and Bc), the difference between our CCA-YOLO and the best is small as DETR [10] in Ck (−1.13%), and YOLOv8-l [28] in BC (−0.67%). Consequently, our CCA-YOLO not only achieves the best overall performance among the 12 categories but the performance under each category is impressive. In addition to mAP_{50}, we further evaluate our method on a high IoU threshold and the comprehensive detection performance using mAP_{75}, and $mAP_{50:95}$. The results have indicated that our method significantly outperforms YOLOv11-x on mAP_{75} (84.36%). Although our method is 0.04% lower in $mAP_{50:95}$, we believe this slight difference is primarily due to YOLOv11-x is a general-purpose object detection method that usually focuses on medium and large scales of the object. The medium and large sizes of defects usually occupy a larger area in the PV image, making it easier to achieve better performance in high IoU confidence, resulting in significant overall detection performance $mAP_{50:95}$. In contrast, our method focuses on multiscale defect detection, especially small-scale defects, thus, our CCA-YOLO is able to achieve the best mAP_{50} and mAP_{75}.

TABLE IV

Comparison of the Proposed CCA-YOLO With SOTA Methods in Terms of P, R, F, AP_S, AP_M, and AP_L on the PVEL-AD Dataset

Methods	Evaluation Metrices					
	P (%)	R (%)	F (%)	AP_S	AP_M	AP_L
YOLOv4 [22], 2020	44.10	54.42	48.42	11.01	27.68	46.18
YOLOv5-l [26], 2021	35.95	46.68	40.24	15.80	45.63	33.26
YOLOv5-m [26], 2021	31.40	43.34	35.54	16.31	47.94	29.57
YOLOv5-x [26], 2021	35.53	46.02	39.78	19.48	45.02	33.70
YOLOv7-l [27], 2022	58.79	65.25	61.76	16.70	50.23	60.19
YOLOv7-x [27], 2022	59.18	66.47	62.49	17.01	53.40	60.03
YOLOv8-l [28], 2023	67.90	75.09	70.76	17.02	60.11	71.74
YOLOv8-x [28], 2023	67.26	74.52	70.16	19.17	63.59	70.94
YOLOv11-x [29], 2024	**71.75**	75.53	73.55	19.91	**67.05**	**76.12**
EfficientDet-D3 [31], 2019	56.77	63.43	59.81	5.65	47.35	59.46
Faster R-CNN [6], 2015	51.03	61.21	55.19	1.05	29.24	61.77
Retinanet [30], 2017	55.40	61.85	58.34	23.58	50.64	59.70
DETR [10], 2020	67.40	76.11	71.49	13.07	52.88	73.66
SQR-DETR [32], 2023	67.67	75.65	71.44	21.41	52.78	70.25
Focus-DETR [12], 2023	70.37	76.89	74.12	20.44	61.89	72.90
Proposed CCA-YOLO	71.71	**76.91**	**74.19**	**26.80**	64.78	74.93

2) Overall Classification and Multiscale Detection Performance: We further conduct an experiment to evaluate the classification and multiscale defect detection performance. Table IV demonstrates the comparison between the proposed CCA-YOLO with SOTA methods in terms of precision (P), recall (R), $F1$-Score (F), AP_S, AP_M, and AP_L where the best results have been highlighted in bold and the second-best have been labeled using underline. In terms of classification performance, our CCA-YOLO is slightly inferior to YOLOv11-x in precision (−0.04%) but our method outperforms the existing methods by achieving the highest recall (76.91%) and $F1$-Score (74.19%). For multiscale detection performance, CCA-YOLO achieves the best AP_S (26.80%) and the second-best AP_M (64.78%) and AP_L (74.93%). Although our method is slightly inferior to the current SOTA YOLOv11-x in terms of AP_M (−2.27%) and AP_L (−1.19%), we believe that these gaps are mainly due to the different emphasis of different models on multiscale target detection capabilities. YOLOv11 is a general-purpose object detection method designed to identify common objects such as pedestrians, vehicles, and animals. Therefore, YOLOv11-x achieves a better AP_M and AP_L, which also explains why it can achieve a better $mAP_{50:95}$. In contrast, our primary focus is on enhancing the ability to locate multiscale defects, particularly small-scale defects like *Finger*, *Thick Line*, *Star Crack*, and *Printing Error*, which constitute more than half of the dataset. Usually, it is hard to locate the small-scale defects accurately; therefore, our CCA-YOLO achieves the best mAP_{50}, AP_{75}, and AP_S but slightly lower than YOLOv11-x in $mAP_{50:95}$. Since small-scale

TABLE V

COMPARISON OF THE PROPOSED CCA-YOLO WITH SOTA METHODS IN TERMS OF AP_{50} AND mAP_{50} ON THE NEU-DET DATASET

Methods (year)	AP_{50}						mAP_{50}
	Cr	In	Pa	PS	RS	Sc	
†Faster R-CNN [6], 2015	44.2	81.9	92.0	78.4	63.1	95.2	75.8
†RetinaNet [30], 2017	42.7	76.5	92.4	85.7	60.5	84.4	74.6
†SparseRCNN [33], 2021	43.9	72.6	75.6	73.6	53.1	81.3	66.7
†YOLOv5 [26], 2021	48.5	85.1	91.6	76.9	66.4	93.1	77.0
†DH-RCNN [34], 2021	45.3	80.5	92.1	85.5	59.8	91.7	75.6
†YoloX [35], 2022	55.4	85.6	94.2	87.4	57.5	96.5	79.3
†YOLOv7 [27], 2022	49.6	83.6	81.9	83.4	59.8	95.7	75.7
†ESNet [36], 2022	56.0	87.6	88.3	87.4	60.4	94.9	79.1
†DetectorRS [37], 2022	**58.9**	88.3	88.5	87.9	62.7	95.7	80.2
†CenterNet2 [38], 2022	45.7	85.1	90.0	82.5	70.7	95.8	79.4
†EDDNet [9], 2023	41.7	76.3	86.3	85.1	58.1	85.6	72.4
†RDD-YOLO [39], 2023	52.9	85.9	94.4	86.2	70.7	96.6	81.1
†MSC-DNet [40], 2023	42.4	84.5	94.3	91.5	71.6	92.0	79.4
†DeformableDETR [41], 2021	26.4	66.0	73.7	67.1	39.1	78.1	58.4
†DeformableDETR$_{\mathrm{IBR}}$ [41], 2021	31.1	79.6	84.8	70.5	61.2	87.3	63.3
†DeformableDETR$_{\mathrm{IBR2}}$ [41], 2021	36.9	84.4	89.4	68.2	65.0	89.7	72.3
†DeformableDETR$^{6}_{\mathrm{IBR2}}$ [41], 2021	32.6	84.8	89.8	71.9	64.5	89.3	72.2
†DeformableDETR$^{2}_{\mathrm{IBR2}}$ [41], 2021	29.4	84.5	90.5	68.3	60.8	83.0	69.4
†DeformableDETR$^{\mathrm{no\_trans}}_{\mathrm{IBR2}}$ [41], 2021	36.9	84.4	89.4	68.2	65.0	89.7	71.3
†SAM-DETR [42], 2022	21.5	74.8	85.0	64.5	50.0	81.0	62.8
†SAM-DETR-DC5 [42], 2022	32.9	82.2	87.0	69.7	60.0	89.4	70.2
†SAM-DETR-SMCA [42], 2022	41.6	83.7	84.9	69.1	60.4	88.0	71.3
†DAB-deform-DETR [43], 2022	39.3	86.8	88.8	73.8	61.5	90.8	73.5
†DN-DETR [44], 2022	21.5	66.0	77.6	60.7	44.1	68.9	56.5
†DN-deform-DETR [44], 2022	44.1	84.5	86.9	71.0	59.3	87.4	72.2
†PMSA-DyTr [11], 2024	54.9	88.6	92.9	84.5	70.5	95.7	81.2
‡DCAM-Net [45], 2023	48.0	86.1	94.2	90.9	78.2	**97.8**	82.6
‡AFF-Net [13], 2024	50.5	88.1	**96.6**	92.9	76.1	96.9	83.5
†† YOLOv4 [22], 2020	13.1	58.6	80.5	93.7	50.8	58.7	59.2
†† YOLOv8-x [28], 2023	43.3	83.2	89.7	95.6	84.1	79.9	79.3
†† YOLOv11-x [29], 2024	49.3	**88.8**	**96.6**	85.9	**100.0**	84.3	84.2
Proposed CCA-YOLO	45.3	85.6	96.0	**100.0**	**100.0**	82.1	**84.8**

† The results are from [11].
‡ The results are from [13].
†† The results are implemented by us.

TABLE VI

COMPARISON OF THE PROPOSED CCA-YOLO WITH SOTA METHODS IN TERMS OF AP_{50} AND mAP_{50} ON THE PCB DATASET

Methods (year)	AP_{50}						mAP_{50}
	MH	MB	OC	SH	SP	Sc	
†SparseRCNN [33], 2021	94.9	95.1	85.4	86.8	91.1	83.7	89.5
†YOLOv5 [26], 2021	98.6	85.0	87.5	95.7	73.8	85.8	87.8
†YOLOX [35], 2021	99.2	83.4	84.6	95.1	72.6	85.4	86.8
†YOLOv7 [27], 2022	89.6	87.9	87.5	86.3	90.4	87.7	88.2
†ESNet [36], 2022	98.1	96.9	98.3	93.4	89.5	95.0	95.2
†DetectorRS [37], 2022	98.4	97.2	87.9	93.8	91.9	94.1	93.9
†CenterNet2 [38], 2022	99.8	95.8	91.9	94.6	94.1	90.5	94.5
†EDDNet [9], 2022	98.2	96.2	**98.8**	97.7	92.7	96.8	**96.7**
†Deformable-DETR [41], 2021	94.3	85.1	76.1	84.8	86.7	65.7	82.1
†SAM-DETR [42], 2022	94.3	96.0	81.6	83.5	90.2	72.3	86.3
†DAB-deform-DETR [43], 2022	99.2	97.6	88.2	89.0	94.0	92.6	93.4
†DN-deform-DETR [44], 2022	96.4	97.2	86.5	88.9	**97.0**	90.9	92.8
†PMSA-DyTr [11], 2024	97.8	97.3	96.8	92.7	92.5	94.9	95.3
‡LDD-Net [15], 2024	99.1	98.7	98.5	97.9	88.7	92.3	95.9
†† YOLOv11-x [29], 2024	99.8	**100.0**	85.6	**98.9**	94.6	92.5	95.3
Proposed CCA-YOLO	**100.0**	96.5	96.9	94.3	92.7	**99.7**	**96.7**

† The results are from [11].
‡ The results are from [15]
†† The results are implemented by us.

defects are more common in PV cells, we believe that our CCA-YOLO has made its own contribution to defect detection for PV cells from different perspectives. Additionally, compared to other SOTA methods such as DETR, SQR-DETR, Focus-DETR, and YOLOv8, our method demonstrates superior performance in multiscale defect detection. In summary, our method performs well in relatively high-confidence and small-scale defect detection. Although it is slightly lower than YOLOv11-x in $mAP_{50:95}$, this gap is minimal. Future work will further optimize our model to achieve overall performance improvements.

3) Generalization Ability: We further conduct an experiment to demonstrate the generalization ability of the proposed CCA-YOLO on the NEU-DET [24] dataset as demonstrated in Table V. Specifically, our proposed CCA-YOLO achieves the best AP_{50} on two categories (PS and RS), one second-best AP_{50} on Pa, and the best mAP_{50} (84.8%). Moreover, we conduct another experiment to evaluate the generalization ability in PCB dataset [25] as demonstrated in Table VI. The experimental results have indicated that our CCA-YOLO achieves the best mAP_{50} (96.7%) and also achieves the best AP_{50} on two categories (MH of 100.0% and Sc of 99.7%). Therefore, the generalization ability of the proposed CCA-YOLO has been demonstrated.

4) Visual Comparison: Fig. 5 demonstrates the defect detection results using YOLOv4 [22], YOLOv5-x [26], YOLOv7-x [27], YOLOv8-x [28], YOLOv11-x [29], Faster R-CNN [6], DETR [10], and the proposed CCA-YOLO on the PVEL-AD dataset. Specifically, the first row is the GT, the second row is the result of YOLOv4, the third row is the result of YOLOv5-x, the fourth row is the result of YOLOv7-x, the fifth row is the result of YOLOv8-x, the sixth row is the result of Faster R-CNN, the seventh row is the result of DETR, the eighth row is the results of YOLOv11-x, and the ninth row is the results of the proposed CCA-YOLO. (a1)–(f1) are the detection results in raw images. (a2)–(f2) are the detection results in binary images. Furthermore, the red cycle demonstrates missed detection, the green cycle demonstrates incomplete detection, and the yellow cycle demonstrates misdiagnosis in the raw images. In the binary images, the black area represents the background, and the white area represents the detected defect areas. As demonstrated in Fig. 5, our CCA-YOLO demonstrates a better ability to locate and classify multiscale defects than existing methods. Regardless of the large-scale defects, such as horizontal and vertical dislocations, that are difficult to locate precisely, or the small-scale defects, like finger anomalies, as well as irregularly-scaled defects such as thick lines and cracks, our CCA-YOLO is able to demarcate boundary with notable accuracy, while correctly categorizing each defect.

5) Computation Efficiency: As illustrated in Table VII, we evaluate the computation efficiency between CCA-YOLO with SOTA methods in terms of parameters, inference time, and FLOPs on the PVEL-AD dataset. Compared to these time-efficient methods such as YOLOv5-l (11.7 ms infer time), YOLOv5-x (14.74 ms infer time), and RetinaNet (13.95 ms infer time), our CCA-YOLO sacrifice a little computational complexity but achieves significant detection performance improvement as reflected in the increased mAP_{50} compared to YOLOv5-l (+22.59% mAP_{50}, −4.59 ms infer time), YOLOv5-x (+23.09% mAP_{50}, −1.55 ms infer time), and RetinaNet (+18.28% mAP_{50}, −2.34 ms infer time). In addition, compared to the baseline YOLOv4, our method only costs 1.25 ms infer time more but achieves 16.8%

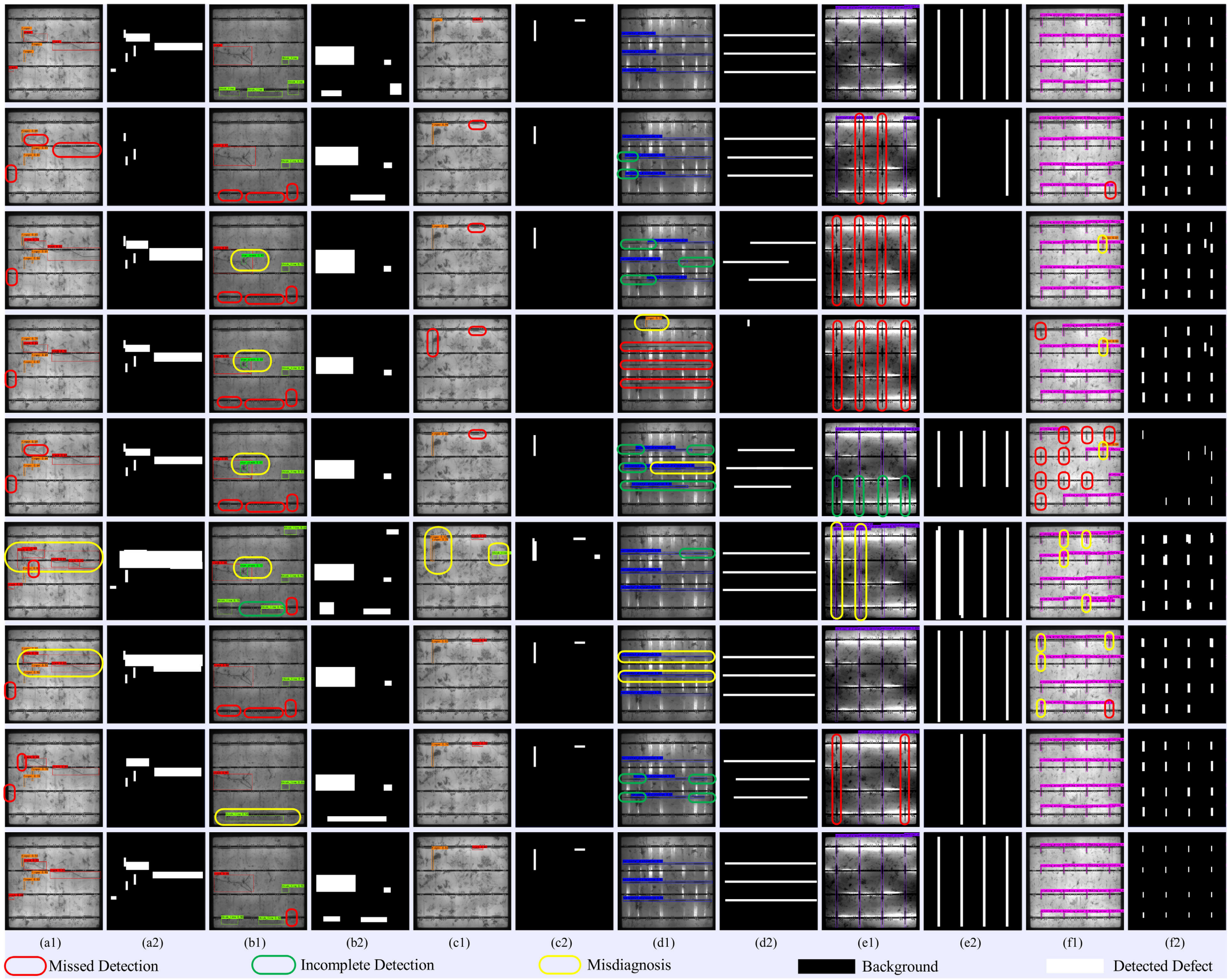


Fig. 5. Visualization of the defect detection results using YOLOv4, YOLOv5-x, YOLOv7-x, YOLOv8-x, Faster R-CNN, DETR, YOLOv11-x, and the proposed CCA-YOLO: the first row is the GT; second row is the result of YOLOv4; third row is the result of YOLOv5-x; fourth row is the result of YOLOv7-x; fifth row is the result of YOLOv8-x; sixth row is the result of Faster R-CNN; seventh row is the result of DETR; eighth row is the results of the YOLOv11-x; and ninth row is the results of the proposed CCA-YOLO. (a1)–(f1) detection results in raw images. (a2)–(f2) detection results in binary images. Furthermore, the red cycle demonstrates missed detection, the green cycle demonstrates incomplete detection, and the yellow cycle demonstrates misdiagnosis in the raw images. In the binary images, the black area represents the background, and the white area represents the detected defect areas.

TABLE VII

COMPARISON OF THE PROPOSED CCA-YOLO WITH SOTA METHODS IN TERMS OF PARAMETERS, INFER TIME, AND FLOPS ON THE PVEL-AD DATASET

Methods	Para. (M)	Infer (ms)	FLOPs (G)
YOLOv4 [22], 2020	63.94	15.04	70.95
YOLOv5-l [26], 2021	46.14	**11.7**	54.11
YOLOv5-x [26], 2021	86.22	14.74	102.31
YOLOv7-l [27], 2022	37.19	18.18	**52.55**
YOLOv7-x [27], 2022	70.81	19.23	94.44
YOLOv8-l [28], 2023	43.63	16.52	82.71
YOLOv8-x [28], 2023	68.16	23.06	129.07
YOLOv11-x [29], 2024	56.9	13.40	194.9
RetinaNet [30], 2017	**36.56**	13.95	83.69
Faster R-CNN [6], 2015	136.91	27.43	200.98
DETR [10], 2020	41.5	30.42	59.8
SQR-DETR [32] 2023	43.65	35.79	64.73
Focus-DETR [12], 2023	49.2	64.44	113.9
Proposed CCA-YOLO	82.61	16.29	96.86

improvement in mAP$_{50}$. Moreover, compared to the methods with high detection performance such as SQR-DETR, DETR, YOLOv8-l, and YOLOv8-x, our CCA-YOLO achieves the best detection results with the shortest infer time. Further, compared to the recent YOLOv11-x, we take 2.83 ms more but achieve improvement in mAP$_{50}$ (+1.94%), mAP$_{75}$ (+8.55%), and AP$_s$ (+6.89%). Although we sacrifice some inference time, we provide a novel and effective PV cell defect detection solution. Therefore, our CCA-YOLO demonstrates its advantages in balancing time efficiency and multiscale detection performance.

IV. CONCLUSION

In conclusion, we propose a novel PV cell defect detection method called CCA-YOLO, which incorporates four modules: RCC-ECA, MDFLM, RCC-Up/Down, and RFC-CBAM. Specifically, the RCC-ECA enhances the backbone feature representation by focusing on both channel and coordinate aspects, which is beneficial for efficiently learning from the backbone features. Furthermore, we introduce an RCC-Up/Down that incorporates RCC-ECA into the sampling process to convey coordinate information during the sampling process. To learn the interclass/intraclass variations and intraclass similarity, we further propose an MDFLM to incorporate a large-scale feature map into the feature fusion process, which overcomes the shortcomings of multiscale defect detection capabilities. Moreover, the RFC-CBAM is adopted to refine the fused feature at three scales by FF-CAM and CC-SAM to maintain the channel and coordinate awareness. An ablation study has demonstrated the superiority of the proposed RCC-ECA, RFC-CBAM, and MDFLM. Finally, we compared the proposed CCA-YOLO with several SOTA methods and the experimental results indicate that our CCA-YOLO achieves better performance on 12-category multiscale defect detection in the PVEL-AD dataset. In addition, we further demonstrate the generalization ability of our CCA-YOLO in the NEU-DET and PCB datasets.

Although our CCA-YOLO achieves impressive performance on defect detection, it still has some limitations, such as more computational resources costs, insufficient bounding box localization, and limited capability for small-scale defects. Multiscale especially small-scale defect detection still faces lots of challenges in the industry, therefore, we will focus on efficient multiscale defect detection in the future. In addition, since time efficiency is important for defect detection, we will also try to find an efficient solution for defect detection in the industry precisely and rapidly.

REFERENCES

[1] B. Su, Z. Zhou, and H. Chen, "PVEL-AD: A large-scale open-world dataset for photovoltaic cell anomaly detection," *IEEE Trans. Ind. Informat.*, vol. 19, no. 1, pp. 404–413, Jan. 2023.

[2] B. Su, H. Chen, Y. Zhu, W. Liu, and K. Liu, "Classification of manufacturing defects in multicrystalline solar cells with novel feature descriptor," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 12, pp. 4675–4688, Dec. 2019.

[3] J. Xu, Y. Liu, and Y. Wu, "Automatic defect inspection for monocrystalline solar cell interior by electroluminescence image self-comparison method," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–11, 2021.

[4] F.-C. Chen and M. R. Jahanshahi, "NB-CNN: Deep learning-based crack detection using convolutional neural network and Naïve Bayes data fusion," *IEEE Trans. Ind. Electron.*, vol. 65, no. 5, pp. 4392–4400, May 2018.

[5] B. Su, H. Chen, and Z. Zhou, "BAF-detector: An efficient CNN-based detector for photovoltaic cell defect detection," *IEEE Trans. Ind. Electron.*, vol. 69, no. 3, pp. 3161–3171, Mar. 2022.

[6] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[7] F.-C. Chen and M. R. Jahanshahi, "NB-FCN: Real-time accurate crack detection in inspection videos using deep fully convolutional network and parametric data fusion," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 8, pp. 5325–5334, Aug. 2020. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8938126

[8] J. Bao and X. Yuan, "YOLO-iCBAM: An improved YOLOv4 based on CBAM for defect detection," *Proc. SPIE*, vol. 13091, pp. 520–525, Mar. 2024.

[9] R. Qin, N. Chen, and Y. Huang, "EDDNet: An efficient and accurate defect detection network for the industrial edge environment," in *Proc. IEEE 22nd Int. Conf. Softw. Quality, Rel. Secur. (QRS)*, Dec. 2022, pp. 854–863.

[10] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Computer Vision—ECCV 2020*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds., Cham, Switzerland: Springer, Aug. 2020, pp. 213–229.

[11] J. Su, Q. Luo, C. Yang, W. Gui, O. Silvén, and L. Liu, "PMSA-DyTr: Prior-modulated and semantic-aligned dynamic transformer for strip steel defect detection," *IEEE Trans. Ind. Informat.*, vol. 20, no. 4, pp. 6684–6695, Apr. 2024.

[12] D. Zheng, W. Dong, H. Hu, X. Chen, and Y. Wang, "Less is more: Focus attention for efficient DETR," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 6674–6683.

[13] Y. Du, H. Chen, Y. Fu, J. Zhu, and H. Zeng, "AFF-Net: A strip steel surface defect detection network via adaptive focusing features," *IEEE Trans. Instrum. Meas.*, vol. 73, pp. 1–14, 2024.

[14] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, Oct. 2021, pp. 9992–10002.

[15] L. Zhang, J. Chen, J. Chen, Z. Wen, and X. Zhou, "LDD-Net: Lightweight printed circuit board defect detection network fusing multi-scale features," *Eng. Appl. Artif. Intell.*, vol. 129, Mar. 2024, Art. no. 107628.

[16] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 11531–11539.

[17] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Computer Vision—ECCV 2018*, vol. 11211, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., Cham, Switzerland: Springer, Oct. 2018, pp. 3–19.

[18] B. Su, H. Chen, P. Chen, G. Bian, K. Liu, and W. Liu, "Deep learning-based solar-cell manufacturing defect detection with complementary attention network," *IEEE Trans. Ind. Informat.*, vol. 17, no. 6, pp. 4084–4095, Jun. 2021.

[19] H. Tan, H. Zhou, C. Duan, T. Zhang, and L. Zhang, "RAFBSD: An efficient detector for accurate identification of defects in photovoltaic cells," *IEEE Access*, vol. 12, pp. 61512–61528, 2024.

[20] D. Lang and Z. Lv, "A PV cell defect detector combined with transformer and attention mechanism," *Sci. Rep.*, vol. 14, no. 1, p. 20671, Sep. 2024.

[21] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A new backbone that can enhance learning capability of CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 390–391.

[22] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[23] R. Liu et al., "An intriguing failing of convolutional neural networks and the CoordConv solution," in *Advances in Neural Information Processing Systems*, vol. 31. Red Hook, NY, USA: Curran Associates, 2018.

[24] K. Song and Y. Yan, "A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects," *Appl. Surf. Sci.*, vol. 285, pp. 858–864, Nov. 2013.

[25] L. Dai. (2024). *Public Synthetic PCB Dataset*. Accessed: Mar. 28, 2024. [Online]. Available: https://robotics.pkusz.edu.cn/resources/dataset/

[26] G. Jocher, 2020, "Ultralytics/YOLOv5: V3. 1-bug fixes and performance improvements," Zenodo, doi: 10.5281/zenodo.4154370.

[27] C.-Y. Wang, A. Bochkovskiy, and H.-Y.-M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 7464–7475.

[28] G. Jocher, A. Chaurasia, and J. Qiu. (2023). *Ultralytics YOLOv8*. [Online]. Available: https://github.com/ultralytics/ultralytics

[29] G. Jocher and J. Qiu. (2024). *Ultralytics YOLO11*. [Online]. Available: https://github.com/ultralytics/ultralytics

[30] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[31] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10781–10790.

[32] F. Chen, H. Zhang, K. Hu, Y.-K. Huang, C. Zhu, and M. Savvides, "Enhanced training of query-based object detection via selective query recollection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 23756–23765.

[33] P. Sun et al., "Sparse R-CNN: End-to-end object detection with learnable proposals," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14454–14463.

[34] Y. Wu et al., "Rethinking classification and localization for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10186–10195.

[35] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO series in 2021," 2021, *arXiv:2107.08430*.

[36] X. Yu, W. Lyu, D. Zhou, C. Wang, and W. Xu, "ES-Net: Efficient scale-aware network for tiny defect detection," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–14, 2022.

[37] S. Qiao, L.-C. Chen, and A. Yuille, "DetectoRS: Detecting objects with recursive feature pyramid and switchable Atrous convolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10213–10224.

[38] L. Cai, Z. Zhang, Y. Zhu, L. Zhang, M. Li, and X. Xue, "BigDetection: A large-scale benchmark for improved object detector pre-training," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 4776–4786.

[39] C. Zhao, X. Shu, X. Yan, X. Zuo, and F. Zhu, "RDD-YOLO: A modified YOLO for detection of steel surface defects," *Measurement*, vol. 214, Jun. 2023, Art. no. 112776.

[40] R. Liu, M. Huang, Z. Gao, Z. Cao, and P. Cao, "MSC-DNet: An efficient detector with multi-scale context for defect detection on strip steel surface," *Measurement*, vol. 209, Mar. 2023, Art. no. 112467.

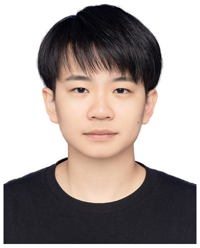
[41] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," 2020, *arXiv:2010.04159*.

[42] G. Zhang, Z. Luo, J. Huang, S. Lu, and E. P. Xing, "Semantic-aligned matching for enhanced DETR convergence and multi-scale feature fusion," *Int. J. Comput. Vis.*, vol. 132, no. 8, pp. 2825–2844, Feb. 2024.

[43] S. Liu et al., "DAB-DETR: Dynamic anchor boxes are better queries for DETR," 2022, *arXiv:2201.12329*.

[44] F. Li, H. Zhang, S. Liu, J. Guo, L. M. Ni, and L. Zhang, "DN-DETR: Accelerate DETR training by introducing query denoising," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 13619–13627.

[45] H. Chen, Y. Du, Y. Fu, J. Zhu, and H. Zeng, "DCAM-Net: A rapid detection network for strip steel surface defects based on deformable convolution and attention mechanism," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–12, 2023.



Junqi Bao (Graduate Student Member, IEEE) received the B.Sc. degree in computer science from Zijin College, Nanjing University of Science and Technology, Nanjing, China, in 2021, and the M.Sc. degree in big data and Internet of Things from Macao Polytechnic University, Macau, China, in 2023, where he is currently pursuing the Ph.D. degree with the School of Applied Sciences.

His research interests include image matching, multiple-view geometry, robotics, and SLAM.

Xiaochen Yuan (Senior Member, IEEE) received the Ph.D. degree in software engineering from the University of Macau, Macau, China, in 2013.

From 2014 to 2015, she was a Post-Doctoral Fellow at the Department of Computer and Information Science, University of Macau. From 2016 to 2021, she was an Assistant Professor and an Associate Professor at the Faculty of Information Technology, Macau University of Science and Technology, Macau. She is currently an Associate Professor with the Faculty of Applied Sciences, Macao Polytechnic University, Macau. Her research interests include multimedia forensics and security, digital watermarking, AI model security, quantum watermarking, remote image processing, and deep learning techniques and applications.

Qingying Wu (Graduate Student Member, IEEE) received the B.S. degree in electrical engineering and automation from Changzhou Institute of Technology, Changzhou, China, in 2020, and the M.S. degree in big data and Internet of Things from Macao Polytechnic University, Macau, China, in 2022, where she is currently pursuing the Ph.D. degree in computer applied technology.

Her research interests include cognitive radio, wireless communications, and artificial intelligence.

Chan-Tong Lam (Senior Member, IEEE) received the B.Sc. (Eng.) and M.Sc. (Eng.) degrees from Queen's University, Kingston, ON, Canada, in 1998 and 2000, respectively, and the Ph.D. degree from Carleton University, Ottawa, ON, Canada, in 2007.

He is currently an Associate Professor with the Faculty of Applied Sciences, Macao Polytechnic University, Macau, SAR, China. From 2004 to 2007, he participated in the European Wireless World Initiative New Radio (WINNER) Project. His research interests include mobile wireless communications, digital signal processing, quantum information, machine learning in communications, and computer vision in smart cities.

Dr. Lam is a member of the IEEE Communications and Signal Processing Society.

Wei Ke (Member, IEEE) received the Ph.D. degree from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2012.

He is a Full Professor in computer applied technology with Macao Polytechnic University, Macau, China. His research interests include programming languages, image processing, computer vision and component-based engineering and systems. His recent research interests include the design and implementation of development tools and open platforms for applications of computer vision and pattern recognition.

Ping Li (Member, IEEE) received the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong, Hong Kong, in 2013.

He is currently an Assistant Professor with the Department of Computing and an Assistant Professor with the School of Design, The Hong Kong Polytechnic University, Hong Kong. He has published over 200 top-tier scholarly research articles (e.g., TPAMI, TVCG, TIP, TNNLS, TMI, TMM, TCSVT, TCYB, TBME, TSMC, TII, AAAI, CVPR, ICCV, NeurIPS, Nature Metabolism), pioneered several new research directions, and made a series of landmark contributions in his areas. He has an excellent research project reported by the *ACM TechNews*, which only reports the top breakthrough news in computer science worldwide. More importantly, however, many of his research outcomes have strong impacts to research fields, addressing societal needs and contributed tremendously to the people concerned. His current research interests include image/video stylization, colorization, artistic rendering and synthesis, computational art, and creative media.