Engineering Applications of Artificial Intelligence 136 (2024) 109058

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

A robust defect detection method with a generalization enhancer and cross-modality aggregator for cylinder bores

Xujie He, Jing Jin *, Duo Chen, Cangtian Zhou

School of Astronautics, Harbin Institute of Technology, Harbin, 150001, China

ARTICLE INFO

Keywords:
Open-set defect detector
Visual foundation model
Generalization enhancer
Cross-modality aggregator
Supervised learning

ABSTRACT

High-quality cylinder bores in automobile engines enable drivers to respond quickly to emergencies. Automated detection methods are gradually being adopted across various industries. However, uncontrollable factors and improper preservation methods lead to various types of defects on cylinder bores, thereby causing existing high-performance detectors to exhibit not only undesirable generalizability for unseen defect types but also a certain degree of missed detection for defects of seen types, thereby allowing defective cylinder bores to flow into the market. To address these issues, we propose a foundation-model-based **r**obust defect detection method with **h**igh **g**eneralizability for cylinder bores (RHG-Detector). Specifically, to address unseen defect categories, we propose a generalization enhancer comprising a box filter, a region extractor and a **de**fect **di**scriminator (DeDi) based on a foundation model to extend defect detection from a closed set to an open set. To reduce missed detections, we adopt a cross-modality aggregator to aggregate the detection results from different modalities. Additionally, we collected and annotated challenging defect classification and detection datasets for cylinder bores, named HIT-EngDC (**H**arbin **I**nstitute of **T**echnology **En**gine **d**efect **c**lassification dataset) and HIT-EngDD2 (**Eng**ine **d**efect **d**etection dataset-version **2**), which cover nearly all types of cylinder bore defects. Extensive experiments on HIT-EngDC and HIT-EngDD2 demonstrate the state-of-the-art performance of RHG-Detector, with a classification accuracy of 92.0 and a mAP@50 (mean average precision under intersection over union = 0.5) score of 45.2, where the latter is increased by ~6 and ~3 compared to the corresponding FasterRCNN (faster region-based convolutional neural networks) and YOLOv7 (you only look once) scores, respectively.

1. Introduction

As core components of automobile engines, cylinder bores provide power and a driving force for automobiles and thus determine their acceleration ability, fuel efficiency and reliability. However, due to poor material quality, manufacturing errors, equipment failure, human error and other factors, defective cylinder bores are often produced during the manufacturing process. Such unqualified engine cylinder bores can weaken the driving force of automobiles, thereby affecting drivers' ability to respond to emergencies. Additionally, worn cylinder bores can leak coolant and oil, thereby potentially causing severe accidents. Therefore, to ensure the quality and reliability of engine cylinder bores, strict quality inspection is indispensable.

Currently, quality inspection of engine cylinder bores is completed mainly via manual visual inspection, which has low efficiency and is easily affected by subjective factors and low traceability. Nevertheless, automated quality inspection equipment, with its advantages of high efficiency, consistency, accuracy and convenient traceability because of reasonably designed hardware devices and detection methods, is gradually being adopted in a wide variety of applications (Aggarwal et al., 2023). As a core functionality of automated detection equipment, detection methods are crucial for determining detection results. Many efforts have been made to address defect detection. Current defect detection methods can be divided into two types: methods based on handcrafted features and methods based on deep features. Handcrafted features, which are obtained by applying traditional oriented image processing methods, can be further divided into color-based features (e.g., color histograms (Wang et al., 2022)), texture-based features (e.g., SIFT features (Zhang et al., 2023)), and shape-based features (e.g., Fourier transform features (Zorić et al., 2022)). Handcrafted features are more suitable for situations where the background is not complex, the anomalies are relatively obvious or there is only a single relevant defect

* Corresponding author.
E-mail addresses: hexujie@stu.hit.edu.cn, hexujiee@126.com (X. He), jinjinghit@hit.edu.cn (J. Jin), chenduoo@126.com (D. Chen), zhoucangtian@126.com (C. Zhou).

https://doi.org/10.1016/j.engappai.2024.109058
Received 21 March 2024; Received in revised form 7 July 2024; Accepted 25 July 2024
Available online 31 July 2024

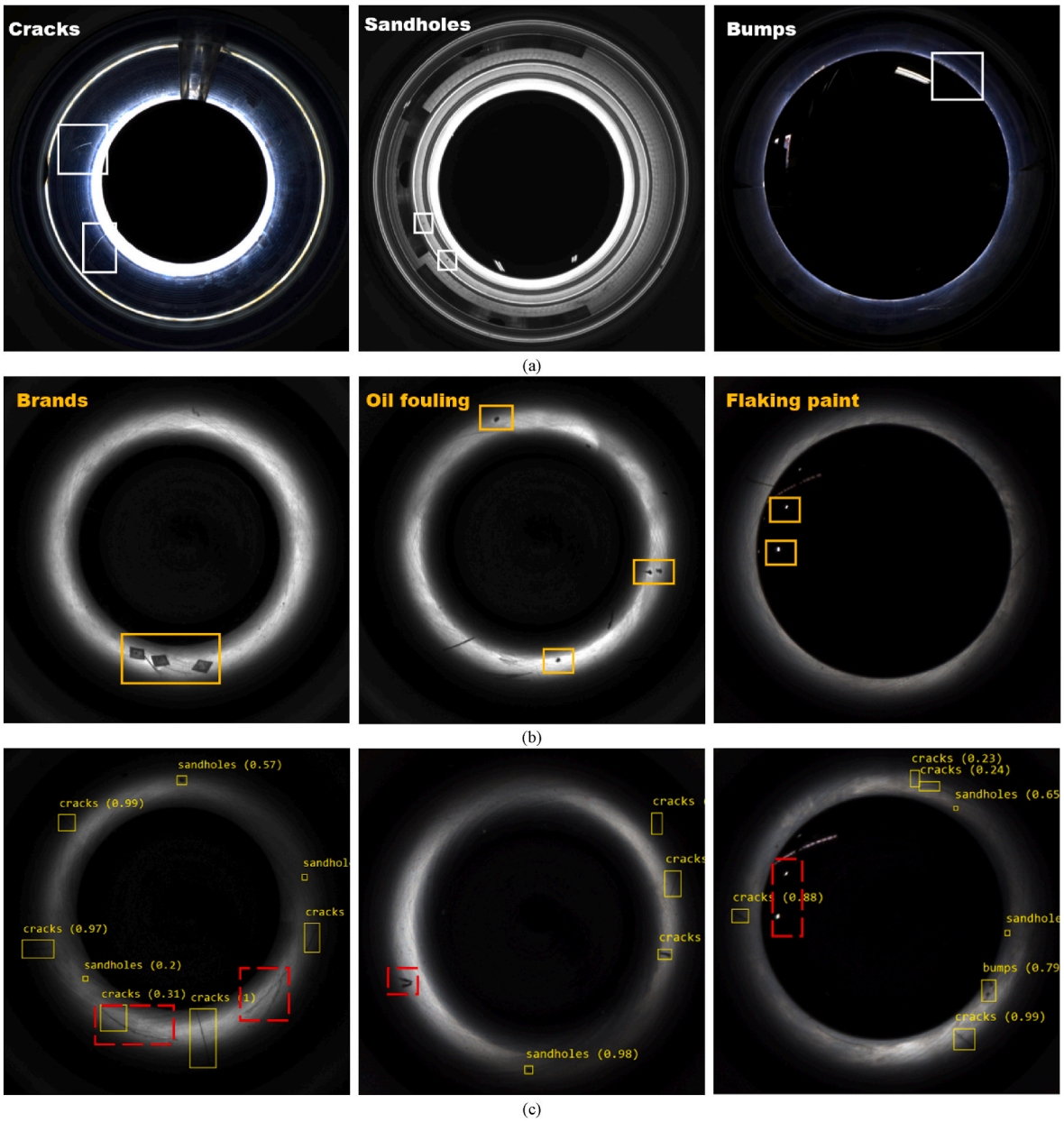

Fig. 1. Challenge illustration. Images (a) and (b) present six types of cylinder bore defects, including cracks, sandholes, bumps, brands, oil fouling and flaking paint. The first three types of defects are acquired in the experimental phase, while the last three types appear when bores are used in a factory. The FasterRCNN (Ren et al., 2017) and YOLO5[1] detection results are presented in (c). It can be noted that 1) new types of defects, e.g., oil fouling and sandholes, are similarly shaped but still cannot be detected owing to their size, which is the same as that of flaking paint; 2) cracks in the experimental phase tend to be small or moderate-sized, and when cracks become larger, detectors become unstable because of the preset anchor sizes.

[1] https://github.com/ultralytics/yolov5/releases.

category. However, the defects that arise in real production processes are often affected by many uncontrollable factors, which present great challenges to handcrafted feature-based methods. Additionally, advancements in deep learning technology in recent years have been beneficial for the rapid development of various computer vision applications, and an increasing number of researchers favor deep learning for universal and high-precision defect detection models. As a result, numerous excellent deep feature-based defect detection methods have been proposed. Current deep feature-based methods can be divided into three types in accordance with their implementation: classification-based defect detection methods (Dong et al., 2022; Yu et al., 2023), segmentation (anomaly)-based defect detection methods (Shao et al., 2022; Liu et al., 2022), and bounding box-based defect detection methods (MA et al., 2022; Wang and Cheung, 2022). From another perspective, deep feature-based defect detection methods can also be divided into three subtypes in accordance with the supervisory approach used: self-supervised methods (Xu et al., 2022; Zou et al., 2022), semisupervised methods (Shao et al., 2022; Kim et al., 2023; Gao et al., 2020) and unsupervised methods (Guo et al., 2022; Taherkhani et al., 2022).

Nonetheless, current high-precision defect detection methods for detecting engine cylinder bore defects face two challenges, as shown in Fig. 1:

- *Inadaptability to previously seen defects in cylinder bores.* Different materials (such as cast iron, aluminum alloy, nickel-based alloy, and

steel) may be used in manufacturing engine cylinder bores. These materials are subject to different stress and thermal expansion effects, resulting in significant differences in the sizes of defects and the degree of damage across different cylinder bores. A consequence is models that are not perfectly suited for accurately detecting defects belonging to previously described categories.

- *Inadaptability to unseen defects in cylinder bores.* Because of the use of different materials, the manufacturing processes used, such as coating technologies and heat treatment processes, tend to vary as well; consequently, new and unpredictable categories of engine cylinder bore defects can also appear during manufacturing, thus making it impossible for closed-set defect detectors to accurately locate and detect them.

To address the above problems, we propose RHG-Detector. Based on a foundation model, RHG-Detector is a ***r**obust* defect detection method with ***h**igh **g**eneralizability* for cylinder bores. Specifically, to address the inadaptability to unseen defects in cylinder bores, we exploit the strong generalization performance of the GDINO foundation model (Liu et al., 2023) and propose a generalization enhancer by cascading a box filter, a region extractor and a defect discriminator in a single pipeline to output defect detection results differentiated by both location and category and thus extend defect detection from a closed set to an open set. To address the inadaptability to previously seen defects in cylinder bores, we propose a cross-modality aggregator that amalgamates detection results from multiple modalities to further mitigate missed detection. Experiments on our publicly released datasets demonstrate the superior performance of the proposed method over current state-of-the-art (SOTA) methods; specifically, our method achieves a mAP@50 score of 45.2, an increase of ~6 compared to that of FasterRCNN + FPN and an increase of ~3 compared to that of YOLOv7. Additionally, our proposed defect discriminator (DeDi series) outperforms current SOTA discriminators and achieves an optimal defect classification accuracy of 92.0%.

To summarize, the contributions of this paper include the following:

1) RHG-Detector, a robust defect detection method with high generalization potential for cylinder bores, is proposed. This method extends the possible targets of defect detection from a closed set to an open set.
2) To address unseen types of defects, a generalization enhancer (GE) is proposed. GE comprises a box filter, a region extractor and a defect discriminator arranged in a cascaded fashion.
3) To address previously seen types of defects, a cross-modality aggregator (CmA) is proposed for aggregating detection results obtained from different modalities.
4) A defect classification dataset (HIT-EngDC) and an expanded defect detection dataset (HIT-EngDD2) were collected and meticulously annotated using our PX1 (Fig. 4). These datasets cover 5 types of cylinder bore defects (such as cracks with significant variations in scale, sandholes in superhigh-resolution images, cluttered rust, and unseen defect types exclusive to the test set) and will be made publicly available to facilitate future work.
5) Extensive experiments conducted on HIT-EngDC and HIT-EngDD2 demonstrate the SOTA performance of the proposed method in addressing not only known defects but also defects belonging to new categories.

The remainder of this paper is organized as follows: In Section 2, we review the literature on defect detection. In Section 3, we introduce the details of the RHD-Detector. In Section 4, we report the results of the experimental evaluations and comparisons, and we further discuss these results in Section 5. Finally, in Section 6, we conclude this study and suggest possible future work.

2. Related work

Since the current deep feature-based defect detection methods are more accurate and stable than handcrafted feature-based methods, deep feature-based defect detection methods are gradually becoming mainstream models for defect detection tasks. Therefore, this section provides an overview of the current deep feature-based defect detection methods. This section details the current deep feature-based defect detection methods from two perspectives: the form of defect presentation and the model design layout. Moreover, since foundation models have rapidly progressed this year and are well generalized to many downstream tasks, we elaborate on current visual foundation models at the end of this section.

2.1. Form of defect presentation

In terms of the forms of defect presentation, current defect detection methods based on deep features can be categorized into classification-based defect detection methods (Dong et al., 2022; Yu et al., 2023), segmentation-based defect detection methods (Shao et al., 2022; Liu et al., 2022), and detection-based defect detection methods (MA et al., 2022; Wang and Cheung, 2022). Furthermore, classification-based defect detection methods complete binary identification for product quality to be inspected; sometimes, it may be necessary to further output specific defect categories for defective products. Segmentation-based defect detection methods refer mostly to anomaly detection, which involves segmentation of defective product regions. Detection-based defect detection methods eventually present defective product regions in a bounding box fashion, which is currently more commonly used in many defect detection applications. For example, Gao et al. (2020) proposed a pseudolabel CNN (PLCNN) to classify defects on steel surfaces in a semisupervised manner. They achieved 90.7% accuracy on the NEU dataset (Song and Yan, 2013) and 86.72% accuracy in real-case applications. Huang et al. (2021) proposed a CNN-based classification network that integrates an attention mechanism into a CNN-based feature extractor to address defects on hot-rolled steel strips specifically. Shang et al. (2023) proposed assembling a transformer mechanism, defect-aware module and graph position encoding into a single pipeline to segment blade and tool wear defects. Yang et al. (2023) trained a model to obtain a robust classification hyperplane from normal and simulated abnormal samples, thereby making it possible to segment anomaly regions in the inference phase. Li et al. (2023) released the you only look once (YOLO)-attention strategy based on YOLO4 (Bochkovskiy et al., 2020), which has been demonstrated to be effective for detecting defects in wire and arc additive manufacturing. Liu et al. (2023) introduced a multiscale context defect detection module to address defects of different sizes on strip steel. The module yields a mAP@50 of 79.4% on the NEU dataset.

2.2. Layout of the model design

Depending on the form of defect presentation, model layouts for defect detection have different paradigms. The prevailing layout involves designing a reasonable feature extractor and cascading a certain number of full connections for classification-based defect detection models to realize the identification task. Currently, the feature extractor is achieved by adopting off-the-shelf high-precision CNNs, such as VGG (Simonyan and Zisserman, 2015), ResNet (He et al., 2016), and DenseNet (Huang et al., 2017); lightweight network architectures, such as ShuffleNet (Ma et al., 2018), MobileNet (Howard et al., 2019), and GhostNet (Tang et al., 2022); and more popular transformer architectures, such as the ViT (Dosovitskiy et al., 2020), PVT2 (Wang et al., 2022) and Swin (Liu et al., 2021) series. For example, Deitsch et al. (2019) used a modified VGG19 network to identify solar panel image defects at 300 × 300 resolution and achieved an 88.42% classification accuracy, thereby exceeding that of multiple handcrafted features. Liang

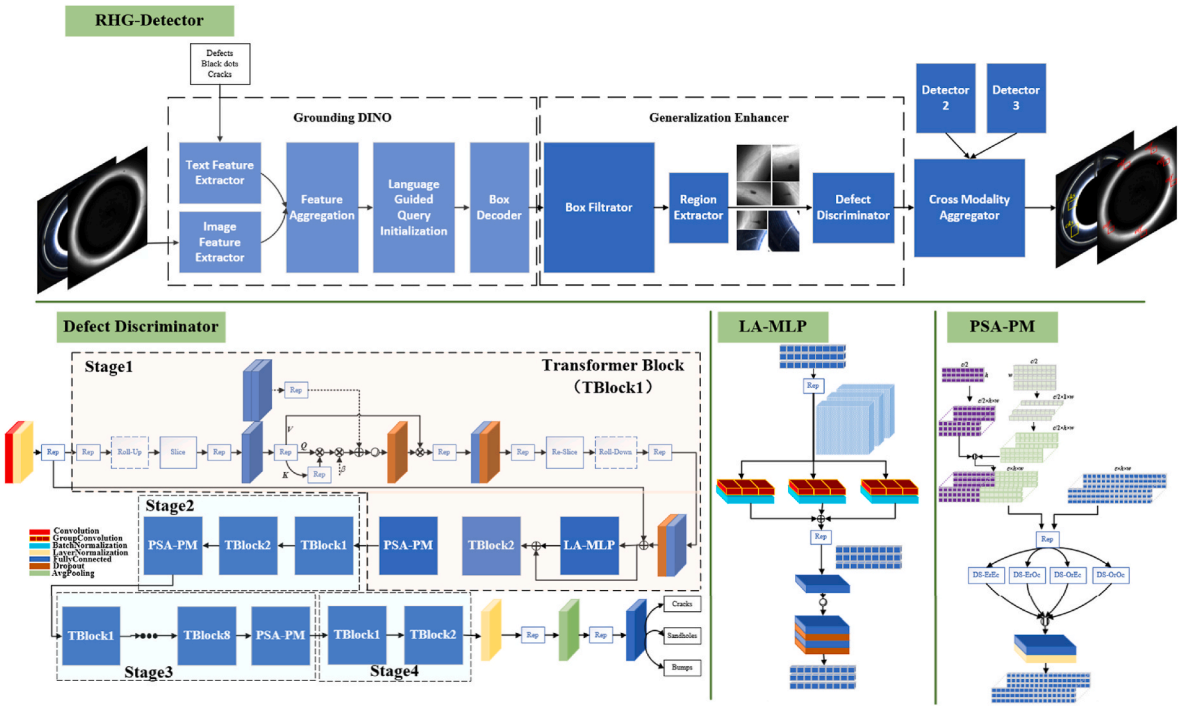

Fig. 2. Deployment of RHG-Detector. The upper part illustrates the overall deployment of the proposed RHG-Detector, which detects defects in cylinder bores in a cascaded manner. The grounding DINO (GDINO) foundation model, prompted with generic prompts ***P,*** is first employed to locate defects of all types (including seen and unseen categories). The defective regions located by GDINO are subsequently passed into the proposed GE, which comprises a box filtrator, a region extractor, and a defect discriminator to transform ***P;*** the GE is prompted into domain-specific defect terms. To further reduce missed detection, GE is back articulated with a CmA, thus enabling the retention of the respective advantages of different methods for addressing different defect categories.

et al. (2019) proposed a bottle inkjet code defect detection method based on ShuffleNetV2 and achieved 99.88% classification accuracy on online inkjet code detection equipment in the plastic container industry. Zhang et al. (2019) proposed a VGG-like plain CNN-based network to address weld defects and achieved an accuracy of 99.38%. The current mainstream paradigm for segmentation-based defect detection models is to design encoder-decoder structures (such as proposed by Ronneberger et al. (2015) and Chen et al. (2018)), in which the encoder also adopts the same feature extractor as the classification-based defect detection models to encode input images. The decoder consists of a series of upsampling units and skip connections to recover the obtained encoded features to the same dimensions as the input. For example, Kaji et al. (2022) used acquired point cloud data to detect defects generated during metal additive manufacturing by using a U-Net structure with unsupervised and supervised learning techniques, thereby achieving an overall accuracy and average intersection-over-union (IoU) of 91.3% and 83.3%, respectively. Tao et al. (2022) proposed a dual-Siamese framework to solve anomaly detection problems (such as reconstruction and inpainting problems) and achieved promising performance. Other attempts, such as that proposed by Tabernik et al. (2020), also possess similar paradigms. The structure of detection-based defect detection models usually comprises two segments (a feature extractor and a detection head), where the former is consistent with the previous section. The detection head regresses the extracted features to coordinate information and corresponding categories. Many current studies apply this paradigm to real applications. Current detection-based methods can be further subdivided into one- and two-stage detectors, where one-stage detectors are highly efficient, while two-stage detectors are highly accurate. For example, Hou et al. (2021) and Zhang et al. (2023) used FasterRCNN-based (Ren et al., 2017) two-stage detectors to address defects in fabric, steel surfaces, medical products, and weld joints. In contrast, Liu et al. (2023) and Zhao et al. (2023) used YOLO-based one-stage detectors to address defects in electronics, sewers, welds, steel surfaces, and noise barriers.

2.3. *Visual foundation model*

Successful foundation models have emerged in computer vision this year. The creation of these foundation models facilitates many downstream tasks. The foundation model, first proposed in Bommasani et al. (2021), refers to a model that can adapt well to many downstream tasks after being trained on an ultralarge-scale dataset. Foundation models are usually based on prompt engineering (PE) (Gu et al., 2023) and commonly use text-, bounding box–, point- and mask-based prompts. Prompt engineering (Zhou et al., 2022), especially the successful introduction of text-based prompts (Wang et al., 2023), has furthered the generalization ability of visual foundation models, thereby greatly advancing the development of the visual zero-shot (Sun et al., 2021) learning field. Among the foundation models applied for segmentation, the model proposed by Kirillov et al. (2023) is arguably the most famous this year. The model uses a hybrid form of text, mask, and point prompts trained on 11 M images in a data collection loop manner and segments for arbitrary targets in the universal world. The emergence of the segment anything model (SAM) benefits numerous downstream tasks, such as medical image segmentation (Ma et al., 2023) and remote sensing image segmentation (Chen et al., 2023). FastSAM (Zhao et al., 2023) further improves SAM in terms of processing efficiency and finally obtains a new model that runs ~60x faster than SAM with comparable accuracy. Another representative model, SegGPT (Wang et al., 2023), which starts from Painter (Wang et al., 2023) and regards segmentation as a random coloring problem based on contextual information, was proposed. Most of the current foundation models for object tracking are based on SAM. For example, Yang et al. (2023), which introduces the corresponding tracking head to track objects, is based on SAM. Currently, there are two mainstream paradigms for foundation object detection models: the foundation detection models extended from CLIP

(Zhong et al., 2022) and open-set detection by introducing a text module on one of the off-the-shelf detectors, such as grounding DINO (GDINO) (Liu et al., 2023). GDINO achieves zero-shot detection by introducing a language-guided module based on the transformer-based detector DINO. Experiments show that compared to CLIP-based models, GDINO achieves better detection results on both closed- and open-set detection datasets. Therefore, we take GDINO (a visual foundation model) as the basic framework to enhance the generalization ability, and improvements are made accordingly. RHG-Detector is the first attempt to use a visual foundation model in engine cylinder bore defect detection.

3. RHG-detector

3.1. Overview

In response to the issues identified in the introduction, specifically those regarding the current high-performing detection methods still struggling with the adaptability to detect both seen and unseen defect categories in engine cylinder bores, we propose RHG-Detector. In this subsection, we provide an overview of the structural architecture of the proposed RHG-Detector. In RHG-Detector, a foundation model is deployed for the first time to address cylinder bore defects. Specifically, RHG-Detector comprises an open-set foundation model, a generalization enhancer and a cross-modality aggregator arranged in a cascaded fashion. The overall structure of RHG-Detector is shown in the upper half of Fig. 2. Specifically, the defect detection problem addressed here can be formulated as $\boldsymbol{D} = \sum_{\mathrm{o}} \left\{ \left(\boldsymbol{i}_j^o, b_j^o, c_j^o \right)_{j=1}^{N} \right\} \in \boldsymbol{I} \times \boldsymbol{B} \times \boldsymbol{C}$, where $o = |\{o^{train}, o^{test}\}|$, $\boldsymbol{D}$ denotes the dataset of current engine cylinders, $\boldsymbol{io}\ j \in \mathbb{R}^{\mathrm{H}\times\mathrm{W}\times 3}$ represents the input image where a defect is to be detected, $\boldsymbol{bo}\ j \in \mathbb{R}^{d1\times 4}$ represents the defect location in $\boldsymbol{io}\ j$, and $co\ j \in \boldsymbol{C}$ is the corresponding defect category. Obviously, $\boldsymbol{C}^{\mathrm{train}} \subseteq \boldsymbol{C}^{\mathrm{test}}$ in the current situation. Specifically, to improve the generalization ability of the model for unseen (new) defect categories, we first input the image $\boldsymbol{io}\ j \in \mathbb{R}^{H\times W\times 3}$ for detection into an open-set detection method termed GDINO and cyclically prompt GDINO with different text prompts $\boldsymbol{P} = \{p_1, p_2 \ldots, p_n\}$. Since this open-set detection method cannot encode domain-specific terms for defects in cylinder bores, we initialize $\boldsymbol{P}$ with generic texts, such as {"*defects*", "*black dots*", "*abnormal areas*"}, which can be easily understood by foundation models. These generic texts are incompatible with the final intention of defect detection, which requires results in the form of domain-specific defect terms; therefore, we cascade a generalization enhancer (GE) after GDINO. GE comprises three submodules: a box filter, a region extractor, and a defect discriminator. GE projects the results into domain-specific defect categories. Furthermore, to reduce missed detections of the types of defects previously seen by the model, using the seen (existing) defect data to train multiple models is necessary to acquire certain detection capabilities and ultimately aggregate the defect detection results from these different modalities. For this purpose, we also deploy a cross-modality aggregator (CmA) on the back end to aggregate the defect detection results. The proposed RHG-Detector ultimately shows promising generalization performance and can properly generalize to defects in both seen and unseen categories. In the remainder of this section, we elaborate in detail only on the modules proposed in this paper, namely, the generalization enhancer and cross-modality aggregator.

Algorithm 1. BoundingBox Filtration

Input: Detected boxes $\mathscr{B} \in \mathbb{R}^{d1\times 4}$,
Image height h, image width w
Corresponding confidence $\boldsymbol{c} \in d_1 \times 1$,
Area threshold δ, Similarity threshold ζ
Output: Filtrated boxes $\mathscr{B}^{BF} \in d_2 \times 4$
Initialization: $\mathscr{B}_o \leftarrow \{\}$, $\mathscr{B}^{BF} \leftarrow \{\}$, $\boldsymbol{c}_o \leftarrow \{\}$
for i=0 **to** d_1 **do** % Area filtration
  if ($\mathscr{B}[i][2]$- $\mathscr{B}[i][0]$)×($\mathscr{B}[i][3]$-$\mathscr{B}[i][1]$)/(h×w)<δ **then**
    $\mathscr{B}_o$.add($\mathscr{B}[i]$)
    $\boldsymbol{c}_o$.add($\boldsymbol{c}[i]$)
  end if
end for
cs, $cs\_idx \leftarrow$ sort($\boldsymbol{c}_o$, descending) % IOU-based NMS filtration
$\mathscr{V}_1 \leftarrow [1] \times \boldsymbol{c}_o$.length, $id \leftarrow 0$, $\ddot{Y} \leftarrow$ True
while $\ddot{Y}$ **do**
  if $\mathscr{V}_1[cs\_idx[id]]$=1 **then**
    cur_box $\leftarrow \mathscr{B}_o[cs\_idx[id]]$
    $\mathscr{V}_1[cs\_idx[id]] \leftarrow$ 'saved'
    for $k \leftarrow 0$ **to** $\boldsymbol{c}_o$.length **do**
      if k!=$cs\_idx[id]$ **then**
        if $\mathscr{V}_1[k]$!= 'deleted' **then**
          similarity $\leftarrow$ IOU(cur_box, $\mathscr{B}_o[k]$)
          if similarity>ζ **then**
            $\mathscr{V}_1[k] \leftarrow$ 'deleted'
          end if
        end if
      end if
    end for
  end if
  $id \leftarrow id + 1$
  if id= $\boldsymbol{c}_o$.length **then**
    $\ddot{Y} \leftarrow$ False
  end if
end while
return $\mathscr{B}^{BF} \leftarrow \mathscr{B}_o$.where($\mathscr{V}_1$='saved')

3.2. Generalization enhancer (GE)

1) ***Box filtrator***: Due to the use of different texts cyclically prompting open-set GDINO, some overlaps exist between the semantics; for example, "*defects*" and "*black area*" have semantic overlaps, giving rise to detection overlaps under different text prompts. To address this problem, we first filter all the detection results $\mathbf{B} = \{\mathrm{b}_1, \mathrm{b}_2 \ldots, \mathrm{b}_{d1}\}$ by using a box filtrator (BF) composed of two segments. Since GDINO produces a few misdetection results that are output in the form of bounding boxes greater than half of the image size, the first attempt is area filtration (AF). AF filters all bounding boxes according to the area and filters out the detections with areas larger than a given threshold δ. The second segment filters the boxes obtained by AF by the nonmaximum suppression (NMS) method under a given threshold ζ, thereby further eliminating the box overlaps resulting from semantic overlaps. The overall bounding box filtration (BF) is given by:

$$\boldsymbol{B}^{BF} = \mathscr{T}_{NMS}^{BF}\left(\left\{b_j \in \boldsymbol{B} \;\middle|\; \mathscr{T}_{AF}^{BF}\left(\boldsymbol{B}_j\right) \le \delta\right\}, \zeta\right) \in \mathbb{R}^{d_2\times 4} \tag{1}$$

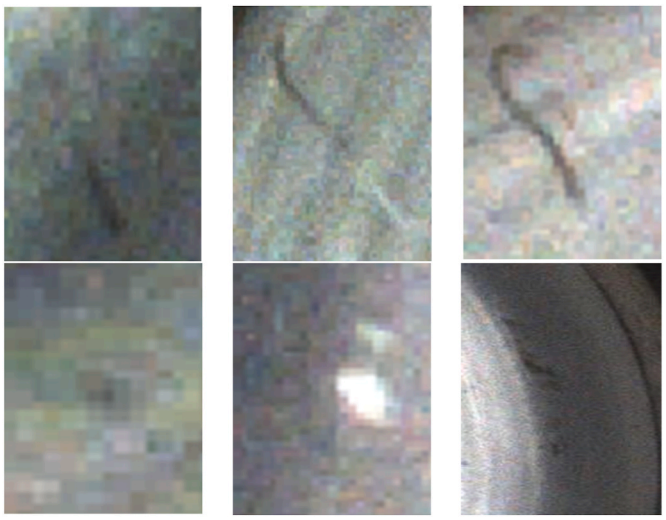

Fig. 3. Image samples of HIT-EngDCs (Section 4). We selected representative specimens from our HIT-EngDC dataset in the following order: oil fouling, cracks, sandholes, flaking paints and bumps. As seen, the clipped image tends to render at a low resolution, introducing difficulties for subsequent defect discrimination.

where b_j represents the GDINO output and d_2 indicates the number of detections after filtration. ΦBF AF and ΦBF NMS represent the area filtration function and IoU-based NMS filtration function, respectively. The overall BBF method is shown in Algorithm 1. Therefore, after BBF filtration, the surplus boxes are further suppressed due to semantic overlaps and misdetection.

2) ***Region extractor (REor):*** To refine the detected regions with domain-specific defect categories, our method proposes a region extractor. The REor cuts out each defective region detected by DGINO from the original image to form a new defective image gallery $\boldsymbol{G} = \{\boldsymbol{g}_1, \boldsymbol{g}_2 \ldots, \boldsymbol{g}_{|G|}\}$, and we perform image augmentation. The first image augmentation method we adopt to increase data diversity is the horizontal flip method, which is given by:

$$\phi_H = \begin{cases} a_2 = W - a_1 \\ b_2 = b_1 \end{cases} = \begin{cases} a_2 = -1 \times a_1 + 0 \times b_1 + W \\ b_2 = 0 \times a_1 + 1 \times b_1 + 0 \times W \end{cases} = \begin{bmatrix} -1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \\ W \end{bmatrix} \tag{2}$$

where (a_1, b_1) and (a_2, b_2) represent the original pixel location and corresponding horizontally flipped location, respectively, and W represents the image width. To prevent specific dimensional data from heavily influencing the overall data and to make the following defect discriminator possess stable ability against outliers, we also normalize the images $\boldsymbol{I} \in \mathbb{R}^{H \times W \times 3}$ according to Eq. (3):

$$\phi_N = \frac{\boldsymbol{I} - \mu}{\sigma} \tag{3}$$

where μ and σ represent the mean and standard deviation, respectively, which are given by:

$$\mu[k] = \left[\sum_{i=1}^{H} \sum_{j=1}^{W} I(k,i,j) \right] \Big/ H \times W \tag{4}$$

and

$$\sigma[k] = \sqrt{\sum_{i=1}^{H} \sum_{j=1}^{W} [I(k,i,j) - \mu[k]]^2 \Big/ H \times W} = (H \times W)^{-\frac{1}{2}} \| I(k) - \mu[k] \|_2 \tag{5}$$

Additionally, we execute a rescaling operation over the images by using image bicubic interpolation as follows:

$$\boldsymbol{I}_{\mathrm{BI}}(x,y) = \sum_{m_1=-1}^{2} \sum_{m_2=-1}^{2} \boldsymbol{I}(\overline{x} + m_1, \overline{y} + m_2) \cdot Ke(h_1 - m_1) \cdot Ke(h_2 - m_2) \tag{6}$$

where $\overline{x} = \lfloor x \rfloor, \overline{y} = \lfloor y \rfloor, \overline{h}_1 = x - \lfloor x \rfloor, \overline{h}_2 = y - \lfloor y \rfloor$ and Ke represents the kernel function adhering to the following:

$$Ke(h) = \begin{cases} 1.5|h|^3 - 2.5|h|^2 + 1 & \text{if} |h| \le 1 \\ -0.5|h|^3 + 2.5|h|^2 - 4|h| + 2 & \text{if } 1 < |h| \le 2 \\ 0 & \text{if} |h| > 2 \end{cases} \tag{7}$$

where the variable h is the independent variable of the cubic kernel function.

Eventually, after executing the three image augmentations, the defective images are stored in the defective image library $\boldsymbol{G} = \{\boldsymbol{g}_1, \boldsymbol{g}_2 \ldots, \boldsymbol{g}_{|G|}\}$ for use by the subsequent defect discriminator.

3) ***Defect discriminator (DeDi):*** Due to the defects of cylinder bores occupying a very small proportion of the entire image, these defects appear at a lower resolution when locally magnified, as shown in Fig. 3, thus presenting challenges for defect identification. We define this issue as *defect category discrimination under low-resolution images*. To address this problem, we argue that adding more information back to the model is necessary. We adopt the current best-performing SwinV2 as the defect discriminator. Two improvements are made accordingly.

Location-aware multilayer perceptron (MLP) module (LA-MLP). We first locate the MLP to introduce spatial information and propose the LA-MLP. The MLP introduced in SwinV2 is plain structured. We locate the features from the vector space back to the image space at the entrance of each MLP and project. Utilizing stacked kernels of different sizes while also being as large as possible is beneficial for better feature formulation. Therefore, in the image space, we finally adopt group convolution with different receptive fields to further formulate features. The extracted features are consequently projected back to the vector space to perform the regular MLP forward extraction. The whole LA-MLP is given by:

$$\mathrm{LA-MLP}(\boldsymbol{x}) = \mathrm{MLP}\left[\mathrm{Proj}_{m \to v} \left(\sum_{i=1}^{3} \mathrm{GConv}(\mathrm{Proj}_{v \to m}(\boldsymbol{x})) \right) \right] \in \mathbb{R}^{l_2 \times q_2} \tag{8}$$

where GConv indicates that the group convolution is given by:

$$\mathrm{GConv}(\mathbf{X}) = \sum\nolimits_i \mathbf{X} * \mathbf{W}_i \tag{9}$$

and $\mathrm{Proj}_{v \to m}$ and $\mathrm{Proj}_{v \to m}$ are the conversion functionalities from vector space to image space and from image space to vector space, respectively. The MLP adheres to the following:

$$\begin{aligned} &\mathrm{MLP}(\mathbf{X}) = \Lambda^{(L)}\left(\mathbf{X}^{(L-1)}\mathbf{W}^{(L)} + \varepsilon^{(L)}\right) \\ &= \Lambda^{(L)}\left(\Lambda^{(L-1)}\left(\ldots\Lambda^{(1)}\left(\mathbf{X}\mathbf{W}^{(1)} + \varepsilon^{(1)}\right)\ldots\right)\mathbf{W}^{(L)} + \varepsilon^{(L)}\right) \end{aligned} \tag{10}$$

where Λ is the activation function (a rectified linear unit is adopted), L is the layer index, and $\mathbf{W}$ represents the learnable weight.

Position self-adaptive patch merging module (PSA-PM). PatchMerging in SwinV2 carries out feature extraction only over the current image in a row- and columnwise manner to 2 × downsample feature maps and flatten the downsampled feature maps into one-dimensional vectors. Obviously, spatial relations between features are abandoned concurrently. To address this problem, we introduce the positional self-adaptation mindset and propose the positional self-adaptive PM module. The structure of the PSA-PM module is shown in Fig. 2. Specifically, we introduce a self-adaptive form of the position encoding module that

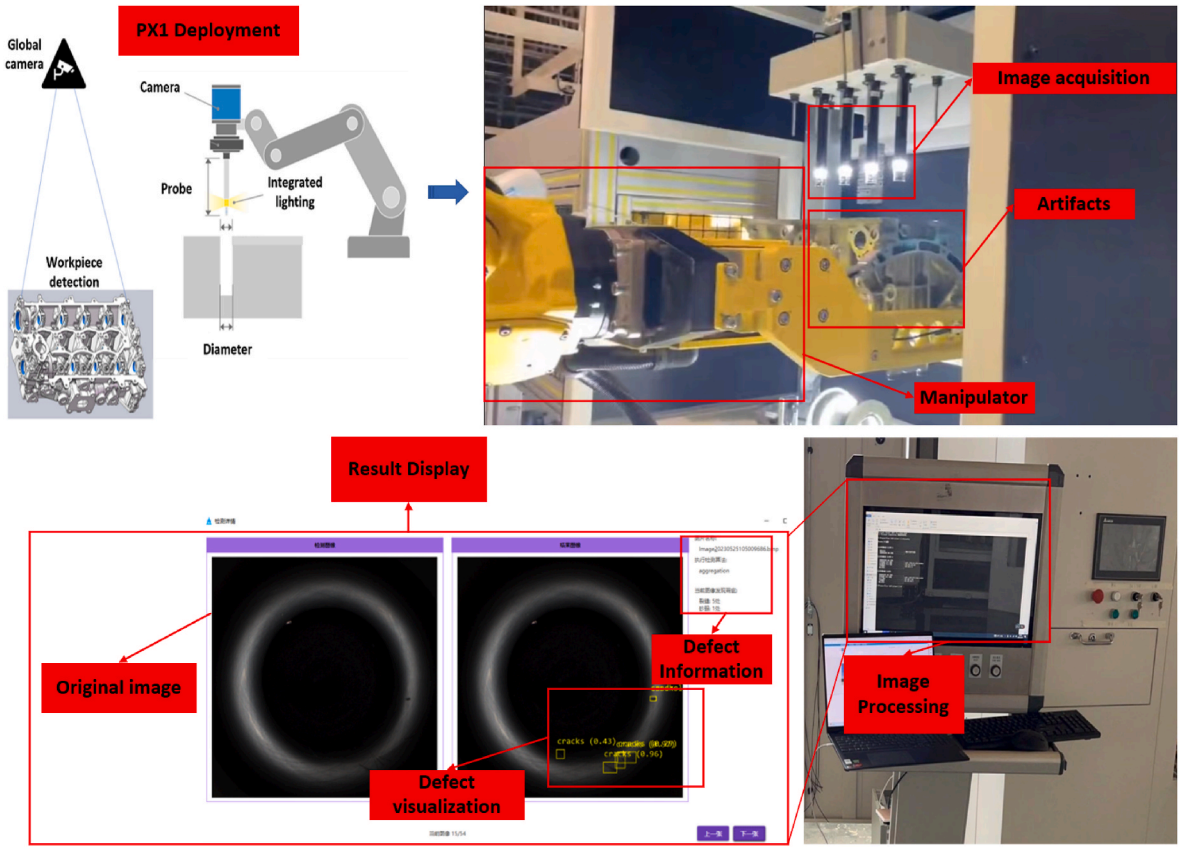

Fig. 4. Illustration of the cylinder bore condition surveillance system (PX1).

corresponds to the feature map before performing downsampling. Then, we feed the module into the downsampling unit after fusing the module with the feature map to carry out 2 × downsampling. The entire PSA-PM module is given by:

$$\mathrm{PSA-PM}(\boldsymbol{x}) = \left\{\mathrm{PM}\left[\rho \cdot Concat(\mathbf{W}\boldsymbol{x}, \mathbf{W}\boldsymbol{y})|_{dim=-1}\right] + \mathrm{PM}(\boldsymbol{x})\right\} \in \mathbb{R}^{I_1 \times q_1} \quad (11)$$

where $\boldsymbol{x}$ is the input feature, $\mathbf{W}$ represents the learnable weight, ρ indicates the modulating coefficient and Concat is the concatenation operation.

3.3. Cross-modality aggregator (CmA)

To 1) ensure that the model possesses a certain defect detection ability for the current defect detection task, training current off-the-shelf detectors with handy data in a supervised manner is necessary. 2) Because most of the detectors suffer from missed detection problems, we introduce a CmA at the back end of the model. CmA takes the detection results of different detectors (modalities) $\Phi = \{\Phi_1, \Phi_2 \ldots, \Phi_{|\Phi|}\}$ as input and aggregates the results. Ultimately, CmA outputs the aggregated results (defect location + defect category) as the output of the whole RHG-Detector. For simple implementation, we adopt Hungarian matching as the aggregation mode. Since the generalized IoU (GIoU) is insensitive to scale, the GIoU can focus on both overlapping regions and other nonoverlapping regions, which can better reflect the contact ratio between two input detections. We finally choose the GIoU-based Hungarian matching mode. CmA is implemented by adhering to:

$$\mathscr{G}\left(\boldsymbol{\Phi}_i^k, \boldsymbol{\Phi}_j^s\right) = \left(\frac{\Phi_i^k \cap \Phi_j^s}{\Phi_i^k \cup \Phi_j^s} - \frac{\left|\mathrm{C}_{\Phi_i^k, \Phi_j^s} - \left(\Phi_i^k \cap \Phi_j^s\right)\right|}{\mathrm{C}_{\Phi_i^k, \Phi_j^s}}\right) \quad (12)$$

where $\mathrm{C}_{\Phi i\ \Phi j}$ are the smallest closure regions formed by Φ_i and Φ_j, respectively. The defect detection results obtained from each modality (old category modality + new category modality) are matched according to the similarity quantified by the GIoU. If the current similarity is higher than a given threshold (Θ), the current detection results are considered to characterize an identical defect; one of the results (the one with higher confidence $\hbar_i$) will be saved. Alternatively, they are considered different defects, and all of them are saved. After the above aggregation, the advantages of each method modality in addressing different defects can be retained, thus aiding in further reducing missed detections.

3.4. Loss penalty

The loss function is composed mainly of two parts, one applied to each method modality and the other applied to the defect discriminator. Specifically, the overall loss function is $\checkmark = \checkmark_{\mathrm{DeDi}} + \checkmark_{\mathrm{CmA}}$, which penalizes DeDi and CmA. $\checkmark_{\mathrm{DeDi}}$ is composed of a classification loss. $\checkmark_{\mathrm{CmA}}$ comprises a classification loss, a regression loss and a confidence loss. In implementation, the cross-entropy loss is employed as the classification loss and the confidence loss, which is given by:

$$\mathscr{L}_{cls|\mathrm{conf}}(q_i, \widetilde{q}_i) = -\sum_{q_i}^{Q} \widetilde{q}_i \cdot \log\left(\frac{\exp(q_i)}{\sum_{q_i'}^{Q} \exp(q_i')}\right) \quad (13)$$

where q_i and $\widetilde{q}_i$ indicate the class predicted and class annotated, respectively. The complete intersection-over-union (CIoU) loss is employed as the regression loss, which adheres to:

$$\mathscr{L}_{reg}(\boldsymbol{c}, \widetilde{\boldsymbol{c}}) = \sum_i \mathscr{L}_{reg}(\boldsymbol{c}_i, \widetilde{\boldsymbol{c}}_i) = \sum_i \left[1 - \frac{\boldsymbol{c}_i \cap \widetilde{\boldsymbol{c}}_i}{\boldsymbol{c}_i \cup \widetilde{\boldsymbol{c}}_i} + \left(\frac{\tau}{l}\right)^2 + \chi_i \varsigma_i\right] \quad (14)$$

where $\boldsymbol{c}_i$ and $\widetilde{\boldsymbol{c}}_i$ are the locations $\{x_i, y_i, w_i, h_i\}$ predicted and annotated, respectively; τ is the distance between the center of $\boldsymbol{c}_i$ and the center of

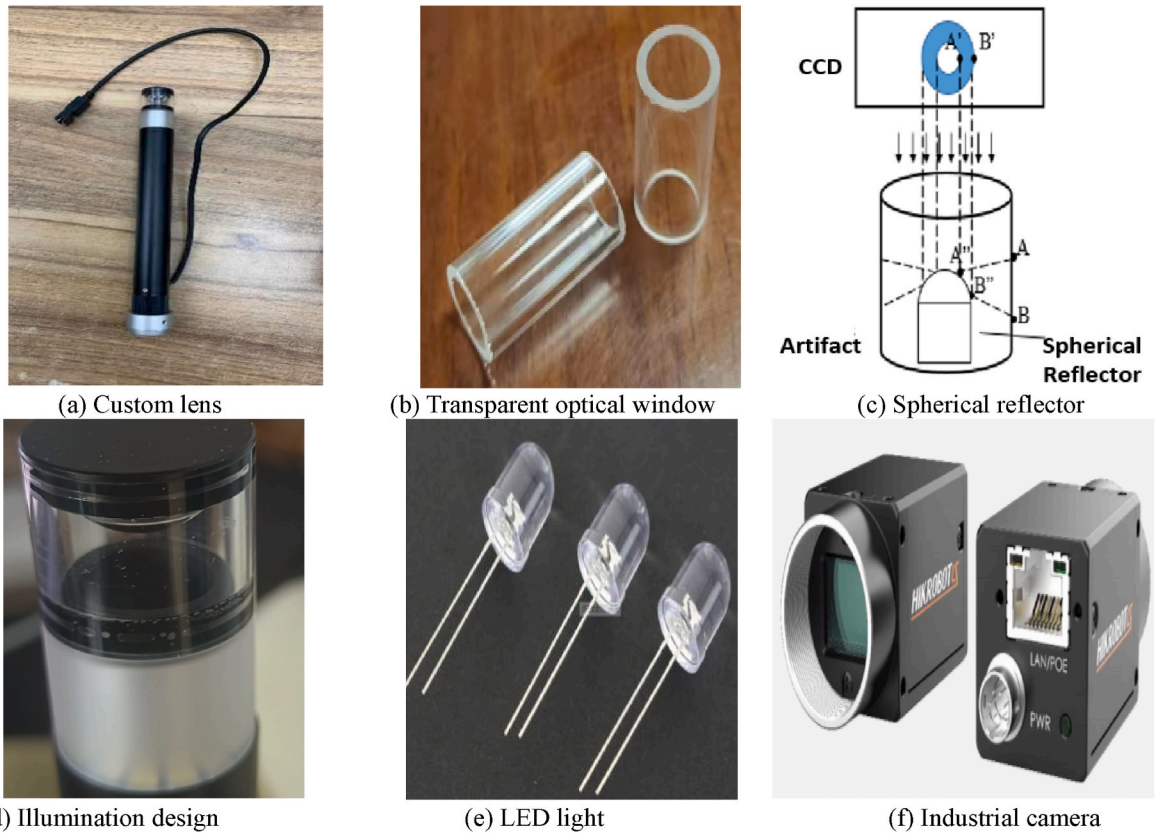

Fig. 5. Imaging acquisition devices.

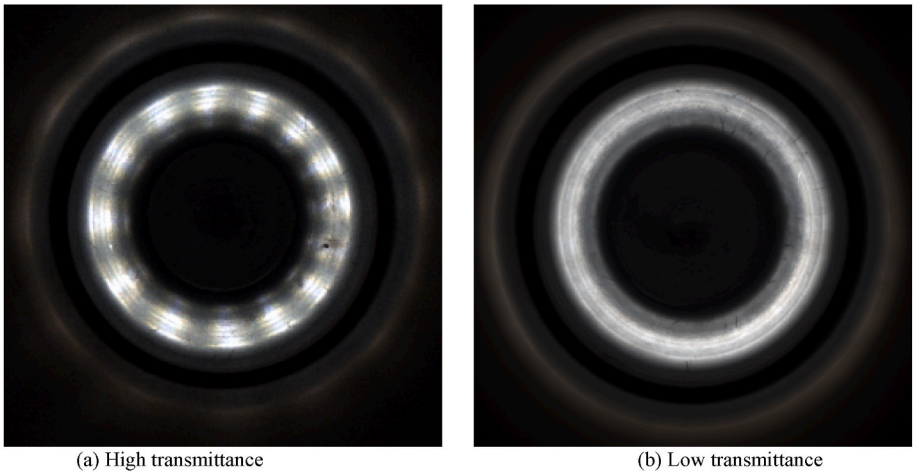

Fig. 6. Image quality illustration.

Table 1
Specifics of the camera and lens.

Camera		Lens	
Resolution	2448 × 2048	Length	228.7 mm
Effective area	2/3″	Focal length	36.98 mm
Pixel size	3.45 μm× 3.45μm	Work distance	169.8 mm
Frame rate	24fps	Lens distortion	<1%
Color	RGB	Effective area matched	2/3″
Shutter type	Global	Len interface	C-Mount
Camera interface	GigE		

$\widetilde{\boldsymbol{c}}_i$; and ι is the diagonal length of the smallest outer rectangle of the region where $\boldsymbol{c}_i$ overlaps $\widetilde{\boldsymbol{c}}_i$. χ_i and ς_i are given by:

$$\varsigma_i = \frac{4}{\pi^2}\left(\arctan\frac{\boldsymbol{c}_i^w}{\boldsymbol{c}_i^h} - \arctan\frac{\widetilde{\boldsymbol{c}}_i^w}{\widetilde{\boldsymbol{c}}_i^h}\right)^2 \tag{15}$$

$$\chi_i = \frac{\varsigma_i}{1 - \left(\boldsymbol{c}_i \cap \widetilde{\boldsymbol{c}}_i / \boldsymbol{c}_i \cup \widetilde{\boldsymbol{c}}\right) + \varsigma_i} \tag{16}$$

4. Experiments

In the experimental section, we will proceed to unveil the

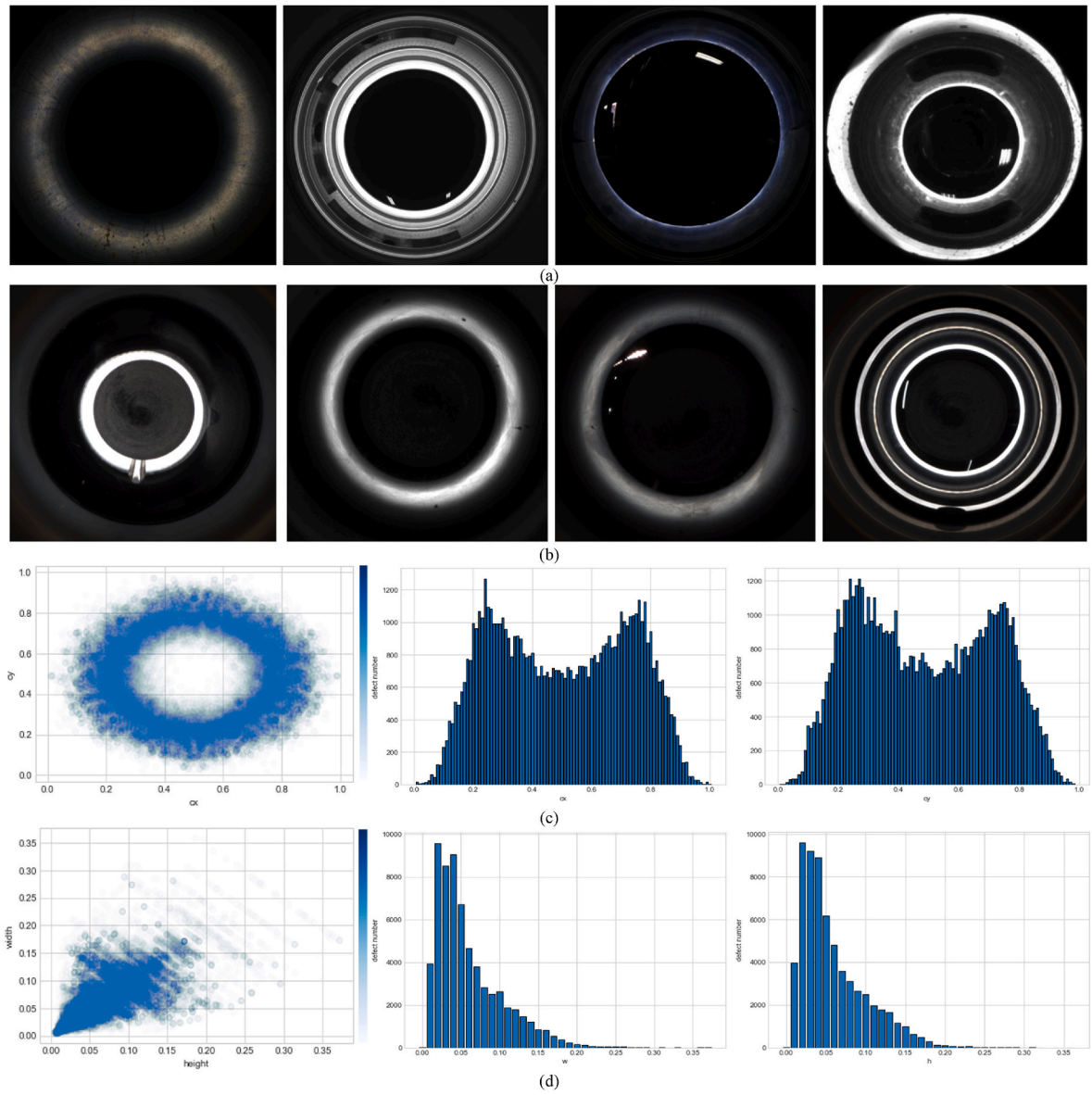

Fig. 7. Dataset illustration. Our images are captured from nearly 200 different types of cylinder bores that vary in shape and size and cover almost all types of defects commonly found in the inner walls of the cylinder bores of automobile engines. As shown in (c) and (d), defects in cylinder bores usually tend to be tiny and occupy only a few pixels, thus making defect detection challenging. Cx and cy represent the normalized central location of each defect, while height and width correspond to the size of the bounding box.

Table 2
Details of HIT-EngD.

Dataset	Type	Samples	Totality	Cracks	Sandholes	Bumps	Flaking paints	Oil fouling
HIT-EngDD1	Train	7040	49,390	30,436	18,518	436	×	×
	Test	1760	12,346	7609	4628	109	×	×
HIT-EngDD2	Train	7040	49,390	30,436	18,518	436	×	×
	Test	1996	14,772	8798	5012	304	77	571
HIT-EngDC	Train	2384		897	997	299	66	125
	Test	1022		384	428	128	28	54

composition of the apparatus; introduce the dataset, training strategies, and evaluation metrics; and conduct a series of corresponding experiments (including comparisons with SOTA algorithms and a series of ablation studies on the proposed method), thereby thoroughly assessing the performance of the proposed model. Moreover, the experimental results are discussed in the corresponding subsections.

4.1. System composition

The overall product quality inspection equipment consists of four parts, as shown in Fig. 4, namely, the overall image acquisition system, servo-controlled system, central processing unit and display unit. Servo-controlled systems are responsible mainly for locating and focusing on specific workpieces by controlling specific manipulators, while the

Table 3
Comparison of the experimental results. The metrics used in this section are AR and mAP^{50}. The superscripts S, M and L indicate small, medium and large defects, respectively. '*' indicates that YOLO5L and YOLO7X are adopted as two methods for detecting CmA. mAPS is acquired by calculating mAP by using only small defects that are less than 32^2; mAPM is acquired for middle defects that are greater than 32^2 but less than 96^2; and mAPL is acquired for large defects that are greater than 96^2 (mAPS, mAPM and mAPL are analogous to ARS, ARM and ARL, respectively). Additionally, we recorded the floating point operations (FLOPs) of each method in the first column. The numbers 3 and 6 following DETR indicate the number of encoders and decoders used, respectively.

Models	FLOPs(G)	ARS	ARM	ARL	mAPS	mAPM	mAPL	mAP@50
FRCNN_FPN(+ResNet50)	269.0	17.10	29.30	70.20	13.40	25.70	64.30	39.00
FRCNN_FPN(+Mobile3)	65.7	9.65	11.50	25.64	2.73	3.57	11.97	15.70
SSD	30.6	9.91	18.29	59.40	6.31	13.85	53.41	33.55
RetinaNet	256.6	14.37	23.84	66.61	8.71	18.43	60.99	37.44
YOLO5S	16.3	18.60	31.30	77.00	16.00	29.00	73.00	42.18
YOLO5L	114.0	20.80	34.90	83.20	18.70	33.20	80.10	42.52
YOLO7T	13.1	16.10	28.30	69.10	13.00	25.10	65.00	41.01
YOLO7X	189.4	20.30	34.40	80.60	17.8	32.30	79.70	42.27
DETR_3	73.4	3.94	11.37	40.59	0.36	1.50	13.42	10.20
DETR_6	73.6	13.02	22.89	66.01	3.25	10.39	50.81	30.91
Deformable DETR	157.4	22.20	35.72	80.67	14.42	29.45	76.26	42.81
DINO	214.3	23.87	39.13	83.81	13.49	32.42	79.40	43.26
RHG-Detector*	331.4	22.60	37.70	81.20	18.60	33.20	78.00	**45.19**

central processing unit mainly controls the imaging acquisition systems and manipulators, executes integrated software methods and transmits corresponding instructions to the display to display the status of the current workpiece in real time. We provide additional details on the imaging acquisition systems in the following.

The imaging acquisition systems include mainly a camera, lens, light source and reflector, as shown in Fig. 5.

1) *Lens:* Detecting defects in cylinder bores requires a detection device that can adapt to different cylinder bore diameters; thus, a custom lens is chosen, as shown in Fig. 5(a). Moreover, the light source and reflector are also integrated on the lens to achieve simultaneous illumination and imaging. To further ensure the isolation of the spherical reflector from the outside during imaging, considering that the impurities in the outside air contaminate the spherical reflector and affect the final imaging effect, a layer of transparent optical window is added to the outside of the spherical reflector to protect the internal optical components, as shown in Fig. 5(b). Due to the strong deflection ability of the spherical reflector to light, as shown in Fig. 5(c), a larger range of images can be captured in cylinder bores during single-image processing. Therefore, a spherical reflector is finally selected and integrated into our custom lens.

2) *Illumination:* The light source is designed to follow the lens and continuously illuminate the inner wall of the cylinder bore. After preliminary investigations, the lighting method that resulted in the best imaging quality was ultimately selected; accordingly, a customized LED light source was integrated on the lens; the light emitted by the light source can uniformly illuminate the inner wall of the cylinder bore after scattering by a diffuse light guide plate, as shown in Fig. 5(d)(e). Moreover, we investigated the different transmittances of the diffuse reflector, as shown in Fig. 6. In the high transmittance mode, the light irradiated by the LED light source cannot be completely dispersed, thus resulting in a glare on the final image; the glare is not conducive to observing the imaging effect. In contrast, in the low transmittance mode, the illuminated light is fully guided by the diffuse reflector and is uniformly formed on the inner wall of the cylinder bore. A spot is no longer generated by glare in the image; consequently, the image can meet the imaging requirements. Finally, the low transmittance mode was selected for our design.

3) *Camera:* Considering that PX1 is designed to meet the detection requirements for a workpiece diameter range of ~ϕ18 mm, a large effective area is chosen—that is, a 2/3″ specification (corresponding to a sensor size of 8.8 × 6.6 mm)—to achieve abundant and clear imaging on the plane. Additionally, when using a spherical reflector to achieve imaging, the expected imaging effective range is 50% of the outer ring, and the diameter of the imaging ring does not exceed the camera's effective area width (6.6 mm). The resolution of imaging with a spherical reflector is determined by:

$$\Xi = \frac{\pi\ell}{\pi\ell_h} \times \Xi_p \tag{17}$$

where l and l_h are the workpiece diameter and imaging ring diameter, respectively, and Ξ and $\Xi_p = 3.45\mu m{\cdot}pixel^{-1}$ represent the imaging resolution and pixel resolution, respectively. The inner ring image with the worst resolution can still reach 0.12556 mm•$pixel^{-1}$. That is, the size of the smallest defect that can be resolved is ~0.25112 mm, which is suitable for the current detection task. Therefore, a 5-megapixel industrial camera is used for high-definition image acquisition. The appearance and specific parameters of the camera are shown in Fig. 5(f) and Table 1.

4.2. Dataset

All the data used in the experiments were acquired by using the product surveillance system (PX1) developed by our own research achievements, as shown in Fig. 4.

1) ***HIT Engine Detection Dataset (HIT-EngDD)***: The dataset we acquired by our PX1 is named HIT-EngDD. The first version of HIT-EngDD was released at HIT-EngDD1. HIT-EngDD1 is acquired from more than 200 different engine cylinder bores containing three common defects on the inner bore wall: cracks, sandholes, and bumps, as shown in Fig. 7(a). Details of HIT-EngDD1 are shown in Table 2. However, in actual manufacturing, due to various random factors or even uncontrollable factors, defects are often not confined to the above three types. Therefore, other new types of cylinder bore defects appear when our PX1 is used. We amalgamated the newly acquired data into a HIT-EngDD1 test set without annotation and then obtained HIT-EngDD2. The newly introduced defect categories in HIT-EngDD2 include oil fouling and flaking paints, as shown in Fig. 7(b). Details of HIT-EngDD2 are listed in Table 2. Overall, HIT-EngDD2 has the following characteristics:
   - There are 5 common types of engine cylinder bore defects: cracks, sandholes, bumps, oil fouling, and flaking paints.
   - The size of the cracks varies within a wide range.
   - The sandholes, some of which occupy only a few pixels, are tiny in super high-resolution images.
   - There are cluttered defects caused by oil splashes and rust.
   - Defect detection problems occur under dark illumination.

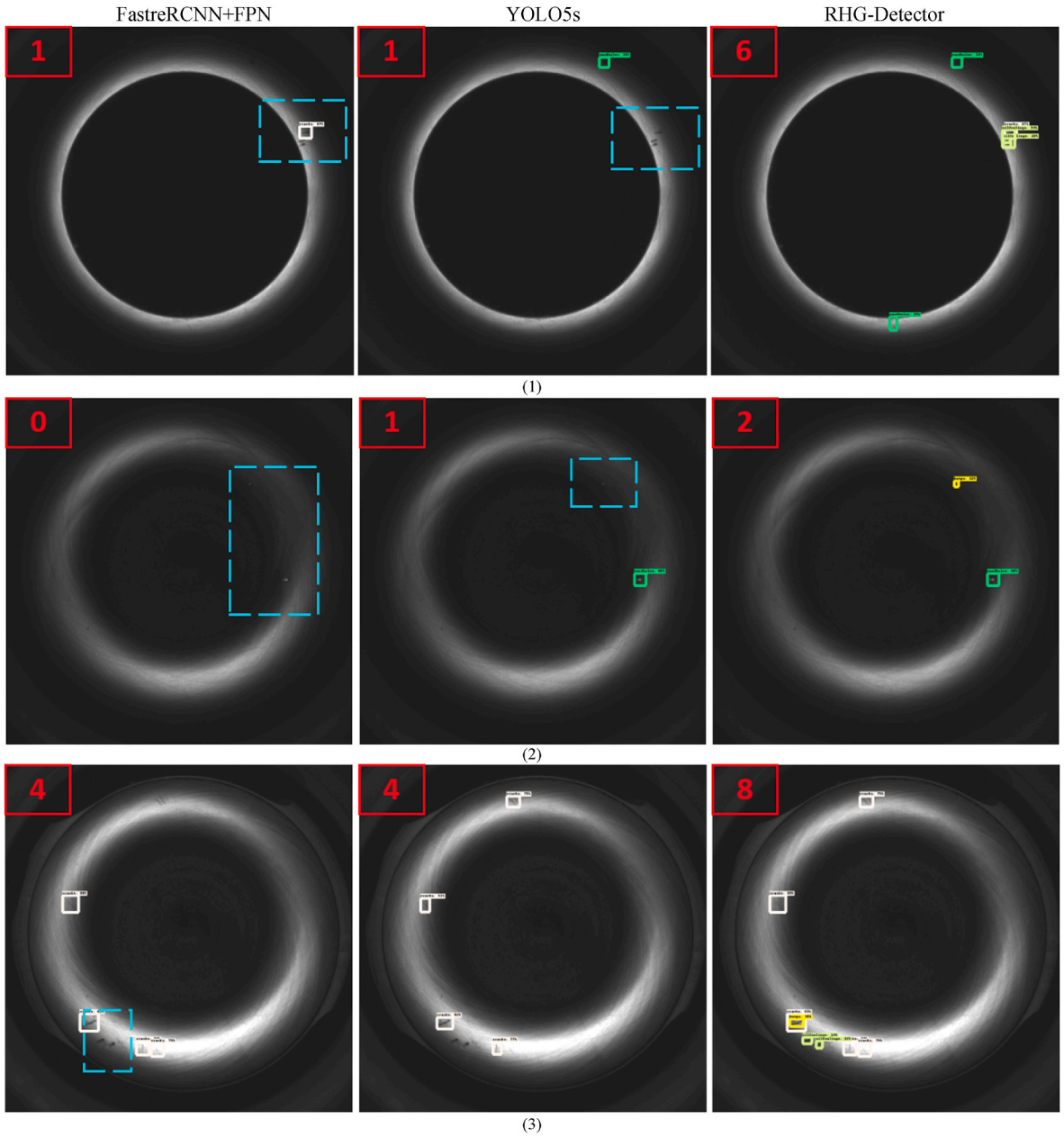

Fig. 8. Qualitative results of the comparison experiments. Nine groups of qualitative experimental results are presented above. Each group of results is obtained from FasterRCNN integrated with FPN, YOLO5S and the proposed RHG-Detector. The number in the red box in the upper left corner of each image indicates the number of defects detected by the current method. Bounding boxes of different colors represent defects of different categories. The solid-line boxes indicate defects detected by the detector, while the dashed-line boxes represent defects missed by the detector.

- The defect categories are inconsistent with the training and testing sets.

Therefore, it is more challenging to conduct experiments on HIT-EngDD2.

2) ***HIT Engine Classification Dataset (HIT-EngDC)***: To further rectify the defect categories, a new dataset for the engine cylinder bore defect classification task, HIT-EngDC, was collected and meticulously annotated. Specifically, HIT-EngDC was intercepted from HIT-EngDD2 according to the location provided by the corresponding annotations. Therefore, HIT-EngDC also contains five defect types: cracks, sandholes, bumps, oil fouling, and flaking paints, as shown in Fig. 3. Details of HIT-EngDC are listed in Table 2.

4.3. Evaluation metrics

Precision, recall, average recall (AR) and the comprehensive metric mAP (mean average precision) are adopted to evaluate the performance of the methods; these metrics are given by:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{18}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{19}$$

$$\text{AR} = = 2 \times \int_{0.5}^{1} R_i d(R_i) \tag{20}$$

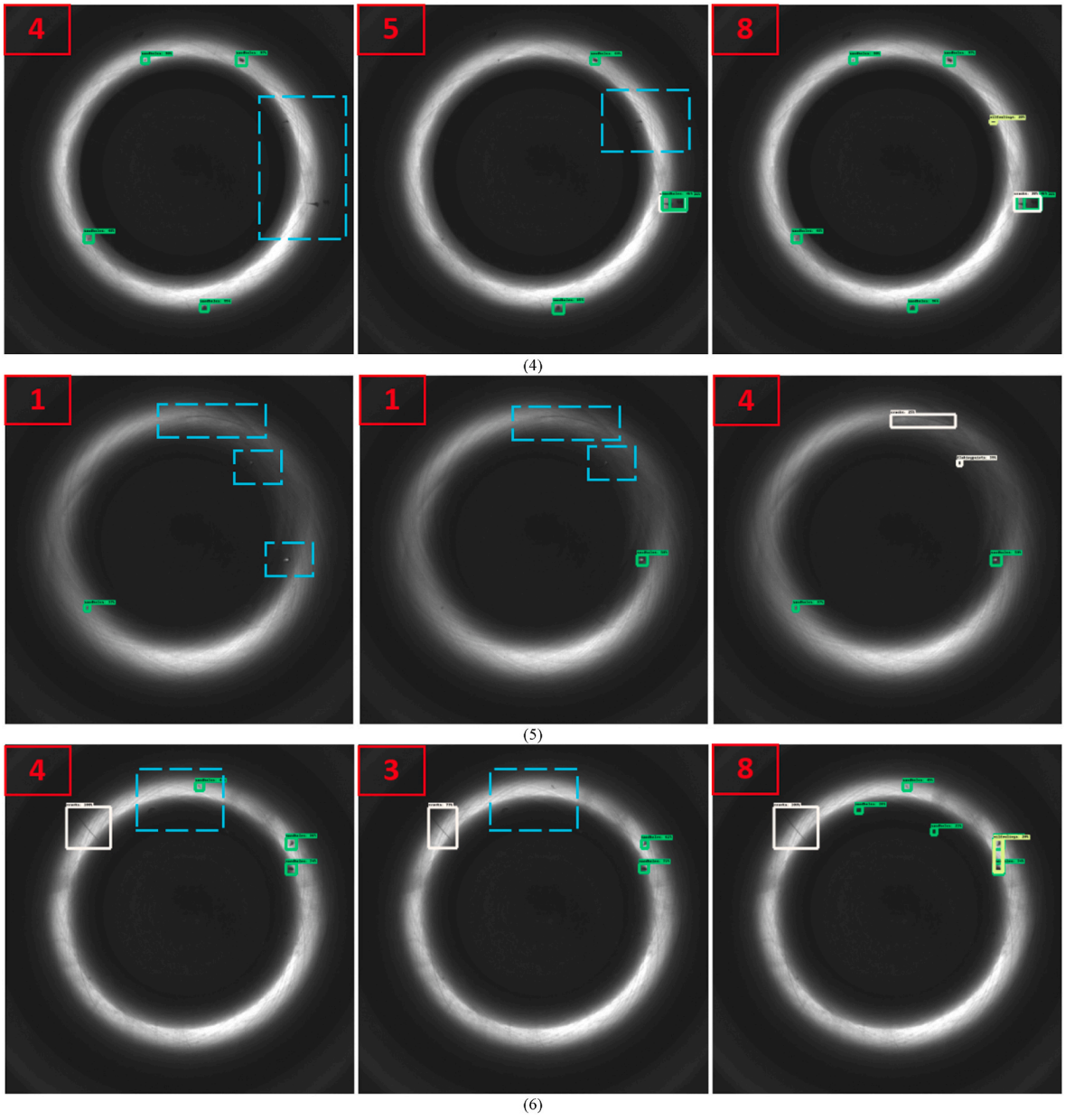

Fig. 8. (*continued*).

$$\mathrm{mAP} = \frac{\sum_{i=1}^{N} \mathrm{AP}_i}{N} = \frac{\sum_{i=1}^{N} \int_0^1 P_i(R_i) d(R_i)}{N} \tag{21}$$

where TP, FP, and FN indicate a defect that is truly detected as a defect, a nondefect that is detected as a defect, and a defect that is not detected, respectively. R in Eq. (21) denotes the metric recall, and P(R) indicates the curves formed by the metric precision and recall. The integral intervals are acquired by setting different thresholds by using IoU similarity from 0 to 0.5 and 0 to 1. N indicates the number of defect classes.

4.4. Comparison experiments

Experimental design: To verify the advancement of the proposed method, we first conducted a group of comparison experiments on HIT-EngDD2. The methods used in the comparison experiments are currently all SOTA detection-based methods, including pure CNN structures (e.g., FasterRCNN + FPN (Ren et al., 2017), SSD (Liu et al., 2016), YOLO5[1], and YOLO7 (Wang et al., 2023)) and pure transformer structures (e.g., DETR (Carion et al., 2020), Deformable DETR (Zhu et al., 2021) and DINO (Zhang et al., 2022)). The dataset used in this section of the experiment is HIT-EngDD2.

Training protocols: For a fair comparison, for all methods, we keep the same training hyperparameters, including the training epoch set to 100, the initial learning rate set to 2e-4, and the decay set to 2e-5 at 75 epochs. For the CNN-structured methods, we use Adam to optimize all learnable weights; for the transformer-structured method, we choose to optimize all learnable weights by using AdamW. The parameter weight decay in AdamW is set to 1e-4. The image size is set to 650 × 650, and the batch size is set to 16 with GPU memory. Due to convergence issues with the DETR model, we extend its training by an additional 100 epochs, thereby resulting in 200 training epochs. Additionally, we apply the same learning rate decay strategy at the 150th epoch for DETR.

Quantitative results: The quantitative experimental results are listed in Table 3. Notably, 1) our proposed RHG-Detector model yields an optimal comprehensive detection performance of 45.19 in terms of

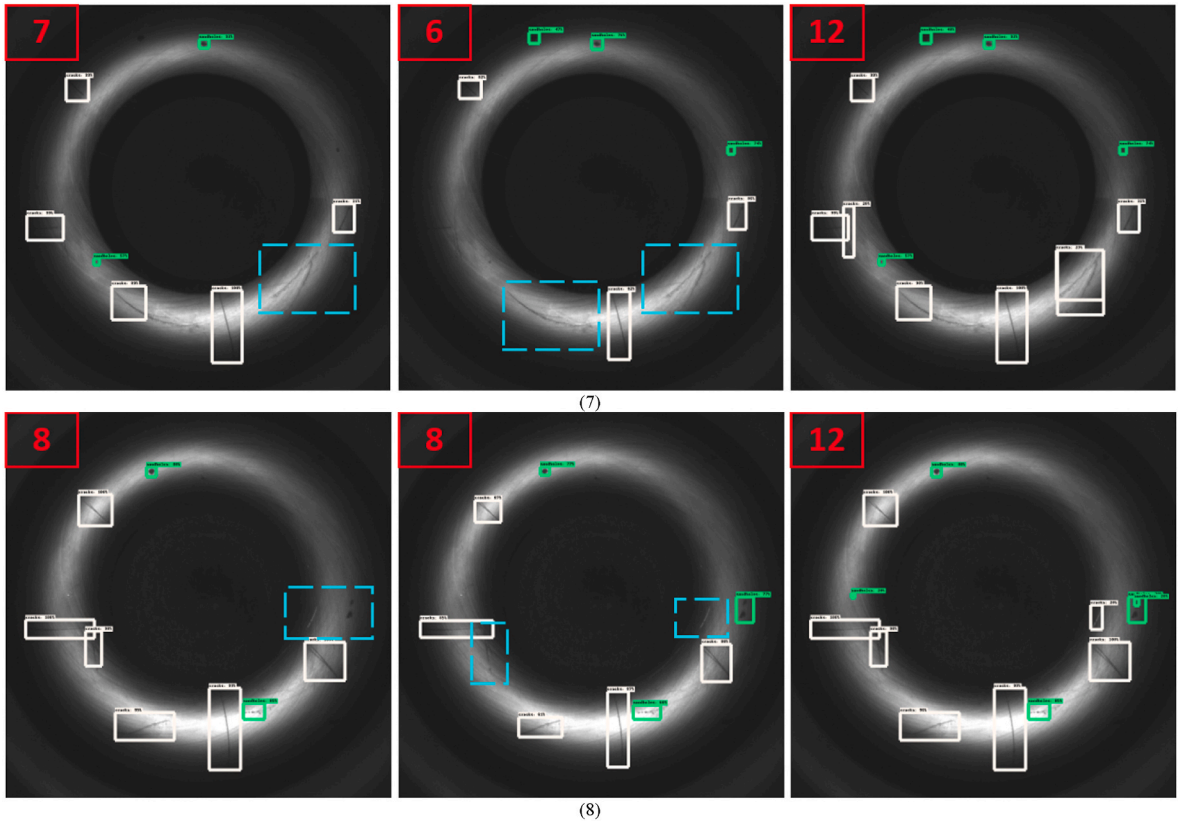

Fig. 8. (*continued*).

Table 4
Quantitative results on the defect discriminator. Methods utilized in these experiments vary from lightweight to intensive and from CNN based to transformer based.

Models	Acc(%)	Models	Acc(%)	Models	Acc(%)
EffNet	81.0	PVT2-b0	87.1	PVT2-b3	78.9
ShuffleNetV2	90.5	PVT2-b1	87.7	PVT2-b4	81.6
MobileNetV3-Large	90.6	PVT2-b2	88.9	PVT2-b5	89.2
ResNet18	88.4	SwinV2-T	87.1	DeDi-T	90.8
ResNet34	91.2	SwinV2-S	88.3	DeDi-S	**92.0**
ResNet50	85.9	SwinV2-B	90.0	DeDi-B	91.8
ResNet101	82.4	SwinV2-L	89.0	DeDi-L	91.7

mAP@50; 2) except for DETR, transformer-based methods generally outperform CNN-based methods. The defect detection performance of DINO achieves a suboptimal value of 43.26 in terms of mAP@50, while Deformable DETR achieves a slightly lower mAP@50 value of 42.81; 3) the performance of one-stage detection methods, such as the YOLO7 series, is comparable to that of the YOLO5 series regarding the current defect detection task; 4) when comparing FasterRCNN + FPN with a ResNet50 backbone and FasterRCNN + FPN with a MobileNetV3 backbone, network lightweighting (which decreases accuracy from 39.00 to 15.70 in terms of mAP@50) greatly affects the final defect detection accuracy; and 5) as the first column of Table 3 shows, it is evident that models based on the transformer architecture, such as DINO and RHG-Detector, are more computationally intensive than those based on the CNN architecture due to the quadratic relationship between the computational complexity of the self-attention mechanism and the sequence length.

Qualitative results: To further qualitatively investigate the ability of the methods to detect defects in engine cylinder bores, we present the qualitative results of FasterRCNN + FPN, YOLO5S and RHG-Detector (aggregated with FasterRCNN + FPN and YOLO5S as method modalities in CmA). Fig. 8 shows eight groups of results yielded from FasterRCNN + FPN, YOLO5 and RHG-Detector. The number in the upper left corner of each figure indicates the number of defects detected. We note that 1) the newly proposed RHG-Detector has better adaptability to new defect categories, thus revealing that RHG-Detector can generalize to the new defect categories and has a more stable detection capability than FasterRCNN + FPN and YOLO5 for the unknown categories of cylinder bore defects, as shown in Fig. 8(1)–(4) (6) (8); 2) the newly proposed RHG-Detector can further reduce the missed detection problem of the models (as shown in Fig. 8(5) (7), both FasterRCNN + FPN and YOLO5 miss large-scale cracks; however, RHG-Detector locates and detects all of them); and 3) RHG-Detector fuses the results of multiple method modalities, thereby retaining the advantages of each algorithmic modality in detecting different defects; thus, RHG-Detector forms a complement to reduce missed detection.

4.5. Study on defect discrimination

Experimental design: To explore the performance comparison between the proposed discriminator DeDi and the current SOTA discriminator, we carried out a group of comparison experiments regarding defect discriminators. The methods used in this section are CNN- and transformer-based methods with SOTA performance, including the ResNet series (He et al., 2016), MobileNet series (Howard et al., 2019), ShuffleNet series (Ma et al., 2018), SwinV2 series (Liu et al., 2021), PVT2 series (Wang et al., 2022), and the proposed DeDi series. Additionally, to further investigate the impact of the proposed two modules (PAS-PM and LA-MLP) on defect classification accuracy, we also conducted a series of ablation experiments on the discriminators. The dataset used in this section of the experiment is HIT-EngDC.

Training protocols: The training epoch is set to 100. The initial learning rate is set to 1e-3 and is decayed by a decay factor of 0.75 at

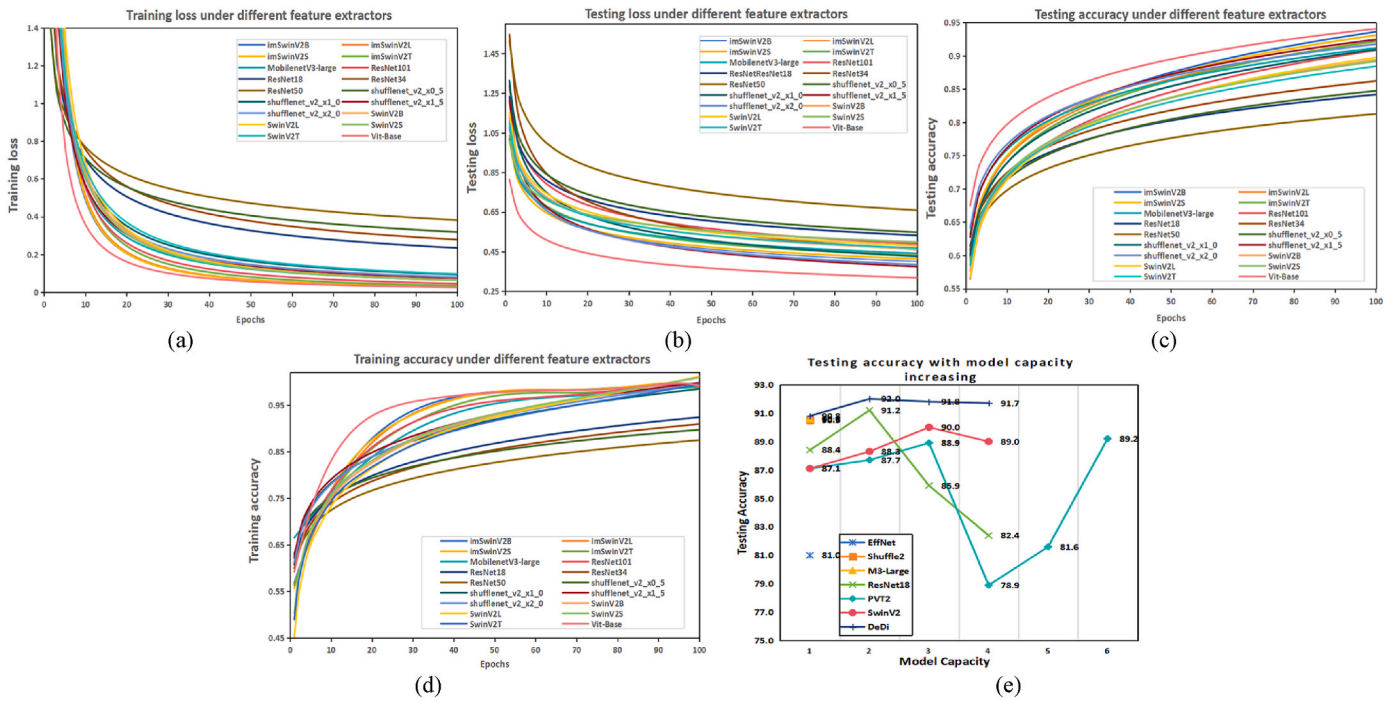

Fig. 9. Dynamic data of all defect discriminators. To more clearly compare the dynamic performance of each defect discriminator, we zoom in on the data of the last 15 epochs in small windows to more clearly visualize the variation in accuracy and loss during training and testing.

Table 5
Discriminator ablation experiments. This section conducts ablation experiments on DeDi-S to study the impact of different proposed modules on accuracy. In this section, "√" indicates the selection of the current module. The optimal experimental results are highlighted in bold black font.

Models	PAS-PM	LA-MLP	Acc(%)
SwinV2-S			88.3
Model_1	✓		89.5
Model_2		✓	91.4
DeDi-S	✓	✓	**92.0**

epoch 85. Similarly, for the CNN-structured methods, we use Adam to optimize all learnable weights, and for the transformer-structured method, we choose to optimize all learnable weights by using AdamW. The parameter weight decay in AdamW is set to 1e-4. The image size is set to 224 × 224, and the batch size is set to 64 with GPU memory.

Quantitative results: The quantitative experimental results are listed in Table 4. Fig. 9(a)–(d) plot the processes of the loss value and accuracy of each model during experimentation. Additionally, to make an intuitive comparison of each model, we plot the relations between the model capacity and the accuracy in Fig. 9(e). We note that 1) as the table shows, DeDi-S yields the optimum defect discrimination accuracy of 92.0%; DeDi-B and DeDi-L achieve suboptimal accuracies with competitive performances of 91.8% and 91.7%, respectively; and PVT2-b3 fails to reach an accuracy of 80% and yields only 78.9% in terms of accuracy. 2) Fig. 9(a)–(d) show that the model accuracies are all positively correlated with the number of training epochs. Except for EffNet, the loss values of the remaining models are negatively correlated with the number of training epochs. Furthermore, the pace of change (the speed of model convergence and accuracy enhancement) of PVT2-b4 and b5 and the ResNet series is slower than that of the remaining methods. 4) Fig. 9(e) clearly shows that, for the current defect discrimination task, the accuracy of different methods tends to saturate or even degrade to varying degrees as the model capacity increases. The ablation experiments in Table 5 further demonstrate significant improvements in defect classification accuracy by employing the two modules proposed in this study, namely, PAS-PM and LA-MLP. Moreover, achieving an optimal accuracy of 92.0% when both modules are used concurrently underscores their complementary roles in enhancing accuracy.

Qualitative results: The qualitative experimental results are shown in Fig. 10. We present five groups of qualitative results according to the methods that yield ideal quantitative results, including MobileNet3-Large (M3), ResNet34 (ResNet), SwinV2-S (Swin), and DeDi-S (DeDi). It can be noted that 1) the proposed DeDi-S possesses a strong discriminative ability for defect categories, such as cracks or sandholes and flaking paints; 2) the randomness of oil splashes results in defects turning into different shapes (e.g., oil fouling with small tails), which interferes with discrimination; and 3) the similarities in the shapes of oil fouling and sandholes (tiny) also interfere with discrimination.

4.6. Study on the CmA unit

To investigate the coupling relationships between the proposed CmA module and similarity threshold Θ to determine the adaptability for the current defect detection task, we accordingly designed another group of comparative experiments. The experiments consisted of two segments, and similarity threshold Θ used in CmA was adopted as a variable. The first step is to investigate the impact of different similarity thresholds on the overall performance of the method, and the second step is to investigate the impact of different similarity thresholds on the detection results of different defect categories. The experimental results are listed in Table 6, Table 7, and Fig. 11. The dataset used in this section of the experiment is HIT-EngDD2.

The impact of similarity threshold Θ on the overall method performance (as shown in Table 6 and Fig. 11(a)) is as follows: 1) the comprehensive defect detection performance (mAP@50) of RHG-Detector first increased and then decreased as the similarity threshold gradually increased; additionally, RHG-Detector's optimal performance reached 45.2% when Θ increased to 0.45 and 0.50; 2) mAP{S, M, L} all tended to first increase and then decrease as Θ increased; additionally, RHG-Detector reached optimal performance when Θ∈[0.35, 0.65]; and

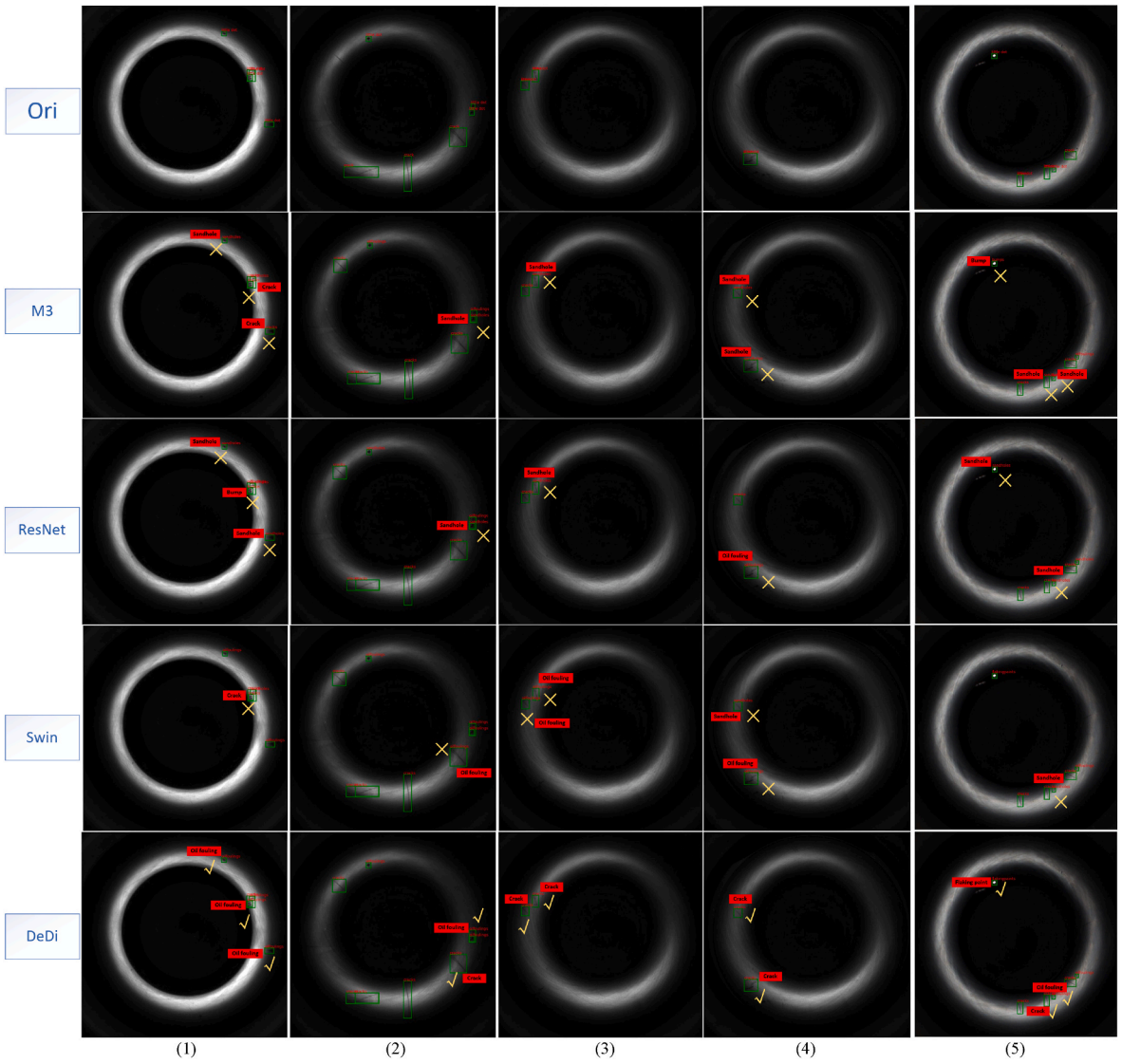

Fig. 10. Qualitative results of defect discriminators. The experimental results of the following four types of discriminators are selected: lightweight CNN-based MobileNetV3, intensive CNN-based ResNet, transformer-based SwinV2 and the proposed DeDi. All the ground truths are presented in the first row. √ and × indicate correct and incorrect categorizations of defect categories, respectively, discriminated by the respective discriminators.

3) AR{S, M, L} positively correlated with Θ as Θ increased; additionally, when Θ = 1.00, AR{S, M, L} reached an optimum of 23.9%, 38.9% and 83.6%, respectively.

The impact of similarity threshold Θ on detecting different defect categories (as shown in Table 7 and Fig. 11) is as follows: 1) as Θ increased, RHG Detector's performance in detecting cracks, sandholes, bumps and oil fouling first increased and then decreased; additionally, the detector achieved optimal performance when Θ∈[0.35, 0.55]; and 2) the detector's performance in detecting flaking paints was unaffected by Θ.

4.7. Study on defect categories

To further investigate the ability of all the models to detect the current types of defects found in cylinder bores, we also conducted a group of studies on defect categories. The methods used in this section are consistent with those used in Table 3. The experimental results are listed in Table 8 and plotted as histograms for better observation, as shown in Fig. 12. The dataset used in this section of the experiment is HIT-EngDD2.

It can be noted that 1) only RHG-Detector is capable of detecting newly emerged defect categories in the testing set, such as flaking paints and especially oil fouling. The open-set detection capability of RHG-Detector reached 11.42 in terms of mAP@50, while the remaining methods all failed to detect any of the newly emerged categories (mAP@50 = 0%). 2) For the most common defect category, such as cracks, RHG-Detector achieves the optimal performance of 90.45 in terms of mAP@50. Likewise, in detecting the bump category, RHG-Detector yields a competitive comprehensive performance of 35.55 in terms of mAP@50; in detecting the sandhole category, YOLO5L achieves the optimal detection performance of 89.37 in terms of mAP@50. 3) The SSD model yields less favorable results for all defect categories: compared to RHG-Detector's performance, SSD's performance in detecting sandholes and cracks is ~29.92% and ~12.86% lower, respectively. Fig. 12 more clearly shows the experimental results for reference.

4.8. Study on text prompts

To further investigate the impact of text prompts on the performance of large models trained with generic scenarios, we have included a set of experiments focusing on text prompts. Specifically, we applied different

Table 6
Quantitative results of the effect of CmA on mAPs and ARs. S, M and L indicate small, middle and large, respectively.

Θ	mAP@50	mAPS	mAPM	mAPL	ARS	ARM	ARL
0.00	0.416	0.177	0.318	0.709	0.214	0.361	0.737
0.05	0.430	0.181	0.326	0.740	0.219	0.370	0.770
0.10	0.438	0.183	0.329	0.757	0.222	0.373	0.788
0.15	0.443	0.185	0.330	0.765	0.224	0.374	0.796
0.20	0.447	0.185	0.331	0.772	0.224	0.375	0.803
0.25	0.447	0.185	0.332	0.776	0.225	0.376	0.807
0.30	0.450	0.185	0.332	0.777	0.225	0.376	0.809
0.35	0.450	0.186	0.332	0.779	0.226	0.377	0.811
0.40	0.450	0.186	0.332	0.780	0.226	0.377	0.811
0.45	**0.452**	0.186	0.332	0.780	0.226	0.377	0.812
0.50	**0.452**	0.186	0.333	0.780	0.226	0.377	0.812
0.55	0.451	0.186	0.332	0.780	0.227	0.378	0.812
0.60	0.451	0.186	0.333	0.782	0.227	0.378	0.814
0.65	0.450	0.186	0.332	0.786	0.228	0.379	0.819
0.70	0.449	0.185	0.333	0.785	0.229	0.380	0.819
0.75	0.447	0.183	0.332	0.785	0.231	0.381	0.819
0.80	0.440	0.179	0.329	0.779	0.233	0.382	0.821
0.85	0.426	0.172	0.323	0.765	0.236	0.384	0.824
0.90	0.400	0.165	0.314	0.740	0.238	0.387	0.830
0.95	0.357	0.160	0.299	0.681	0.239	0.389	0.834
1.00	0.343	0.159	0.297	0.655	0.239	0.389	0.836

Table 7
Quantitative results of the effect of CmA on defect categories in terms of mAP@50.

Θ	mAP@50					
	Cracks	Sandholes	Bumps	Oil fouling	Flaking paints	Average
0.00	0.797	0.819	0.356	0.102	0.007	0.416
0.05	0.827	0.859	0.356	0.102	0.007	0.430
0.10	0.856	0.869	0.356	0.102	0.007	0.438
0.15	0.867	0.878	0.356	0.107	0.007	0.443
0.20	0.885	0.878	0.356	0.107	0.007	0.447
0.25	0.887	0.878	0.356	0.107	0.007	0.447
0.30	0.896	0.878	0.356	0.112	0.007	0.450
0.35	0.896	0.878	0.356	0.115	0.007	0.450
0.40	0.896	0.878	0.356	0.115	0.007	0.450
0.45	0.905	0.878	0.356	0.114	0.007	**0.452**
0.50	0.905	0.878	0.356	0.113	0.007	**0.452**
0.55	0.905	0.877	0.355	0.112	0.007	0.451
0.60	0.904	0.877	0.355	0.112	0.007	0.451
0.65	0.904	0.875	0.353	0.112	0.007	0.450
0.70	0.903	0.872	0.353	0.111	0.007	0.449
0.75	0.900	0.864	0.352	0.111	0.007	0.447
0.80	0.893	0.847	0.344	0.111	0.007	0.440
0.85	0.870	0.812	0.330	0.111	0.007	0.426
0.90	0.814	0.748	0.319	0.111	0.007	0.400
0.95	0.700	0.663	0.304	0.111	0.007	0.357
1.00	0.655	0.638	0.303	0.111	0.007	0.343

textual prompts to the original GDINO by using different numbers and semantics of text prompts. As shown in Fig. 13, we used five distinct prompt sets, including three single prompts ({"*Cracks*"}, {"*Little dots*"}, and {"*Defects*"}), one merged text prompt ({"*Cracks*, *Little dots*, *Defects*"}), and an itemized cycling prompt approach ({"*Cracks*, *Little dots*, *Defects*"}), which correspond to the five rows in Fig. 13. The experimental results indicate that 1) overall, the itemized cycling prompt approach identified the greatest number of defects, as depicted in the last row of Fig. 13; 2) the single prompts exhibited significant missed detection issues and demonstrated complementary detection capabilities. For instance, a *crack* defect in the lower part of the first column in the first row of Fig. 13 was detected when prompted with "*Cracks*" but was missed when the other single prompts ("*Little dots*" and "*Defects*") were used, as observed in the second and third rows of the first column in Fig. 13; 3) When the merged text prompt was used, some defects detectable by single prompts were missed. For example, a "crack" in the lower part of the first column in the fourth row of Fig. 13 was detected using the "cracks" single prompt but was not detected using the merged prompts. Consequently, we adopted the itemized cycling prompt approach to construct a large model. The dataset used in this section of the experiment is HIT-EngDD2.

5. Discussion

A comparison of the experimental results (Table 3) shows that 1) when comparing FasterRCNN using ResNet and FasterRCNN using MobileNetV3, when the quantity of data is large, a certain model scale is still needed as a counterpart to enhance the fitting ability of the model to learn more sufficient features to obtain better detection results. 2) The single-stage detection methods represented by the YOLO series outperform the two-stage methods, such as FasterRCNN. We also noticed that, as shown in Fig. 14, the two-stage FasterRCNN yields a certain number of false positives (FPs) due to its two-step defect detection paradigm, which makes the detection accuracy (mAP comprehensively characterizes the performance of recall and precision) inferior to that of the YOLO series. Concerning the advantage of the YOLO series, we further blocked the data augmentations used in YOLO, such as mosaics, and found that the detection performance substantially decreased, thus further proving that reasonable data augmentation methods facilitate defect detection. 3) When the number of data specimens is sufficient to support model training, the transformer-based defect detection methods (except for DETR) outperform those based on the pure CNN paradigm by modeling long dependencies between pixels and features. We argue that the reason for the relatively inferior performance of DETR, which is also based on the transformer architecture, lies in the sparse initialization of its attention modules. This sparse initialization necessitates more iterations for the model to converge to a relatively optimal performance. Additionally, as observed in Fig. 7(d), our dataset exhibits a prominent issue concerning detecting small and even tiny objects. However, during the design of DETR, insufficient consideration was given to addressing the challenges posed by small object detection, thereby resulting in the model's slightly inferior performance. 4) The proposed RHG-Detector, which inherits the advantages of multiple method modalities, achieves the optimal defect detection performance (mAP@50 = 45.19%); the model can detect new defect categories that each method modality does not possess, thus making the model more generalizable to the current engine cylinder bore defect detection task. The qualitative experimental results in Fig. 8 further show that the proposed RHG-Detector has a more robust and comprehensive detection capability than FasterRCNN and YOLO.

A study on the proposed defect discriminators (DeDi, Table 4) shows that the proposed discriminators of the DeDi series are more advantageous in discriminating the defects of cylinder bores: DeDi-S achieves the optimal discrimination performance (acc = 92.0%), followed by DeDi-B (acc = 91.8%) and DeDi-L (acc = 91.7%), thereby proving that filling more spatial information is a correct choice for the current defect discrimination task in low-resolution images. Therefore, we improve upon both absolute location spatial information (LA-MLP) and image spatial information (PSA-PM). The dynamic data recorded from the experiments indicate (Fig. 9) that all the methods in the proposed DeDi series possess better convergence speed and discrimination accuracy both in terms of training/testing loss and training/testing accuracy, thus further verifying the rationality of the structure we designed. Due to the fast convergence property of DeDi, as shown in Fig. 9(e), the model capacity and discrimination accuracy always maintain a positively correlated relationship. Additionally, the qualitative experimental results in Fig. 9 also reveal that the discriminator based on the transformer (Swin) overall outperforms the discriminators based on CNN (MobileNetV3/ResNet); these results ultimately drove us to introduce spatial information into the transformer-based method.

The experimental results in Table 6 and Fig. 11 clearly indicate that setting the similarity threshold in the CmA unit greatly affects the current defect detection task. The reason stems from the fact that the

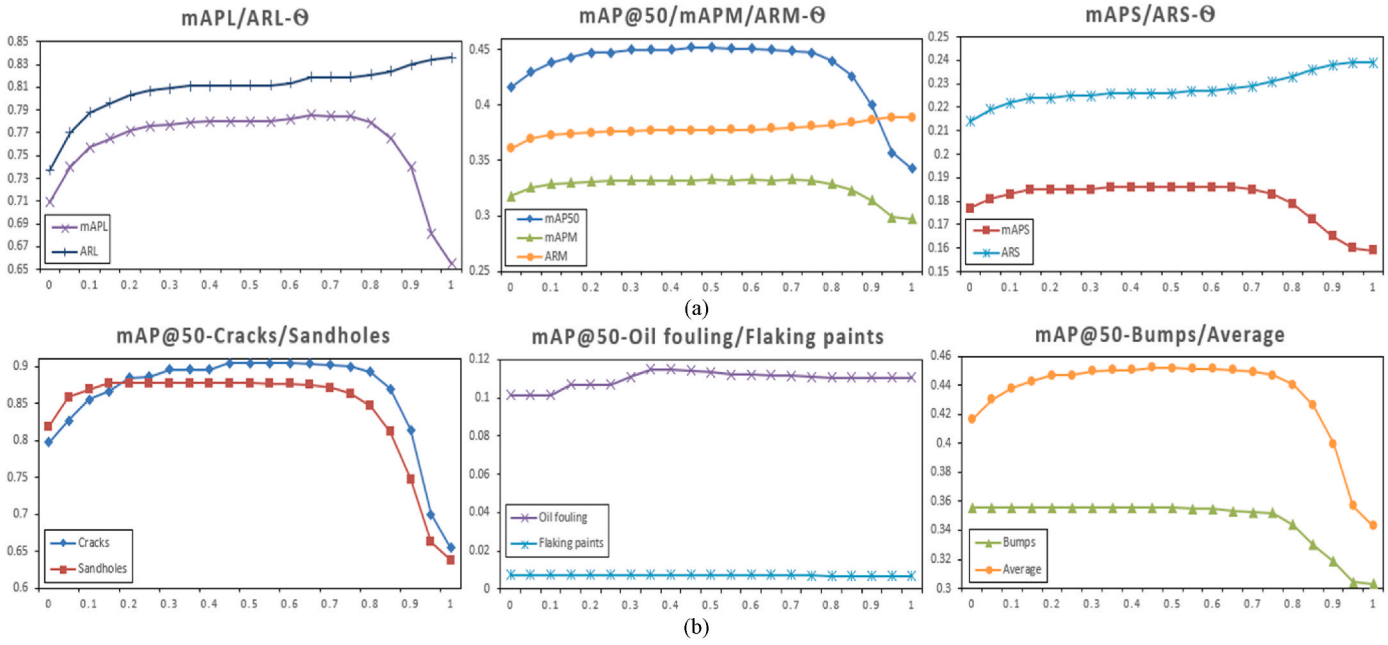

Fig. 11. Results of the study of CmA. The variations in mAPL, M, S; ARL, M, S; and mAP@50 with respect to Θ are presented as line graphs in the first row, and similarly, the variations in detection accuracy with respect to Θ for all defect categories are presented as line graphs in the second row. Θ was varied uniformly from 0 in steps of 0.5–1 in this experiment.

Table 8
Quantitative performance results of methods in detecting defect categories.

Models	Defect					
	mAP@50					
	Cracks	Sandholes	Bumps	Oil fouling	Flaking paints	All
YOLO7S	87.11	84.70	33.23	0	0	41.01
YOLO7X	88.75	87.05	35.54	0	0	42.27
YOLO5S	88.52	86.93	35.43	0	0	42.18
YOLO5L	87.77	**89.37**	35.47	0	0	42.52
SSD	77.59	57.85	32.32	0	0	33.55
RetinaNet	82.83	69.87	34.51	0	0	37.44
FRCNN_FPN (Res50)	83.31	78.40	33.31	0	0	39.00
RHG-Detector	**90.45**	87.77	**35.55**	**11.42**	**0.74**	**45.19**

comprehensive performance indicated by mAP@50 simultaneously depends on both recall and precision, as shown in Eq. (18) to Eq. (21). A greater similarity threshold will leave more redundant detections to be judged as more FPs, and a lower similarity threshold will result in the detected defects being filtered out, thereby increasing the number of FNs and degrading the comprehensive detection accuracy. Therefore, a moderate similarity threshold will modulate FPs and FNs into an optimal state to obtain better defect detection accuracy. The experimental results (all the metrics in Fig. 11 follow a bell-shaped curve, with maximum values in the middle and minimum values on both sides) also show that a similarity threshold near $\Theta = 0.5$ will achieve a better defect detection performance; conclusion is consistent with the theoretical expectation.

The experimental results in Fig. 12 and Table 8 further validate the ability of the methods to detect various defect categories inside cylinder bores in automobile engines. The table and histograms clearly show that, compared to other detection methods, the newly proposed RHG-Detector can detect defects in old categories while also detecting defects in new categories. Additionally, RHG-Detector performs better in

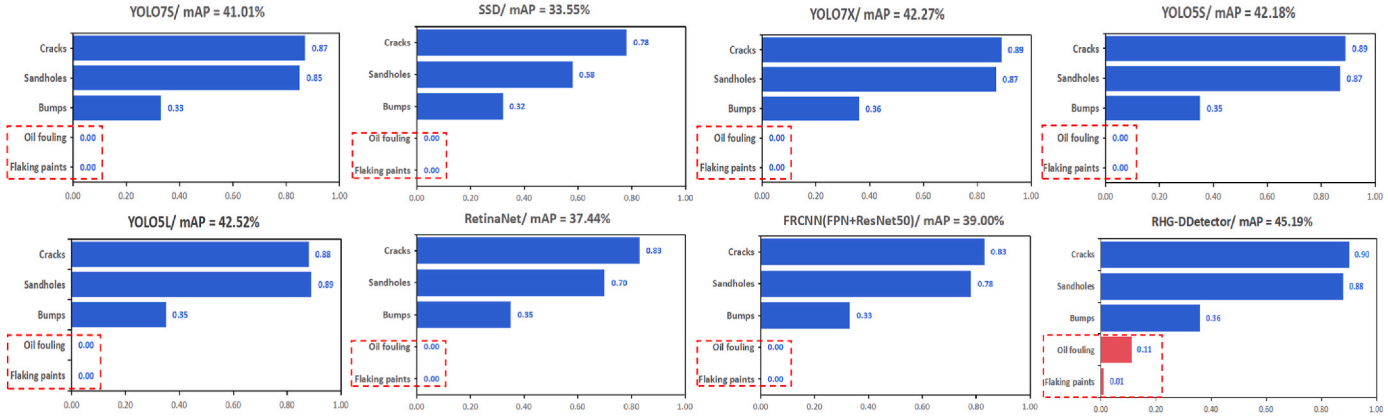

Fig. 12. Experimental results of defect categories. In addition to our RHG-Detector, seven other methods are used in defect category experiments: six one-stage defect detectors (YOLO (7S/7X, 5S/5L), SSD and RetinaNet) and a two-stage detector (FasterRCNN integrated with FPN). All the methods are compared in terms of the metric mAP under IoU = 50 (mAP@50).

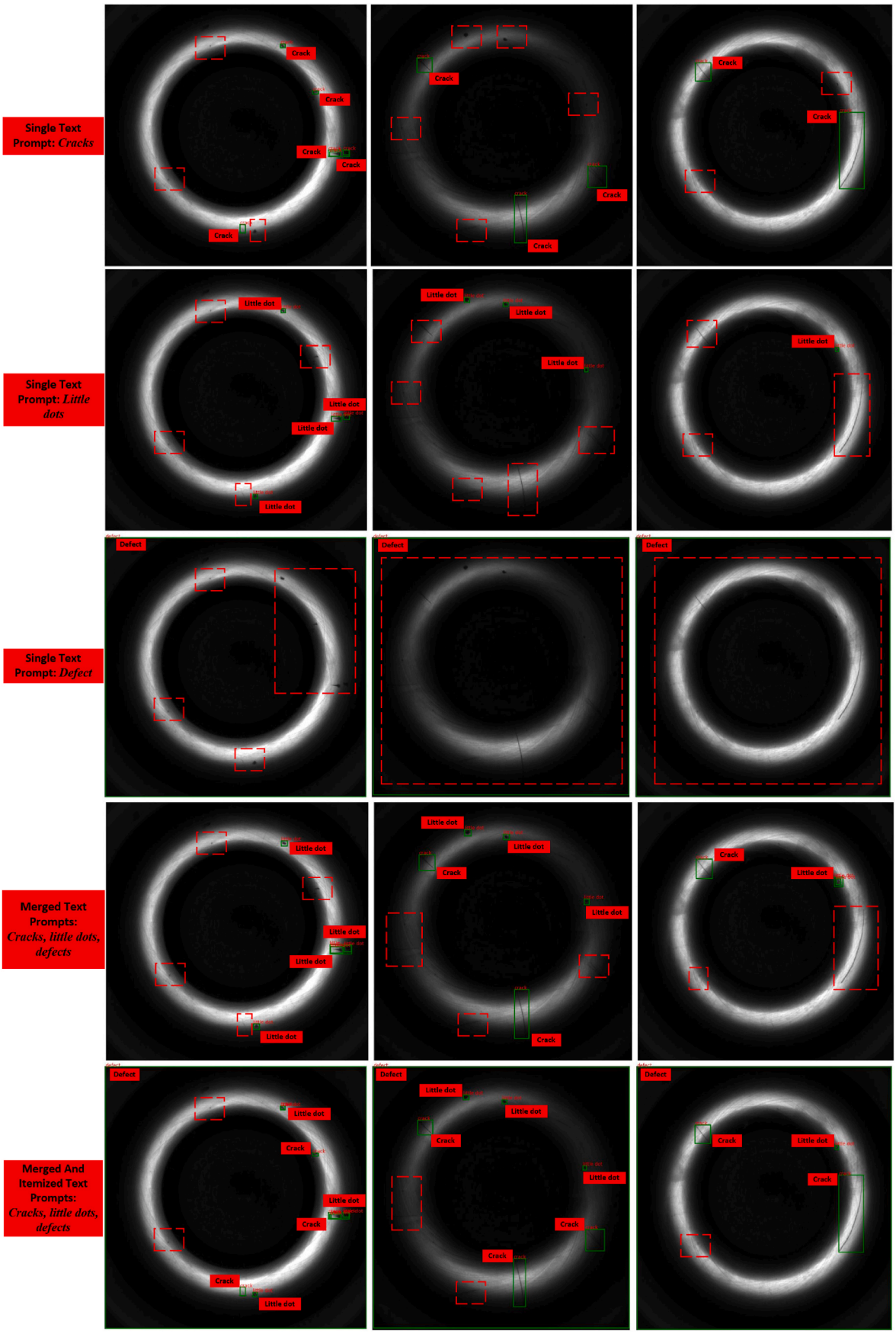
Single Text Prompt: Cracks
Single Text Prompt: Little dots
Single Text Prompt: Defect
Merged Text Prompts: Cracks, little dots, defects
Merged And Itemized Text Prompts: Cracks, little dots, defects
Crack
Little dot
Defect

Fig. 13. Prompting experiments. The above figures present the experimental results of three different images under five different prompting methods. The green solid boxes indicate the defects detected by the large model after prompting with the corresponding textual prompts, while the red dashed boxes denote the missed defects.

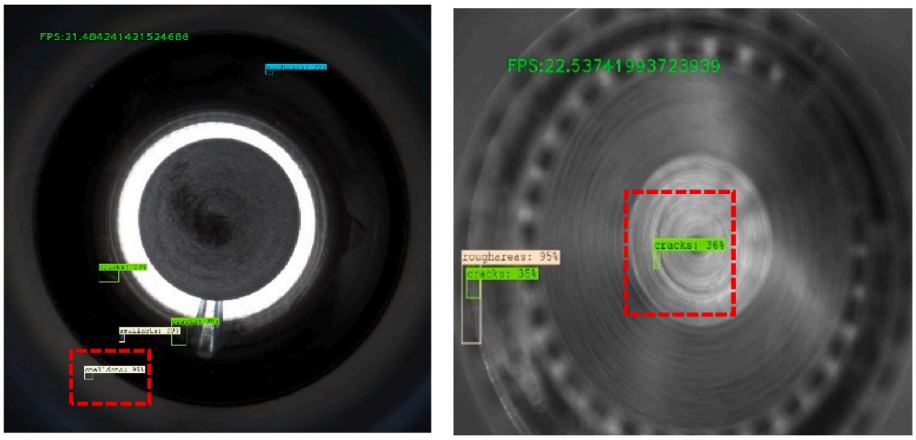

Fig. 14. False positive illustration of FasterRCNN. The FPs yielded by FasterRCNN + FPN(Res50) are rectangles with red dashed boxes in the image.

detecting old defect categories, such as cracks (mAP@50 = 90.45%) and bumps (mAP@50 = 35.55%). YOLO5L performs better in detecting sandholes (mAP@50 = 89.37%); we speculate results from the aforementioned data augmentations used, such as mosaic and mixup, and equivalently oversample tiny and hard-to-detect defects, thereby allowing for better pattern recognition and model learning.

According to Fig. 12 and Table 8, except for RHG-Detector, none of the other algorithms can detect the new categories (oil fouling and flaking paints). Furthermore, we would like to conduct an in-depth discussion on the results of RHG-Detector concerning the new defect categories. In terms of mAP@50, the capability of RHG-Detector to detect oil fouling is 11.42, while in the case of flaking paint, the capability is 0.74. We argue that the main reason for the current results for the flaking paint category lies at the data level. In the upgraded HIT-EngDD2 dataset, the newly emerged defect categories of oil fouling and flaking paints appear less frequently than cracks, sand holes, and bumps in actual production, thus leading to fewer data collected for these categories, especially in the case of flaking paints. In the HIT-EngDC training set, as listed in Table 2, there were only 66 flaking paint samples, accounting for approximately $66/2348 \times 100\% \approx 2.81\%$, which is relatively small compared to cracks ($897/2348 \times 100\% \approx 38.20\%$) and sand holes ($997/2348 \times 100\% \approx 42.46\%$). This observation would force discriminators to tend to favor categories with more data, such as cracks and sand holes. To address this issue, we used the focal loss to handle the class imbalance when training various detectors in this study. In addition, we believe that data augmentation techniques, such as oversampling, can be employed in practical applications to address data imbalance issues. Furthermore, generative large models (diffusion models) can also be used to synthesize defect categories that closely resemble actual defects, further enhancing the dataset.

Although, unlike other methods, RHG-Detector can detect unseen categories, the model efficiency problem needs to be further addressed (model training was conducted on the NVIDIA A100 (40G) platform, while integration into the production line uses the NVIDIA Jetson AGX Orin (64 GB) Developer Kit; the inference time on the A100 is ~4.1 FPS, whereas on the NVIDIA Jetson AGX Orin, it is ~3.2 FPS); this problem is also common among current foundation models. We argue that the following two approaches can be used to address this issue effectively: 1) Model compression. Techniques such as knowledge distillation can be used to improve the performance of foundation models to equal that of small models (preferred). 2) Structure optimization. It is noted in ShuffleNetV2 that a fragmented structure reduces the running efficiency of models, and transforming the bypath structure into an equivalent plain structure by relying on parameterization techniques can be another option. Additionally, as our method is specifically designed for engine cylinder bore defect detection, adapting it to other applications requires training methods consistent with those used for the engine cylinder bore model. A critical consideration will be selecting text prompts for RHG-Detector, which are determined through experiments to identify the most suitable text that accurately conveys intent and goals.

6. Conclusion

In this paper, we propose RHG-Detector, a robust defect detector specifically designed for automobile engine cylinder bores in industrial scenarios. We successfully integrate the generalization enhancer and the cross-modality aggregator into a unified pipeline. This integration transitions the defect detector from a closed set to an open set, thereby maximizing aggregation while preserving the detection advantages of each algorithmic modality for older categories of various cylinder bore defects. Experiments on our established HIT-EngDC verify the SOTA performance of our defect discriminator, DeDi, which achieves an optimal accuracy of 92% in identifying defect categories. Additionally, experiments on our challenging HIT-EngDD2 demonstrate the SOTA performance of our method compared to that of other SOTA methods: for example, in terms of mAP@50, the SOTA performance of our method was ~6% and ~3% higher than that of two-stage FasterRCNN integrated with FPN and that of the one-stage YOLOv7 series, respectively; thus, these experimental results offer a new perspective for addressing the defect detection of the cylinder bores of automobile engines.

CRediT authorship contribution statement

Xujie He: Writing – original draft, Methodology, Conceptualization. **Jing Jin:** Writing – review & editing, Supervision, Funding acquisition. **Duo Chen:** Data curation. **Cangtian Zhou:** Software, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence

the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by the research and development project of a complex workpiece visual defect detection imaging system and software tools of China [No. 2023ZXJ01A01] and Heilongjiang Province's "Million and Ten Million" Major Project in Science and Technology, China [No. 2021ZX10A01].

References

Aggarwal, M., Khullar, V., Goyal, N., Singh, A., Tolba, A., Thompson, E.B., Kumar, S., 2023. Pre-trained deep neural network-based features selection supported machine learning for rice leaf disease classification. Agric. For. 13 https://doi.org/10.3390/agriculture13050936.

Bochkovskiy, A., Wang, C.-Y., Liao, H.-Y.M., 2020. YOLOv4: optimal speed and accuracy of object detection. http://arxiv.org/abs/2004.10934.

Bommasani, R., Hudson, D.A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J.Q., Demszky, D., et al., 2021. On the Opportunities and Risks of Foundation Models, 1–214. http://arxiv.org/abs/2108.07258.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S., 2020. End-to-End object detection with transformers. Lect. Notes Comput. Sci. 213–229. https://doi.org/10.1007/978-3-030-58452-8_13.

Chen, K., Liu, C., Chen, H., Zhang, H., Li, W., Zou, Z., Shi, Z., 2023. RSPrompter: Learning to Prompt for Remote Sensing Instance Segmentation Based on Visual Foundation Model, 1, pp. 1–18. http://arxiv.org/abs/2306.16269.

Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. Lect. Notes Comput. Sci. 833–851. https://doi.org/10.1007/978-3-030-01234-2_49.

Deitsch, S., Christlein, V., Berger, S., Buerhop-Lutz, C., Maier, A., Gallwitz, F., Riess, C., 2019. Automatic classification of defective photovoltaic module cells in electroluminescence images. Sol. Energy 185 (July 2018), 455–468. https://doi.org/10.1016/j.solener.2019.02.067.

Dong, X., Taylor, C.J., Cootes, T.F., 2022. Defect classification and detection using a multitask deep one-class CNN. IEEE Trans. Autom. Sci. Eng. 19 (3), 1719–1730. https://doi.org/10.1109/TASE.2021.3109353.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2020. An image is worth 16x16 words: transformers for image recognition at scale. http://arxiv.org/abs/2010.11929.

Gao, Y., Gao, L., Li, X., Yan, X., 2020. A semii-supervised convolutional neural network-based method for steel surface defect recognition. Robot. Comput. Integrated Manuf. 61 (May 2019) https://doi.org/10.1016/j.rcim.2019.101825.

Gu, J., Han, Z., Chen, S., Beirami, A., He, B., Zhang, G., Liao, R., Qin, Y., Tresp, V., Torr, P., 2023. A Systematic Survey of Prompt Engineering on Vision-Language Foundation Models, pp. 1–21. http://arxiv.org/abs/2307.12980.

Guo, Y., Zhong, L., Qiu, Y., Wang, H., Gao, F., Wen, Z., Zhan, C., 2022. Using ISU-GAN for unsupervised small sample defect detection. Sci. Rep. 12 (1), 1–13. https://doi.org/10.1038/s41598-022-15855-7.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016-Decem, pp. 770–778. https://doi.org/10.1109/CVPR.2016.90.

Hou, Wan, Sui, Wen, Peng, Li, F, S., 2021. Surface defect detection of fabric based on improved faster R-CNN. In: 2021 IEEE 9th International Conference on Information, Communication and Networks. ICICN 2021, pp. 527–531. https://doi.org/10.1109/ICICN52636.2021.9673969.

Howard, A., Sandler, M., Chen, B., Wang, W., Chen, L.C., Tan, M., Chu, G., Vasudevan, V., Zhu, Y., Pang, R., Le, Q., Adam, H., 2019. Searching for mobileNetV3. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1314–1324. https://doi.org/10.1109/ICCV.2019.00140.

Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q., 2017. Densely connected convolutional networks. In: Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017-Janua, pp. 2261–2269. https://doi.org/10.1109/CVPR.2017.243.

Huang, Z., Wu, J., Xie, F., 2021. Automatic recognition of surface defects for hot-rolled steel strip based on deep attention residual convolutional neural network. Mater. Lett. 293, 129707 https://doi.org/10.1016/j.matlet.2021.129707.

Kaji, F., Nguyen-Huu, H., Budhwani, A., Narayanan, J.A., Zimny, M., Toyserkani, E., 2022. A deep-learning-based in-situ surface anomaly detection methodology for laser directed energy deposition via powder feeding. J. Manuf. Process. 81 (July), 624–637. https://doi.org/10.1016/j.jmapro.2022.06.046.

Kim, Y., Lee, J.S., Lee, J.H., 2023. Automatic defect classification using semi-supervised learning with defect localization. IEEE Trans. Semicond. Manuf. 36 (3), 476–485. https://doi.org/10.1109/TSM.2023.3278036.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.-Y., Dollár, P., Girshick, R., 2023. Segment Anything. http://arxiv.org/abs/2304.02643.

Li, W., Zhang, H., Wang, G., Xiong, G., Zhao, M., Li, G., Li, R., 2023. Deep learning based online metallic surface defect detection method for wire and arc additive manufacturing. Robot. Comput. Integrated Manuf. 80 (October 2022), 102470 https://doi.org/10.1016/j.rcim.2022.102470.

Liang, Q., Zhu, W., Sun, W., Yu, Z., Wang, Y., Zhang, D., 2019. In-line inspection solution for codes on complex backgrounds for the plastic container industry. Measurement: J. Int. Measur. Confed. 148, 106965 https://doi.org/10.1016/j.measurement.2019.106965.

Liu, M., Chen, Y., Xie, J., He, L., Zhang, Y., 2023. LF-YOLO: a lighter and faster YOLO for weld defect detection of X-ray image. IEEE Sensor. J. 23 (7), 7430–7439. https://doi.org/10.1109/JSEN.2023.3247006.

Liu, R., Huang, M., Gao, Z., Cao, Z., Cao, P., 2023. MSC-DNet: an efficient detector with multi-scale context for defect detection on strip steel surface. Measurement: J. Int. Measur. Confed. 209 (October 2022), 112467 https://doi.org/10.1016/j.measurement.2023.112467.

Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., Zhang, L., 2023. Grounding DINO: Marrying DINO with Grounded Pre-training for Open-Set Object Detection. http://arxiv.org/abs/2303.05499.

Liu, T., He, Z., Lin, Z., Cao, G.Z., Su, W., Xie, S., 2022. An adaptive image segmentation network for surface defect detection. IEEE Transact. Neural Networks Learn. Syst. 1–14. https://doi.org/10.1109/TNNLS.2022.3230426.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C., 2016. SSD: single shot MultiBox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (Eds.), Computer Vision – ECCV 2016. Springer International Publishing, pp. 21–37.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021. Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 9992–10002. https://doi.org/10.1109/ICCV48922.2021.00986.

Ma, J., He, Y., Li, F., Han, L., You, C., Wang, B., 2023. Segment Anything in Medical Images, 1–9. http://arxiv.org/abs/2304.12306.

Ma, N., Zhang, X., Zheng, H.T., Sun, J., 2018. Shufflenet V2: practical guidelines for efficient cnn architecture design. Lect. Notes Comput. Sci. 122–138. https://doi.org/10.1007/978-3-030-01264-9_8.

MA, Z., Li, Y., Huang, M., Huang, Q., Cheng, J., Tang, S., 2022. A lightweight detector based on attention mechanism for aluminum strip surface defect detection. Comput. Ind. 136, 103585 https://doi.org/10.1016/j.compind.2021.103585.

Ronneberger, Olaf, Fischer, Philipp, B, T., 2015. U-net: convolutional networks for biomedical image segmentation. Lect. Notes Comput. Sci. 9351 (Cvd), 12–20. https://doi.org/10.1007/978-3-319-24574-4.

Ren, S., He, K., Girshick, R., Sun, J., 2017. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. 39 (6), 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031.

Shang, H., Sun, C., Liu, J., Chen, X., Yan, R., 2023. Defect-aware transformer network for intelligent visual surface defect detection. Adv. Eng. Inf. 55 (November 2022), 101882 https://doi.org/10.1016/j.aei.2023.101882.

Shao, L., Zhang, E., Ma, Q., Li, M., 2022. Pixel-wise semisupervised fabric defect detection method combined with multitask mean teacher. IEEE Trans. Instrum. Meas. 71, 1–11. https://doi.org/10.1109/TIM.2022.3162286.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, pp. 1–14.

Song, K., Yan, Y., 2013. A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects. Appl. Surf. Sci. 285 (PARTB), 858–864. https://doi.org/10.1016/j.apsusc.2013.09.002.

Sun, X., Gu, J., Sun, H., 2021. Research Progress of Zero-Shot Learning. November 2020, pp. 3600–3614.

Tabernik, D., Šela, S., Skvarč, J., Skočaj, D., 2020. Segmentation-based deep-learning approach for surface-defect detection. J. Intell. Manuf. 31 (3), 759–776. https://doi.org/10.1007/s10845-019-01476-x.

Taherkhani, K., Eischer, C., Toyserkani, E., 2022. An unsupervised machine learning algorithm for in-situ defect-detection in laser powder-bed fusion. J. Manuf. Process. 81 (March), 476–489. https://doi.org/10.1016/j.jmapro.2022.06.074.

Tang, Y., Han, K., Guo, J., Xu, C., Xu, C., Wang, Y., 2022. GhostNetV2: enhance cheap operation with long-range attention. Adv. Neural Inf. Process. Syst. 35, 1–12. NeurIPS.

Tao, X., Zhang, D., Ma, W., Hou, Z., Lu, Z.F., Adak, C., 2022. Unsupervised anomaly detection for surface defects with dual-siamese network. IEEE Trans. Ind. Inf. 18 (11), 7707–7717. https://doi.org/10.1109/TII.2022.3142326.

Wang, C.-Y., Bochkovskiy, A., Liao, H.-Y.M., 2022. YOLOv7: Trainable Bag-Of-Freebies Sets New State-Of-The-Art for Real-Time Object Detectors, pp. 1–15. http://arxiv.org/abs/2207.02696.

Wang, F., Ding, L., Rao, J., Liu, Y., Shen, L., Ding, C., 2023. Can linguistic knowledge improve multimodal alignment in vision-language pretraining? arXiv e-prints arXiv: 2308.12898. https://doi.org/10.48550/arXiv.2308.12898.

Wang, R., Cheung, C.F., 2022. CenterNet-based defect detection for additive manufacturing. Expert Syst. Appl. 188 (June 2021), 116000 https://doi.org/10.1016/j.eswa.2021.116000.

Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L., 2022. PVT v2: improved baselines with pyramid vision transformer. Computational Visual Media 8 (3), 415–424. https://doi.org/10.1007/s41095-022-0274-8.

Wang, X., Wang, W., Cao, Y., Shen, C., Huang, T., 2023. Images Speak in Images: A Generalist Painter for In-Context Visual Learning, pp. 6830–6839. https://doi.org/10.1109/cvpr52729.2023.00660.

Wang, X., Zhang, X., Cao, Y., Wang, W., Shen, C., Huang, T., 2023. SegGPT: segmenting everything in context. http://arxiv.org/abs/2304.03284.

Wang, Z., Xie, W., Chen, H., Liu, B., Shuai, L., 2022. Color point defect detection method based on color salient features. Electronics (Switzerland) 11 (17), 1–13. https://doi.org/10.3390/electronics11172665.

Xu, R., Hao, R., Huang, B., 2022. Efficient surface defect detection using self-supervised learning strategy and segmentation network. Adv. Eng. Inf. 52 (January) https://doi.org/10.1016/j.aei.2022.101566.

Yang, J., Gao, M., Li, Z., Gao, S., Wang, F., Zheng, F., 2023. Track Anything: Segment Anything Meets Videos. http://arxiv.org/abs/2304.11968.

Yang, M., Wu, P., Feng, H., 2023. MemSeg: a semi-supervised method for image surface defect detection using differences and commonalities. Eng. Appl. Artif. Intell. 119 (December 2022), 105835 https://doi.org/10.1016/j.engappai.2023.105835.

Yu, X., Han-Xiong, L., Yang, H., 2023. Collaborative learning classification model for PCBs defect detection against image and label uncertainty. IEEE Trans. Instrum. Meas. 72, 1–8. https://doi.org/10.1109/TIM.2023.3235461.

Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.-Y., 2022. DINO: DETR with improved DeNoising anchor boxes for end-to-end object detection. http://arxiv.org/abs/2203.03605.

Zhang, N.X., Zhong, Y., Dian, S., 2023. Rethinking unsupervised texture defect detection using PCA. Opt Laser. Eng. 163 (October 2022), 107470 https://doi.org/10.1016/j.optlaseng.2022.107470.

Zhang, Y., Zeng, W., Zheng, J., Liu, J., Zhao, Y., Fan, T., 2023. Tile surface defect detection based on improved YOLOv5. In: 2023 IEEE International Conference on Control, Electronics and Computer Technology, 7. ICCECT 2023, pp. 1138–1141. https://doi.org/10.1109/ICCECT57938.2023.10141339, 1.

Zhang, Z., Wen, G., Chen, S., 2019. Weld image deep learning-based on-line defects detection using convolutional neural networks for Al alloy in robotic arc welding. J. Manuf. Process. 45 (July), 208–216. https://doi.org/10.1016/j.jmapro.2019.06.023.

Zhao, C., Shu, X., Yan, X., Zuo, X., Zhu, F., 2023. RDD-YOLO: a modified YOLO for detection of steel surface defects. Measurement: J. Int. Measur. Confed. 214 (October 2022), 112776 https://doi.org/10.1016/j.measurement.2023.112776.

Zhao, X., Ding, W., An, Y., Du, Y., Yu, T., Li, M., Tang, M., Wang, J., 2023. Fast Segment Anything, 2. http://arxiv.org/abs/2306.12156.

Zhong, Y., Yang, J., Zhang, P., Li, C., Codella, N., Li, L.H., Zhou, L., Dai, X., Yuan, L., Li, Y., Gao, J., 2022. RegionCLIP: region-based language-image pretraining. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2022-June, pp. 16772–16782. https://doi.org/10.1109/CVPR52688.2022.01629.

Zhou, K., Yang, J., Loy, C.C., Liu, Z., 2022. Learning to prompt for vision-language models. Int. J. Comput. Vis. 130, 2337–2348. https://doi.org/10.1007/s11263-022-01653-1.

Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J., 2021. Deformable detr: deformable transformers for end-to-end object detection. ICLR 2021 - 9th Int. Conf. Learn. Represent. 1–16.

Zorić, B., Matić, T., Hocenski, Ž., 2022. Classification of biscuit tiles for defect detection using Fourier transform features. ISA (Instrum. Soc. Am.) Trans. 125, 400–414. https://doi.org/10.1016/j.isatra.2021.06.025.

Zou, Y., Jeong, J., Pemula, L., Zhang, D., Dabeer, O., 2022. SPot-the-difference self-supervised pre-training for anomaly detection and segmentation. In: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 13690 LNCS. Springer, Nature Switzerland. https://doi.org/10.1007/978-3-031-20056-4_23.