Ordered Cross-Scale Interaction Network for No-Service Rail Surface Defect Segmentation

Gongyang Li, *Member, IEEE*, Xiaofei Zhou, and Hongyun Li

***Abstract*—No-service rail surface defect (NRSD) segmentation plays a key role in industrial intelligent manufacturing to achieve pixel-level defect localization and ensure the quality of rails. However, the unique properties of no-service rails, such as blurred boundaries and limited semantic information, make surface defect segmentation extremely challenging. In this paper, we introduce the Pyramid Vision Transformer (PVT) to NRSD segmentation, and propose a novel transformer-based Ordered Cross-scale Interaction Network (*OCINet*). The core of OCINet is the global-local-global strategy. Obviously, a transformer backbone is arranged to extract global features. Subsequently, three cross-scale interaction modules, including Cross-scale Channel Interaction Module (CCIM), Cross-scale Spatial Interaction Module (CSIM), and Cross-scale Pixel Interaction Module (CPIM), are employed to achieve ordered channel, spatial, and pixel interactions of defect features across adjacent scales. Among them, CCIM and CSIM are used for local interaction, while CPIM is used for global interaction. These modules not only locate the defect regions, but also model the pixel by pixel relationships between defect regions and backgrounds, generating discriminative features for defect segmentation. Comprehensive experiments and analysis on the NRSD-MN dataset, which contains 4,101 NRSD images, indicate that our OCINet outperforms 19 state-of-the-art methods, achieving 68.7% in mean IoU and 78.3% in mean Dice. The code and results of our method are available at https://github.com/MathLee/OCINet.**

***Index Terms*—Surface defect segmentation, no-service rail, transformer, cross-scale interaction.**

I. Introduction

SURFACE defect segmentation is an indispensable part of industrial manufacturing. It precisely locates defect regions at the pixel level, making it more accurate and challenging than the detection task [1], [2]. To every country in the world, railway transportation is important. For railway transportation, the inspection of on-track rails is very important for railway safety [3], [4]. Similarly, the manufacturing process and quality inspection of no-service rails are also crucial [5]. The on-track rails refer to the steel rails that have already been laid on the railway, while the no-service rails refer to steel rails that are still in the production factory [5]. In this paper, we focus on the latter one, *i.e.*, No-Service Rail Surface Defect (NRSD) segmentation [5]–[7], to improve the yield rate and ensure the quality of rails in the production workshop. With technological advancements, industrial manufacturing is becoming increasingly intelligent [8], and the requirements for segmentation accuracy are becoming more stringent. Therefore, we are committed to developing an intelligent and high-performance NRSD segmentation method.

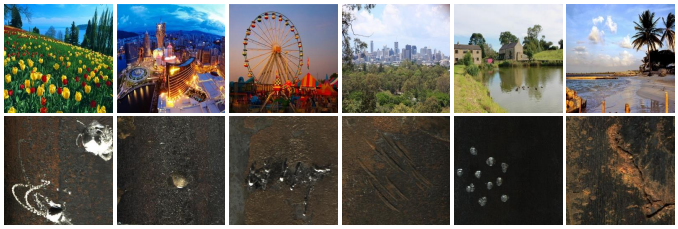

Fig. 1. Typical example scenes of natural images and NRSD images.

Due to the unique characteristics of the acquisition scene, NRSD images differ significantly from natural images [5], [10]. As shown in Fig. 1, natural images have rich colors and clear textures and details, with well-defined semantics for the objects. NRSD images contain minimal semantic information, with monotonous colors, low contrast, low illumination, and cluttered scenes. The defect regions exhibit varying manifestations, random locations, irregular shapes, various sizes, and blurred boundaries. These unique properties of NRSD images and defect regions put classical natural image segmentation methods [11]–[13] in a dilemma (as shown in Tab. I), and also hinder the development of NRSD segmentation.

In the era of deep learning, data shortage poses obstacles to NRSD segmentation. With the introduction of the NRSD-MN dataset [5], NRSD segmentation has ushered in a turning point, with a multitude of methods emerging. The attention mechanism [5], weakly supervised learning [6], and attention normalization [7] have been introduced into specialized NRSD segmentation methods, significantly improving segmentation accuracy. However, these specialized methods have limitations. Firstly, they are all based on Convolutional Neural Network (CNN) backbones, such as ResNet [14] and DenseNet [15], which is not conducive to modeling the complex relationship between defects and backgrounds. Secondly, their feature enhancement manners are relatively simple, focusing only on local or global enhancement, and cannot effectively highlight the defect regions. The limited feature extraction capabilities

This work was supported in part by the National Natural Science Foundation of China under Grant 62401350 and Grant 62271180, in part by the Shanghai Sailing Program under Grant 24YF2713000, and in part by the Opening Foundation of Quanzhou Vocational and Technical University in 2024 (Project number LERIS24-02). *(Corresponding author: Hongyun Li.)*

Gongyang Li is with the Laboratory of Environment Recognition and Intelligent Systems, Quanzhou Vocational and Technical University, Quanzhou 362000, China, and the School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China (email: ligongyang@shu.edu.cn).

Xiaofei Zhou is with the School of Automation, Hangzhou Dianzi University, Hangzhou 310018, China (e-mail: zxforchid@outlook.com).

Hongyun Li is with the Industrial School of Joint Innovation, Quanzhou Vocational and Technical University, Quanzhou 362000, China (email: ynlihy@163.com).

and single feature enhancement manners are suboptimal for NRSD images [5]–[7].

Inspired by the above observations, in this paper, we develop a novel global-local-global strategy to improve the capabilities of relationship modeling and feature enhancement. With this strategy, we propose a novel specialized Ordered Cross-scale Interaction Network (*OCINet*) for NRSD segmentation. OCINet is built on the transformer backbone [16], which differs from the CNN backbones used in previous methods and is better suitable for handling the unique properties of NRSD images. Notably, this is the first use of a transformer for NRSD segmentation to extract the global relationship between each basic patch of NRSD images. Furthermore, we enhance features comprehensively, considering both local and global aspects through cross-scale interactions, which helps highlight complex and variable defect regions in features.

In particular, our OCINet is built on the well-known encoder-decoder framework with the Pyramid Vision Transformer (PVT) [16] as the backbone. In addition to the basic feature extractor (*i.e.*, encoder) and defect reasoner (*i.e.*, decoder), our OCINet contains three ordered cross-scale interaction modules: Cross-scale Channel Interaction Module (CCIM), Cross-scale Spatial Interaction Module (CSIM), and Cross-scale Pixel Interaction Module (CPIM). All three modules leverage the complementary information from features at adjacent scales. They sequentially perform channel, spatial, and pixel interactions to respectively locate defects, outline defects, and model the relationships between defects and backgrounds. Through these three carefully designed modules, as shown in Fig. 6, defect regions gradually stand out from the cluttered background and become obvious. By perfectly integrating novel modules with a well-known framework, our OCINet breaks the traditional thinking of previous methods, and achieves promising defect segmentation accuracy.

Our main contributions are summarized as follows:

- We explore NRSD segmentation using the transformer for the first time, and propose the first transformer-based specialized NRSD segmentation method, termed *OCINet*, with the global-local-global strategy.
- We propose three ordered cross-scale interaction modules, namely CCIM, CSIM, and CPIM, to perform tailored channel, spatial, and pixel interactions on defect features. CCIM determines *where the defect is*, CSIM perceives *what the defect is*, and CPIM models *how the relationship between the defect and the background is*, thereby facilitating defect perception and segmentation.
- We evaluate the proposed OCINet on the challenging NRSD-MN dataset which includes both man-made and natural NRSDs. Experiments demonstrate the advantages of our OCINet over three types of comparison methods, highlight the effectiveness of our modules, and confirm the superiority of our strategy.

The rest of this paper is organized as follows. In Sec. II, we introduce the related work of natural image segmentation and NRSD segmentation. In Sec. III, we detail our proposed method. In Sec. IV, we conduct extensive experiments. In Sec. V, we give the conclusion.

II. Related Work

A. Natural Image Segmentation

Natural image segmentation plays an important role in computer vision and is crucial to understanding images. Deep learning technology has promoted the development of image segmentation. A large number of excellent methods have emerged [17]–[19]. As a representative work, Long *et al*. [11] broke through traditional thinking and proposed the subversive Fully Convolutional Network (FCN), which is the first deep learning-driven end-to-end segmentation method. Similar to the skip operation of FCN, Badrinarayanan *et al*. [12] connected the encoder and the decoder through the skip operation in the well-known SegNet, and transferred the encoder features and pooling indices to the decoder. The encoder-decoder architecture of SegNet has brought a huge impact on other related dense prediction tasks [7], [20]–[22].

With the rapid development of natural image segmentation, Chen *et al*. [13] introduced the atrous convolution to the semantic segmentation and made a pyramid structure to capture multi-scale information, boosting the performance. Fu *et al*. [23] modeled the global spatial and channel interdependencies using the position and channel attention modules from local features in DANet. Xie *et al*. [24] proposed a simple yet powerful semantic segmentation framework, termed SegFormer, which combines transformers with lightweight multilayer perceptron decoders. During the same period, Strudel *et al*. [25] constructed a semantic segmentation model based on visual transformers, named Segmenter, for capturing contextual information. SegFormer and Segmenter marked the beginning of segmentation models based on transformers. Kirillov *et al*. [26] proposed the famous foundation model for segmentation, named Segment Anything Model (SAM). SAM is versatile, supporting various types of segmentation prompts, including mask, point, bounding box, or text, and has the ability to generalize to unfamiliar objects and images without additional training.

In addition to the above methods that focus on segmentation performance, researchers also pay attention to model complexity and weak supervision learning. For example, Lu *et al*. [27] introduced a token reduction approach, *i.e.*, content-aware token sharing, to improve the computational efficiency of semantic segmentation networks that use vision transformers. Norouzi *et al*. [28] analyzed the similarities between intra- and inter-class tokens within local windows and across network layers to achieve another effective token reduction method. Shi *et al*. [29] proposed the unified multi-feature fusion module to efficiently fuse multiple features at a low computational cost. Lu *et al*. [30] constructed a three-branch architecture network with detail branch, semantic branch, compensation branch, and an efficient aggregation layer. Shi *et al*. [31] developed a hybrid and efficient transformer-CNN structure to better model the long-range and short-range spatial dependence. To alleviate the problem of difficulty in obtaining pixel-level labels, Wang *et al*. [32] solved semantic segmentation with image-level supervision only, and proposed two co-attentions to obtain cross-image semantic similarities and differences for object localization.

There is a big gap between natural images and NRSD images in terms of scenes, objects, and backgrounds. General natural image segmentation methods may not be able to handle NRSD images well. But we draw inspiration from these classic methods. Our OCINet is built on the encoder-decoder architecture [12], that is, we carefully enhance features extracted from the encoder with three cross-scale interaction modules to adapt to NRSD images, and pass them to the decoder. These three cross-scale interaction modules can capture local and global interdependencies.

B. No-Service Rail Surface Defect Segmentation

NRSD segmentation is an essential part of rail production to ensure the quality of no-service rails. The unique properties of no-service rails have led researchers to propose many specialized segmentation methods. For example, Zhang *et al*. [5] proposed the first deep learning-based NRSD segmentation method, which adopts contextual attention to get the local attention for each pixel. A large-scale dataset of about 4,000 NRSD images is also presented. Li *et al*. [7] modified the traditional channel attention through the normalization operation to adapt to NRSD images, further improving the segmentation accuracy. In addition to the above fully supervised NRSD segmentation methods, similar to [32] in natural image segmentation, Zhang *et al*. [6] made a weakly supervised attempt on NRSD segmentation, and proposed the pooling combination module to generate pseudo pixel-level labels from image-level defect category labels.

With the development of depth sensors, multimodal industrial data can be easily obtained. Wang *et al*. [33] collected the first dataset of RGB images and depth images for NRSD segmentation. They implicitly exploited the complementarity of multimodal images through image concatenation and feature extraction operations for NRSD segmentation. Zhou *et al*. [34] and Huang *et al*. [35] extracted multimodal features using two independent networks, and focused on lightweight and fast segmentation methods.

The strip steel and no-service rail are made of the same material and the production lines are similar. Therefore, we introduce some methods for salient object detection of strip steel surface defects. This task regards defect regions as salient regions. Zhou *et al*. [36] first predicted the defect regions, and then refined them. Han *et al*. [9] followed this two-stage strategy, and additionally introduced edge information in these two stages. Edge information is also taken into account in [37] and [38]. But in [37], Zhou *et al*. focused on exploring the multi-Level interactive information. In [38], Ding *et al*. explored the contextual information for feature calibration and fusion. Shen *et al*. [39] proposed the multiscale interactive module, and achieved an extremely lightweight network with only 0.28M parameters. Differently, Zhou *et al*. [40] extracted multi-scale features from multi-resolution strip steel images instead of from the single images like the above methods.

Since strip steel and no-service rail have different production processes, their defect manifestations are different. The methods of strip steel cannot be transferred to no-service rails well. As for the specialized NRSD segmentation methods [5]–[7], their feature extractors are all CNN backbones (such as ResNet and DenseNet), which are not good at modeling the relationship between different regions of NRSD images. In addition, some typical operations, such as contextual attention in [5] and normalized channel attention [7], have room for improvement in sufficiently enhancing the defect features with weak semantics. To this end, in our OCINet, we implement the global-local-global strategy. We adopt the PVT as the feature extractor to achieve the global defect features. We also propose three cross-scale interaction modules, *i.e*., CCIM, CSIM, and CPIM, to achieve comprehensive feature enhancement at channel, spatial, and pixel aspects. These efforts make our OCINet an excellent NRSD segmenter.

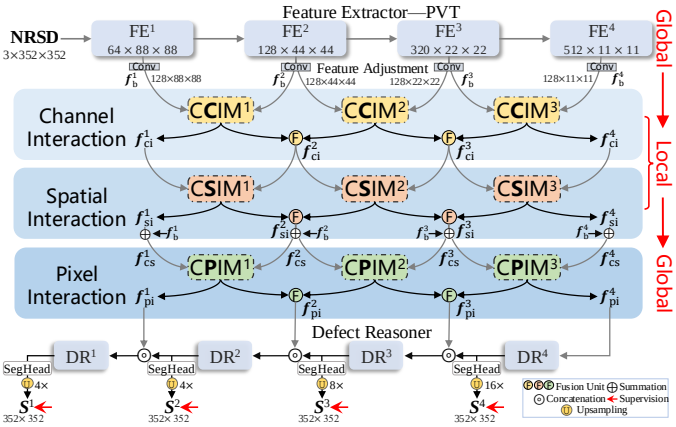

Fig. 2. Architecture of the proposed OCINet for NRSD segmentation, which is based on the global-local-global strategy. OCINet consists of a feature extractor (*i.e*., encoder), three cross-scale interaction modules, *i.e*., Cross-scale Channel Interaction Module (CCIM), Cross-scale Spatial Interaction Module (CSIM) and Cross-scale Pixel Interaction Module (CPIM), and a defect reasoner (*i.e*., decoder). The transformer backbone, *i.e*., PVT-v2-b2 [16], extracts the global features. The three modules sequentially perform channel, spatial, and pixel interactions on defect features. Here, CCIM and CSIM are for local enhancement, while CPIM is for global enhancement. The defect reasoner gradually infers and outlines defect regions. Please zoom in for the best view.

III. Methodology

In this section, we introduce our OCINet. We first give an overview of our OCINet, then elaborate on our three cross-scale interaction modules, and finally present the loss function.

A. Network Overview

As illustrated in Fig. 2, our OCINet follows the global-local-global strategy. It consists of a feature extractor, three cross-scale interaction modules (*i.e*., CCIM, CSIM, and CPIM), and a defect reasoner. Specifically, we adopt the popular PVT-v2-b2 [16] as the feature extractor to model the global long-range dependencies between each patch of the complex NRSD image. The input size of PVT-v2-b2 is 3×352×352. It consists of four transformer blocks, namely FE^i ($i \in \{1, 2, 3, 4\}$). We append a convolution layer after each transformer block to achieve feature adjustment (*i.e*., unifying the feature channel to c), generating four-level basic features $\boldsymbol{f}_{\mathrm{b}}^i \in \mathbb{R}^{c \times h_i \times w_i}$, where c is 128 and $h_i/w_i = \frac{352}{2^{i+1}}$. With these global features, we sequentially perform the channel interaction and the spatial interaction on them in CCIM and CSIM. CCIM and CSIM are

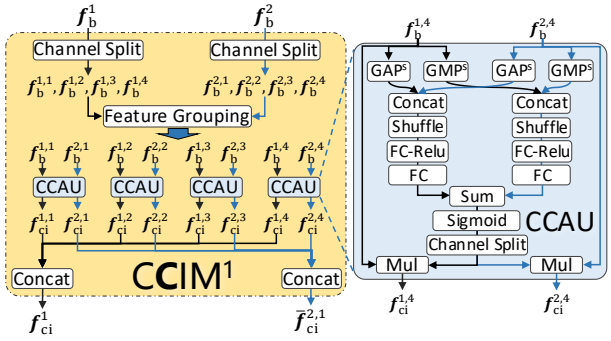

Fig. 3. Illustration of the Cross-scale Channel Interaction Module and its key component of Collaborative Channel Attention Unit. Here, GAP^{s}/GMP^{s} means the spatial-wise global average/max pooling.

extended from channel and spatial attention mechanisms [41], responsible for local enhancement. CCIM determines the defect location in the channel, generating $\boldsymbol{f}_{\mathrm{ci}}^{i}$. CSIM perceives the defect in the spatial domain, generating $\boldsymbol{f}_{\mathrm{si}}^{i}$. Moreover, we perform the pixel interaction on $\boldsymbol{f}_{\mathrm{si}}^{i}$ in CPIM. CPIM is extended from the self-attention mechanism [23], responsible for global enhancement. It models the pixel-level relationships between defects and backgrounds, generating $\boldsymbol{f}_{\mathrm{pi}}^{i}$. This ordered and comprehensive feature interaction allows the defect regions to gradually emerge from $\boldsymbol{f}_{\mathrm{b}}^{i}$, enabling accurate defect segmentation in the defect reasoner with four blocks (*i.e.*, DR^{i}, $i \in \{1,2,3,4\}$) and four segmentation heads (SegHeads), as shown at the bottom of Fig. 2.

B. Cross-scale Channel Interaction Module

It is well-known that the coordination of global and local information plays a crucial role in achieving high segmentation performance. Since $\boldsymbol{f}_{\mathrm{b}}^{i}$ extracted from PVT contains rich global information, we aim to perform local enhancement on it to achieve a balance between global and local information. Since the channel attention mechanism [41] is a commonly used local enhancement operation for natural images, it may not be sufficient for defect feature enhancement. Therefore, we extend the vanilla channel attention to a cross-scale and grouping scheme, proposing the Cross-scale Channel Interaction Module to achieve more suitable local enhancement for defect features.

Taking CCIM^{1} as an example, we illustrate its detailed structure in Fig. 3. The inputs of CCIM^{1} are $\boldsymbol{f}_{\mathrm{b}}^{1}$ and $\boldsymbol{f}_{\mathrm{b}}^{2}$. To explore the cross-scale complementary information, we here adopt the separation-recombination strategy to process $\boldsymbol{f}_{\mathrm{b}}^{1}$ and $\boldsymbol{f}_{\mathrm{b}}^{2}$. Specifically, we perform channel split on $\boldsymbol{f}_{\mathrm{b}}^{1}$ and $\boldsymbol{f}_{\mathrm{b}}^{2}$ respectively, and get two feature groups, *i.e.*, $\{\boldsymbol{f}_{\mathrm{b}}^{1,1}, \boldsymbol{f}_{\mathrm{b}}^{1,2}, \boldsymbol{f}_{\mathrm{b}}^{1,3}, \boldsymbol{f}_{\mathrm{b}}^{1,4}\} \in \mathbb{R}^{\frac{c}{4}\times h_1\times w_1}$ and $\{\boldsymbol{f}_{\mathrm{b}}^{2,1}, \boldsymbol{f}_{\mathrm{b}}^{2,2}, \boldsymbol{f}_{\mathrm{b}}^{2,3}, \boldsymbol{f}_{\mathrm{b}}^{2,4}\} \in \mathbb{R}^{\frac{c}{4}\times h_2\times w_2}$. Subsequently, the four features in the two groups are paired together to obtain four feature subsets, *i.e.*, $\{\boldsymbol{f}_{\mathrm{b}}^{1,1}, \boldsymbol{f}_{\mathrm{b}}^{2,1}\}$, $\{\boldsymbol{f}_{\mathrm{b}}^{1,2}, \boldsymbol{f}_{\mathrm{b}}^{2,2}\}$, $\{\boldsymbol{f}_{\mathrm{b}}^{1,3}, \boldsymbol{f}_{\mathrm{b}}^{2,3}\}$, and $\{\boldsymbol{f}_{\mathrm{b}}^{1,4}, \boldsymbol{f}_{\mathrm{b}}^{2,4}\}$.

These four feature subsets are processed by the Collaborative Channel Attention Unit (CCAU). CCAU is an upgraded version of channel attention. This unit identifies channels crucial for NRSD segmentation by leveraging the correlation between cross-scale features. Its structure is shown on the right side of Fig. 3. Taking $\{\boldsymbol{f}_{\mathrm{b}}^{1,4}, \boldsymbol{f}_{\mathrm{b}}^{2,4}\}$ for example, we perform the spatial-wise global average pooling and global max pooling on $\boldsymbol{f}_{\mathrm{b}}^{1,4}$, getting $\{\boldsymbol{f}_{\mathrm{ba}}^{1,4}, \boldsymbol{f}_{\mathrm{bm}}^{1,4}\} \in \mathbb{R}^{\frac{c}{4}\times 1\times 1}$. With the same operations, we obtain $\{\boldsymbol{f}_{\mathrm{ba}}^{2,4}, \boldsymbol{f}_{\mathrm{bm}}^{2,4}\} \in \mathbb{R}^{\frac{c}{4}\times 1\times 1}$ from $\boldsymbol{f}_{\mathrm{b}}^{2,4}$. Then, we concatenate $\boldsymbol{f}_{\mathrm{ba}}^{1,4}$ with $\boldsymbol{f}_{\mathrm{ba}}^{2,4}$, and $\boldsymbol{f}_{\mathrm{bm}}^{1,4}$ with $\boldsymbol{f}_{\mathrm{bm}}^{2,4}$ along the channel dimension. To further strengthen channel interactions, we perform the channel shuffle on the two groups of concatenated features. Similar to the traditional channel attention, as shown in the CCAU of Fig. 3, we sequentially perform two Fully Connected (FC) layers, summation, and the sigmoid activation function to obtain the channel attention map, *i.e.*, $\boldsymbol{ca}^{4} \in \mathbb{R}^{\frac{c}{2}\times 1\times 1}$. $\boldsymbol{ca}^{4}$ is the result of the interaction of cross-scale features and has a strong ability to identify important channels. It is separated into $\boldsymbol{ca}^{1,4} \in \mathbb{R}^{\frac{c}{4}\times 1\times 1}$ and $\boldsymbol{ca}^{2,4} \in \mathbb{R}^{\frac{c}{4}\times 1\times 1}$ through channel split. $\boldsymbol{ca}^{1,4}$ and $\boldsymbol{ca}^{2,4}$ then are used to enhance the corresponding $\boldsymbol{f}_{\mathrm{b}}^{1,4}$ and $\boldsymbol{f}_{\mathrm{b}}^{2,4}$ by multiplication, producing the output of CCAU, *i.e.*, $\boldsymbol{f}_{\mathrm{ci}}^{1,4} \in \mathbb{R}^{\frac{c}{4}\times h_1\times w_1}$ and $\boldsymbol{f}_{\mathrm{ci}}^{2,4} \in \mathbb{R}^{\frac{c}{4}\times h_2\times w_2}$.

After all four feature subsets are processed in CCAUs, we get four enhanced feature subsets, *i.e.*, $\{\boldsymbol{f}_{\mathrm{ci}}^{1,1}, \boldsymbol{f}_{\mathrm{ci}}^{2,1}\}$, $\{\boldsymbol{f}_{\mathrm{ci}}^{1,2}, \boldsymbol{f}_{\mathrm{ci}}^{2,2}\}$, $\{\boldsymbol{f}_{\mathrm{ci}}^{1,3}, \boldsymbol{f}_{\mathrm{ci}}^{2,3}\}$, and $\{\boldsymbol{f}_{\mathrm{ci}}^{1,4}, \boldsymbol{f}_{\mathrm{ci}}^{2,4}\}$. Finally, we perform channel concatenation on $\boldsymbol{f}_{\mathrm{ci}}^{1,1}$, $\boldsymbol{f}_{\mathrm{ci}}^{1,2}$, $\boldsymbol{f}_{\mathrm{ci}}^{1,3}$, and $\boldsymbol{f}_{\mathrm{ci}}^{1,4}$, getting one output of CCIM^{1}, *i.e.*, $\boldsymbol{f}_{\mathrm{ci}}^{1} \in \mathbb{R}^{c\times h_1\times w_1}$. Similarly, we obtain the another output of CCIM^{1}, *i.e.*, $\bar{\boldsymbol{f}}_{\mathrm{ci}}^{2,1} \in \mathbb{R}^{c\times h_2\times w_2}$. Following CCIM^{1}, CCIM^{2} produces $\bar{\boldsymbol{f}}_{\mathrm{ci}}^{2,2}$ and $\bar{\boldsymbol{f}}_{\mathrm{ci}}^{3,1}$, and CCIM^{3} outputs $\bar{\boldsymbol{f}}_{\mathrm{ci}}^{3,2}$ and $\boldsymbol{f}_{\mathrm{ci}}^{4}$. Notably, as shown Fig. 2, we employ the fusion unit (*i.e.*, the summation and a convolution layer) to integrate $\bar{\boldsymbol{f}}_{\mathrm{ci}}^{2,1}$ and $\bar{\boldsymbol{f}}_{\mathrm{ci}}^{2,2}$, and $\bar{\boldsymbol{f}}_{\mathrm{ci}}^{3,1}$ and $\bar{\boldsymbol{f}}_{\mathrm{ci}}^{3,2}$, getting $\boldsymbol{f}_{\mathrm{ci}}^{2} \in \mathbb{R}^{c\times h_2\times w_2}$ and $\boldsymbol{f}_{\mathrm{ci}}^{3} \in \mathbb{R}^{c\times h_3\times w_3}$. In this way, all basic global features are enhanced in the channel to express defects.

C. Cross-scale Spatial Interaction Module

The single channel interaction is insufficient for local enhancement [41]. Similar to CBAM [41], we further employ spatial interaction and extend the vanilla spatial attention to a cross-scale and grouping scheme. Therefore, we propose the Cross-scale Spatial Interaction Module to achieve spatial enhancement for defect features. Taking CSIM^{1} as an example, we illustrate its detailed structure in Fig. 4. The inputs of CSIM^{1} are $\boldsymbol{f}_{\mathrm{ci}}^{1}$ and $\boldsymbol{f}_{\mathrm{ci}}^{2}$. Similar to CCIM, we also implement the separation-recombination strategy in CSIM. In contrast, this strategy is executed in the spatial dimension (*i.e.*, the height dimension). We first align the dimensions of $\boldsymbol{f}_{\mathrm{ci}}^{1}$ and $\boldsymbol{f}_{\mathrm{ci}}^{2}$, and upsample $\boldsymbol{f}_{\mathrm{ci}}^{2}$ to $c\times h_1\times w_1$. Then, through the separation-recombination strategy, we obtain two feature subsets, *i.e.*, $\{\hat{\boldsymbol{f}}_{\mathrm{ci}}^{1,1}, \hat{\boldsymbol{f}}_{\mathrm{ci}}^{2,1}\} \in \mathbb{R}^{c\times\frac{h_1}{2}\times w_1}$ and $\{\hat{\boldsymbol{f}}_{\mathrm{ci}}^{1,2}, \hat{\boldsymbol{f}}_{\mathrm{ci}}^{2,2}\} \in \mathbb{R}^{c\times\frac{h_1}{2}\times w_1}$.

These two feature subsets are processed by the Collaborative Spatial Attention Unit (CSAU). This unit can outline the defect regions in the features through cross-scale and cross-space interactions. Its structure is shown on the right side of Fig. 4. Taking $\{\hat{\boldsymbol{f}}_{\mathrm{ci}}^{1,2}, \hat{\boldsymbol{f}}_{\mathrm{ci}}^{2,2}\}$ for example, we first concatenate them along the height dimension. Then, similar to the traditional spatial attention, we sequentially perform the parallel channel-wise global average pooling and global max pooling, the channel concatenation, a convolution layer, and

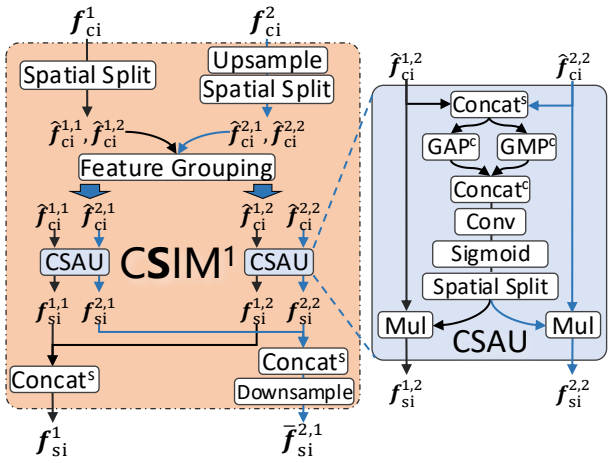

Fig. 4. Illustration of the Cross-scale Spatial Interaction Module and its key component of Collaborative Spatial Attention Unit. Here, Concat$^{c/s}$ means the channel/spatial concatenation, and GAP^c/GMP^c means the channel-wise global average/max pooling.

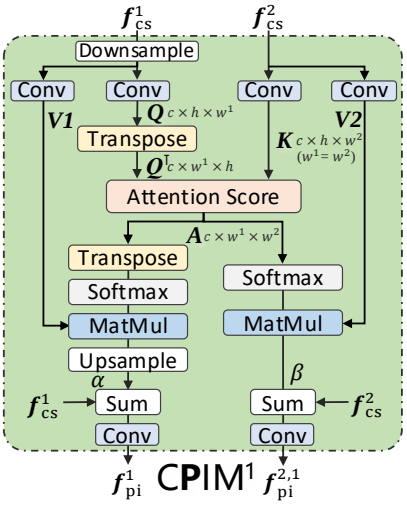

Fig. 5. Illustration of the Cross-scale Pixel Interaction Module.

the sigmoid activation function to obtain the spatial attention map, *i.e.*, $\boldsymbol{sa}^2 \in \mathbb{R}^{1\times h_1\times w_1}$. Such spatial interaction can learn common defect regions on cross-scale features, giving $\boldsymbol{sa}^2$ a powerful ability to outline the spatial shape of defect regions. It is separated along the height dimension, generating $\boldsymbol{sa}^{1,2} \in \mathbb{R}^{1\times \frac{h_1}{2}\times w_1}$ and $\boldsymbol{sa}^{2,2} \in \mathbb{R}^{1\times \frac{h_1}{2}\times w_1}$. They are used to enhance the corresponding $\hat{\boldsymbol{f}}_{\text{ci}}^{1,2}$ and $\hat{\boldsymbol{f}}_{\text{ci}}^{2,2}$ by multiplication, generating the output of CSAU, *i.e.*, $\boldsymbol{f}_{\text{si}}^{1,2} \in \mathbb{R}^{c\times \frac{h_1}{2}\times w_1}$ and $\boldsymbol{f}_{\text{si}}^{2,2} \in \mathbb{R}^{c\times \frac{h_1}{2}\times w_1}$.

After all feature subsets are processed in CSAUs, we obtain two enhanced feature subsets, *i.e.*, $\{\boldsymbol{f}_{\text{si}}^{1,1}, \boldsymbol{f}_{\text{si}}^{2,1}\}$ and $\{\boldsymbol{f}_{\text{si}}^{1,2}, \boldsymbol{f}_{\text{si}}^{2,2}\}$. Finally, we perform spatial concatenation on $\{\boldsymbol{f}_{\text{si}}^{1,1}, \boldsymbol{f}_{\text{si}}^{1,2}\}$ and $\{\boldsymbol{f}_{\text{si}}^{2,1}, \boldsymbol{f}_{\text{si}}^{2,2}\}$, and additionally downsample the latter subset, obtaining outputs of CSIM^1, *i.e.*, $\boldsymbol{f}_{\text{si}}^{1} \in \mathbb{R}^{c\times h_1\times w_1}$ and $\bar{\boldsymbol{f}}_{\text{si}}^{2,1} \in \mathbb{R}^{c\times h_2\times w_2}$. Following CSIM^1, we get $\bar{\boldsymbol{f}}_{\text{si}}^{2,2}$ and $\bar{\boldsymbol{f}}_{\text{si}}^{3,1}$ from CSIM^2, and $\bar{\boldsymbol{f}}_{\text{si}}^{3,2}$ and $\boldsymbol{f}_{\text{si}}^{4}$ from CSIM^3. As shown in Fig. 2, we also fuse $\bar{\boldsymbol{f}}_{\text{si}}^{2,1}$ and $\bar{\boldsymbol{f}}_{\text{si}}^{2,2}$, and $\bar{\boldsymbol{f}}_{\text{si}}^{3,1}$ and $\bar{\boldsymbol{f}}_{\text{si}}^{3,2}$ through fusion units, getting $\boldsymbol{f}_{\text{si}}^{2} \in \mathbb{R}^{c\times h_2\times w_2}$ and $\boldsymbol{f}_{\text{ci}}^{3} \in \mathbb{R}^{c\times h_3\times w_3}$. Moreover, the basic features $\boldsymbol{f}_{\text{b}}^{i}$ are added to $\boldsymbol{f}_{\text{si}}^{i}$, generating $\boldsymbol{f}_{\text{cs}}^{i} \in \mathbb{R}^{c\times h_i\times w_i}$. In this way, $\boldsymbol{f}_{\text{cs}}^{i}$ can perceive and highlight defect regions in detail.

D. Cross-scale Pixel Interaction Module

CCIM and CSIM enhance $\boldsymbol{f}_{\text{b}}^{i}$ locally in the channel and spatial. $\boldsymbol{f}_{\text{cs}}^{i}$ can effectively characterize the defect regions. We further introduce global interaction to model the relationship between the defect region and the background in $\boldsymbol{f}_{\text{cs}}^{i}$, making the defect region more prominent. Thus, we propose the Cross-scale Pixel Interaction Module, which extends the vanilla self-attention to the cross-scale and channel-wise scheme.

Taking CPIM^1 as an example, we illustrate its detailed structure in Fig. 5. The inputs of CPIM^1 are $\boldsymbol{f}_{\text{cs}}^{1}$ and $\boldsymbol{f}_{\text{cs}}^{2}$. We first downsample $\boldsymbol{f}_{\text{cs}}^{1}$ to match the size of $\boldsymbol{f}_{\text{cs}}^{2}$, *i.e.*, $c\times h_2\times w_2$. For convenience, we here denote the size of $\boldsymbol{f}_{\text{cs}}^{1}$ as $c\times h\times w^1$, and the size of $\boldsymbol{f}_{\text{cs}}^{2}$ as $c\times h\times w^2$, where w^1 equals to w^2, as shown in Fig. 5. Then, similar to the vanilla self-attention, we get $\{\boldsymbol{Q}, \boldsymbol{V1}\} \in \mathbb{R}^{c\times h\times w^1}$ from $\boldsymbol{f}_{\text{cs}}^{1}$, and $\{\boldsymbol{K}, \boldsymbol{V2}\} \in \mathbb{R}^{c\times h\times w^2}$ from $\boldsymbol{f}_{\text{cs}}^{2}$. The attention score, *i.e.*, $\boldsymbol{A} \in \mathbb{R}^{c\times w^1\times w^2}$, is obtained by performing channel-wise matrix multiplication (denoted as $\circledast$) on $\boldsymbol{Q}^\top$ and $\boldsymbol{K}$[1], which is different from that in the vanilla self-attention. The attention score $\boldsymbol{A}$ models the pixel-level relationships between the defect region and the background. We transfer the relationship encoded in $\boldsymbol{A}$ to $\boldsymbol{V1}$ and $\boldsymbol{V2}$ through $\circledast$. The subsequent operations are the same as in vanilla self-attention, except that we upsample the downsampled features to align their sizes, as shown in Fig. 5. In this way, we get the output of CPIM^1, denoted as $\boldsymbol{f}_{\text{pi}}^{1} \in \mathbb{R}^{c\times h\times w^1}$ (*i.e.*, $\mathbb{R}^{c\times h_1\times w_1}$) and $\boldsymbol{f}_{\text{pi}}^{2,1} \in \mathbb{R}^{c\times h\times w^2}$ (*i.e.*, $\mathbb{R}^{c\times h_2\times w_2}$). The computational complexity of $\circledast$ is significantly smaller than that of the original matrix multiplication, achieving a balance between effectiveness and efficiency.

Following CPIM^1, we get $\boldsymbol{f}_{\text{pi}}^{2,2}$ and $\boldsymbol{f}_{\text{pi}}^{3,1}$ from CPIM^2, and $\boldsymbol{f}_{\text{pi}}^{3,2}$ and $\boldsymbol{f}_{\text{pi}}^{4}$ from CPIM^3. As shown Fig. 2, we also fuse $\boldsymbol{f}_{\text{pi}}^{2,1}$ and $\boldsymbol{f}_{\text{pi}}^{2,2}$, and $\boldsymbol{f}_{\text{pi}}^{3,1}$ and $\boldsymbol{f}_{\text{pi}}^{3,2}$ through fusion units, getting $\boldsymbol{f}_{\text{pi}}^{2} \in \mathbb{R}^{c\times h_2\times w_2}$ and $\boldsymbol{f}_{\text{pi}}^{3} \in \mathbb{R}^{c\times h_3\times w_3}$. Through the three modules, the discriminative feature $\boldsymbol{f}_{\text{pi}}^{i}$ contains rich local and global information, which is beneficial to the subsequent defect segmentation in the defect reasoner.

E. Loss Function

As shown in Fig. 2, the defect reasoner is composed of four DR blocks. For DR^2, DR^3, and DR^4 blocks, there are two convolution layers, a dropout layer with the hyper-parameter of 0.5, and a deconvolution layer in sequence. DR^1 block contains only two convolution layers. The final segmentation map $\boldsymbol{S}^1$ with the size of 352×352 and three side segmentation maps $\boldsymbol{S}^2$, $\boldsymbol{S}^3$, and $\boldsymbol{S}^4$ with the size of 352×352 are all generated using SegHeads and upsampling operations. Here, the SegHead is a regular convolution layer with the kernel size of 3×3.

For these segmentation maps, we adopt the widely used deep supervision strategy [45] in the training phase. Moreover,

[1] $\boldsymbol{Q}^\top \circledast \boldsymbol{K} = \mathbb{R}^{(c\times w^1\times h)}\circledast(c\times h\times w^2) = \mathbb{R}^{c\times(w^1\times h)}\circledast(h\times w^2) = \mathbb{R}^{c\times w^1\times w^2}$.

TABLE I
QUANTITATIVE COMPARISONS (%) WITH STATE-OF-THE-ART METHODS. THE BEST TWO RESULTS ARE MARKED IN **RED** AND **BLUE**, RESPECTIVELY.

Methods	Backbone	#Param (M)↓	FLOPs (G)↓	Speed (FPS)↑	Man-made (965) PA↑	wFm↑	IoU↑	Dice↑	Natural (165) PA↑	wFm↑	IoU↑	Dice↑
$UNet_{15}$ [42]	VGG	31.04	91.34	213	79.8	79.9	66.4	78.1	64.0	68.7	57.2	66.5
FCN_{15} [11]	VGG	18.64	40.58	264	79.3	78.2	64.8	77.1	59.9	65.2	53.3	63.5
$SegNet_{17}$ [12]	VGG	29.44	75.82	194	75.8	76.8	62.3	74.8	57.6	62.6	50.8	60.5
$Deeplabv3_{17}$ [13]	ResNet	16.48	11.88	142	81.1	80.8	67.4	79.1	63.7	68.7	57.3	66.8
$SegFormer_{21}$ [24]	MiT	84.59	47.13	19	77.0	77.2	63.3	75.6	65.9	66.6	55.5	65.6
$Segmenter_{21}$ [25]	ViT	103.15	144.94	32	81.5	83.0	69.2	80.6	69.8	72.9	62.1	71.4
$SwinUnet_{22}$ [43]	SwinT	41.46	21.38	42	78.6	74.6	61.2	74.2	59.3	62.3	50.0	60.8
SAM_{23} [26]	ViT	97.05	385.25	8	83.9	82.8	69.7	81.0	69.8	72.6	61.6	71.1
$TransUNet_{24}$ [44]	ViT	105.57	60.77	47	83.2	82.7	69.6	80.5	72.0	73.5	63.1	72.0
$EDRNet_{20}$ [36]	ResNet	39.31	79.61	47	81.9	82.5	69.0	80.1	68.3	70.8	60.5	69.2
$EMINet_{21}$ [37]	ResNet	99.13	263.87	91	81.7	83.4	69.6	80.7	68.6	71.6	60.7	70.0
$TSERNet_{22}$ [9]	ResNet	189.64	502.36	45	80.7	82.0	68.7	79.6	66.2	69.3	59.3	67.7
$DACNet_{22}$ [40]	ResNet	98.39	269.85	55	77.0	81.2	66.3	77.9	61.7	66.0	55.4	63.7
$CSEPNet_{22}$ [38]	VGG	18.78	110.17	35	81.1	82.9	69.3	80.3	65.1	69.0	59.1	67.2
$ICON_{23}$ [20]	PVT	65.68	61.79	34	**84.6**	81.5	68.8	80.3	70.8	73.0	62.2	71.9
$GeleNet_{23}$ [21]	PVT	25.45	11.66	83	83.3	83.4	70.2	81.1	70.6	73.4	62.7	71.7
$TSCNet_{24}$ [22]	VGG	103.56	116.61	53	84.5	83.6	70.7	81.5	70.4	72.9	62.4	71.3
$MCnet_{21}$ [5]	DenseNet	38.44	90.29	39	79.4	82.3	68.1	79.7	69.5	73.3	62.6	71.6
$NaDiNet_{23}$ [7]	DenseNet	33.11	34.86	26	84.2	**84.4**	**71.3**	**82.1**	**72.5**	**75.5**	**65.2**	**73.6**
OCINet (Ours)	PVT	29.95	18.82	108	**85.4**	**84.7**	**71.8**	**82.4**	**72.9**	**76.0**	**65.5**	**74.2**

we use a hybrid loss function $\mathcal{L}_{\text{hyb}}$ containing the pixel-level binary cross-entropy loss function [46] and the map-level intersection over union loss functions [47] to supervise each segmentation map. Binary cross-entropy loss function is primarily used for pixel-level classification, capturing local details, while the intersection over union loss focuses more on the overall structure at the image level. The combination of the two can pay attention to both local and global features, thereby improving the segmentation accuracy [7], [9], [38]. We define the total loss function $\mathcal{L}_{\text{total}}$ as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}^1_{\text{hyb}} + \mathcal{L}^2_{\text{hyb}} + \mathcal{L}^3_{\text{hyb}}/2 + \mathcal{L}^4_{\text{hyb}}/4, \tag{1}$$

$$\mathcal{L}^i_{\text{hyb}} = l_{\text{bce}}(\boldsymbol{S}^i, \boldsymbol{G}) + l_{\text{iou}}(\boldsymbol{S}^i, \boldsymbol{G}), \tag{2}$$

$$l_{\text{bce}} = -\sum_{j=1}^{W \cdot H} [\boldsymbol{G}(j)\log(\boldsymbol{S}(j)) + (1-\boldsymbol{G}(j))\log(1-\boldsymbol{S}(j))], \tag{3}$$

$$l_{\text{iou}} = 1 - \frac{\sum_{j=1}^{W \cdot H} \boldsymbol{S}(j) \cdot \boldsymbol{G}(j)}{\sum_{j=1}^{W \cdot H} [\boldsymbol{S}(j) + \boldsymbol{G}(j) - \boldsymbol{S}(j) \cdot \boldsymbol{G}(j)]}, \tag{4}$$

where l_{bce} and l_{iou} are the binary cross-entropy loss function and the intersection over union loss function, respectively, $\boldsymbol{G}$ is the ground truth map, and W and H respectively represent the width and height of $\boldsymbol{G}$.

IV. EXPERIMENTS

A. Experimental Setup

1) Datasets: We conduct quantitative and qualitative experiments on the NRSD segmentation dataset, namely NRSD-MN dataset [5]. The dataset consists of three subsets, *i.e.*, training set, validation set, and test set, with a total of 4,101 NRSD images. There are two types of NRSD images in the NRSD-MN dataset, including 3,936 man-made NRSD images and 165 natural NRSD images. Here, the man-made NRSD images refer to that the defects are artificially created through cutting, grinding, turning, and welding [5]. To be specific, we only use the training set containing 2,086 man-made NRSD images and the test set containing 965 man-made NRSD images and 165 natural NRSD images in our experiments, just like [5], [7].

2) Implementation Details: For data processing, we rotate and flip the training images, and resize them to 352×352 for network training, while retaining the resizing operation only during network testing. For network implementation, we achieve our OCINet on the PyTorch platform [48] with an NVIDIA RTX 3090 GPU (24GB memory). In our OCINet, we initialize the feature extractor using the pre-trained PVT-v2-b2 parameters and the newly added layers using the "Kaiming" method [49]. For training hyper-parameters, we adopt the Adam optimizer [50] for parameter optimization, and set the batch size, the initial learning rate, the training epoch, and the decay epoch to 16, $1e^{-4}$, 70, and 30, respectively. Notably, the rationale of hyper-parameter selection refers to the first NRSD segmentation method MCnet [5] and our previous experience.

3) Evaluation Metrics: We adopt four commonly used evaluation metrics to comprehensively assess the segmentation performance in our experiments, including pixel accuracy (PA) [11], weighted F-measure (wFm) score [51], intersection over union (IoU) [52], and dice coefficient (Dice) [53]. For all metrics, the larger the value, the better the performance.

PA [11] represents the ratio of correctly segmented pixels to total pixels, defined as follows:

$$PA = \frac{\sum_{i=0}^{1} p_{ii}}{\sum_{i=0}^{1} \sum_{j=0}^{1} p_{ij}}, \tag{5}$$

where p_{ij} means the number of pixels that belong to class i but are predicted to be class j.

wFm [51] is a weighted average of precision and recall, taking into account the problem of sample imbalance, defined as follows:

$$wFm = \frac{(1+\beta^2) \cdot Precision^w \cdot Recall^w}{\beta^2 \cdot Precision^w + Recall^w}, \tag{6}$$

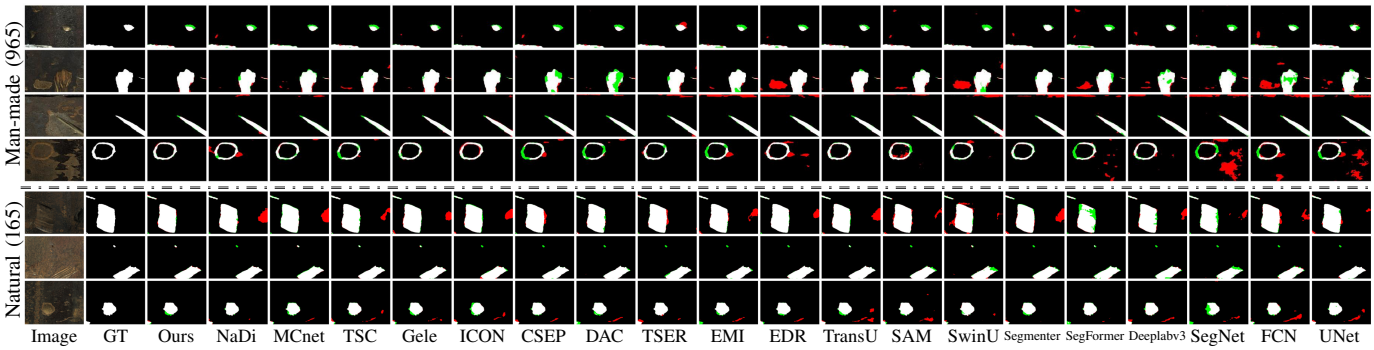

Fig. 6. Qualitative comparisons with 19 state-of-the-art methods on the NRSD-MN dataset. We omit the 'net' suffix from some method names, such as changing 'NaDiNet' to 'NaDi'. In these segmentation maps, we annotate the correctly segmented pixels as white, the incorrectly segmented pixels as red, and the missed segmented pixels as green. Please zoom-in for details.

where β^2 is set to 1.

IoU [52] is the most representative segmentation metric. It is the ratio of the intersection and union of the predicted segmentation map and the ground truth, defined as follows:

$$IoU = \frac{|\boldsymbol{S}^1 \cap \boldsymbol{G}|}{|\boldsymbol{S}^1 \cup \boldsymbol{G}|}. \tag{7}$$

Dice [53] is a set similarity measurement function and also a representative segmentation metric, defined as follows:

$$Dice = \frac{2|\boldsymbol{S}^1 \cap \boldsymbol{G}|}{|\boldsymbol{S}^1| + |\boldsymbol{G}|}. \tag{8}$$

B. Comparison with State-of-the-arts

We compare our OCINet with 19 state-of-the-art methods of three types, including CNN-based and transformer-based methods. The first type is the segmentation method for biomedical and natural images. It has nine methods, *i.e.*, UNet [42], FCN [11], SegNet [12], Deeplabv3 [13], SegFormer [24], Segmenter [25], SwinUnet [43], SAM [26], and TransUNet [44]. The second type is the salient object detection method for strip steel surface defects, natural images, and optical remote sensing images. It has eight methods, *i.e.*, EDRNet [36], EMINet [37], TSERNet [9], DACNet [40], CSEPNet [38], ICON [20], GeleNet [21], and TSCNet [22]. The third type is the specialized NRSD segmentation method. It has two methods, *i.e.*, MCnet [5] and NaDiNet [7]. For a fair comparison, we retrain the first and second types of methods on the NRSD-MN dataset with default parameter settings until their losses converge. We then test these retrained methods on the same test set as ours to obtain segmentation maps for comparison.

1) Quantitative Comparison: We show the quantitative comparisons of our OCINet and 19 state-of-the-art methods in Tab. I. Overall, our OCINet achieves better segmentation performance than all three types of methods on man-made and natural NRSD images, no matter whether they are based on CNN or transformer. Compared with the general segmentation method, the advantages of our OCINet are obvious, reaching about 5.0% on man-made NRSD images and about 7.9% on natural NRSD images. Compared with the salient object detection method, the advantages of our OCINet are also outstanding, especially on natural NRSD images, reaching about 5.2% in IoU and about 5.2% in Dice. The gap between the specialized NRSD segmentation methods and our OCINet is also considerable. On man-made images, our OCINet is 6.0% better than MCnet in PA. On natural images, our OCINet is 0.6% better than NaDiNet in Dice. The segmentation performance in Tab. I shows that NRSD segmentation has huge room for development but is also very challenging.

2) Model Complexity Comparison: We report the model parameters (*i.e.*, #Param), the floating point operations (*i.e.*, FLOPs), and the inference speed (without I/O time) of all methods in Tab. I. Overall, our method ranks sixth in terms of the number of parameters, third in FLOPs, and fifth in inference speed among all methods. Specifically, compared to the suboptimal methods (such as NaDiNet, TSCNet, and TransUNet), our method has fewer parameters than these three methods, *i.e.*, 29.95M *v.s.* 33.11M/103.56M/105.57M. Compared to methods with high FLOPs (such as TSERNet, SAM, and DACNet), our method outperforms them in segmentation performance. This indicates that the effectiveness of our method does not stem from an increase in the number of parameters and FLOPs but rather from the effective design of the module structure. In terms of inference speed, our method achieves over 100 frames per second, only inferior to earlier segmentation methods with simple structures (such as UNet, FCN, SegNet, and Deeplabv3). We summarize that our method achieves optimal segmentation performance while maintaining a small model complexity.

Overall, our method ranks sixth in terms of parameter count, fourth in computational complexity, and fifth in inference speed among all methods

3) Qualitative Comparison: We show the qualitative comparisons of our OCINet and 19 state-of-the-art methods on man-made and natural NRSD images in Fig. 6. In these segmentation maps, we annotate the correctly segmented pixels as white, the incorrectly segmented pixels as red, and the missed segmented pixels as green. There are several typical NRSD scenes in Fig. 6, such as irregular, tiny, and multiple defects, low contrast, low illumination, and cluttered background. We can observe that the segmentation maps generated by our method are the most accurate among all comparison methods. The segmentation maps generated by the general segmentation methods are the worst among all methods. Their segmentation

TABLE II
QUANTITATIVE RESULTS (%) OF EVALUATING THE ADVANTAGE OF THE GLOBAL-LOCAL-GLOBAL STRATEGY.

Models	#Param (M)↓	Man-made (965) IoU↑	Man-made (965) Dice↑	Natural (165) IoU↑	Natural (165) Dice↑
Global-global-local	29.952	71.5	82.1	65.0	73.9
Global-local-global (**Ours**)	29.952	**71.8** +0.3	**82.4** +0.3	**65.5** +0.5	**74.2** +0.3

maps incorrectly segment the background. The salient object detection methods can highlight defects, but sometimes defect regions are incomplete. The specialized segmentation methods are comparable to our method, but the details of the defects are still slightly inferior. The results of the qualitative comparison are basically consistent with those of the quantitative comparison, which demonstrates that our method is an excellent NRSD segmenter.

C. Ablation Studies

We conduct comprehensive ablation studies to evaluate the effectiveness of each part of our OCINet. Specifically, we investigate our OCINet from the following three aspects: 1) the advantage of the global-local-global strategy, 2) the contribution of three modules, and 3) the effectiveness of the cross-scale interaction scheme in three modules.

1) The advantage of the global-local-global strategy: We implement the global-local-global strategy in OCINet by executing CCIM, CSIM, and CPIM sequentially to achieve channel, spatial, and pixel interactions. To study the advantage of the global-local-global strategy, we swap the order of the three modules to CPIM, CCIM, and CSIM to implement the global-global-local strategy. The quantitative results are shown in Tab. II. When the modules are the same, simply swapping the order in which local and global enhancements are performed can hurt the performance of our OCINet. Moreover, models with the global-local-global strategy and global-global-local strategy have the same number of parameters. This shows that our carefully designed global-local-global strategy is superior and also shows the importance of interweaving global and local enhancements for NRSD segmentation.

2) The contribution of three modules: To evaluate the contribution of our CCIM, CSIM, and CPIM, we provide various combinations of these three modules: 1) Base, which is the simple encoder-decoder framework, 2) Base+CCIM, 3) Base+CSIM, 4) Base+CCIM+CSIM, and 5) Base+CPIM. We show the quantitative results in Tab. III. The performance of the above five variants is worse than the full model. The individual contribution of CCIM and CSIM responsible for local enhancement is smaller than CPIM, but their combined contribution is higher than CPIM. From the perspective of the number of parameters, CPIM has the largest increase, with an addition of 1.479M parameters. CCIM and CSIM together increase the number of parameters by 0.616M, enhancing the performance of "Base" by 0.5∼1.8%. With all three modules working together, our full model surpasses "Base" by around 1.0% on man-made images and by around 2.0% on natural images, with an increase of around 2M parameters. The above

TABLE III
QUANTITATIVE RESULTS (%) OF EVALUATING THE CONTRIBUTION OF THREE MODULES AND THE EFFECTIVENESS OF THE CROSS-SCALE INTERACTION SCHEME. C., S., AND P. REPRESENT CCIM, CSIM, AND CPIM, RESPECTIVELY.

No.	Models	#Param (M)↓	Man-made (965) IoU↑	Man-made (965) Dice↑	Natural (165) IoU↑	Natural (165) Dice↑
1	Base	27.857	70.7	81.6	63.4	72.1
2	Base+C.	28.177	71.3 +0.6	82.0 +0.4	64.3 +0.9	73.0 +0.9
3	Base+S.	28.153	71.2 +0.5	81.9 +0.3	64.3 +0.9	73.4 +1.3
4	Base+C.+S.	28.473	71.4 +0.7	82.1 +0.5	65.1 +1.7	73.9 +1.8
5	Base+P.	29.336	71.4 +0.7	82.0 +0.4	64.6 +1.2	73.6 +1.5
6	Base+vanilla CA	28.177	71.1	81.8	64.0	72.8
7	Base+vanilla SA	28.152	70.7	81.6	64.1	73.2
8	Base+SWSAM [21]	28.154	70.9	81.7	63.9	73.1
9	Base+RSCAM [54]	28.176	71.2	81.9	63.8	72.8
10	Base+LSKM [55]	28.507	71.0	81.7	64.3	73.2
11	Base+C. *w/o CS*	27.865	70.9	81.6	63.9	72.9
12	Base+S. *w/o CS*	27.857	70.8	81.6	63.8	72.6
13	Base+P. *w/o CS*	29.436	71.0	81.8	63.7	72.5
14	Base+C.+S.+P. *w/o CS*	29.444	71.3	82.0	64.4	73.4
15	Base+C.+S.+P. (**Ours**)	29.952	**71.8** +1.1	**82.4** +0.8	**65.5** +2.1	**74.2** +2.1

analysis intuitively demonstrates that the contribution of each module is clear, and with the complementary effect of the three modules, our full model achieves the best performance.

In addition, we modify CCIM and CSIM to vanilla channel attention (CA) and spatial attention (SA), that is we directly concatenate cross-scale features without adopting the unique separation and recombination strategy. We provide two variants, named "Base+vanilla CA" and "Base+vanilla SA" in Tab. III. Compared with CCIM and CSIM, the ability of the vanilla CA and SA to improve performance is not as good as theirs. Moreover, the number of parameters of our CCIM is almost the same as the vanilla CA (*i.e.*, 28.177M *v.s.* 28.177M), and the same goes for our CSIM compared to the vanilla SA (*i.e.*, 28.153M *v.s.* 28.152M). This demonstrates that the vanilla CA and SA are suboptimal, and the separation-recombination strategy in CCIM and CSIM is effective and efficient for feature interactions.

We also integrate existing state-of-the-art attention modules (such as Shuffle Weighted Spatial Attention Module (SWSAM) [21], Rectangular Self-Calibration Attention Module (RSCAM) [54], and Large Selective Kernel Module (LSKM) [55]) into "Base", providing three variations, named "Base+SWSAM", "Base+RSCAM", and "Base+LSKM" in Tab. III. These three state-of-the-art attention modules all improve the performance of "Base", but their improvement is inferior to the three modules we proposed (except for the performance of "Base+LSKM" on natural images). This demonstrates that the three modules we proposed have advantages compared to existing state-of-the-art attention modules.

3) The effectiveness of the cross-scale interaction scheme in three modules: In our proposed CCIM, CSIM, and CPIM, the cross-scale interaction scheme plays a crucial role in exploring the complementary information of features at adjacent scales, and is also one of the cores of our OCINet. Therefore, we here assess the effectiveness of the cross-scale interaction

scheme in our modules. We remove the cross-scale interaction scheme in CCIM, CSIM, and CPIM, and perform the vanilla channel attention, spatial attention, and self-attention on single-scale features to achieve feature enhancement. We provide four variants, namely "Base+C. *w/o CS*", "Base+S. *w/o CS*", "Base+P. *w/o CS*", and "Base+C.+S.+P. *w/o CS*", in Tab. III. Without the complementary information across scales, the performance of all four variants degrades. This is foreseeable. Because the interaction of features of different granularities can explore the common regions among them, which is better than feature enhancement under a single granularity. The performance degradation indicates that cross-scale interaction is effective in highlighting defect regions and demonstrates the indispensability of this scheme in our modules. Moreover, the cross-scale interaction scheme only occupies 0.5M parameters in the final model, leading to a 1.1% performance improvement in IoU on natural images, which also demonstrates that the scheme is relatively efficient.

D. Limitation and Discussion

Our proposed OCINet achieves a relatively balanced trade-off between performance and mode complexity. However, in scenarios with limited hardware or handheld defect inspection devices, the parameter count of our OCINet is still relatively large, *i.e.*, 29.95M, limiting its application in real-world scenarios. In addition, our proposed method only achieves a segmentation performance of approximately 70% in IoU, and there is still a lot of room for improvement.

For future work, we have two directions. First, we aim to develop a lightweight NRSD segmentation model that balances model parameters and performance, with the goal of achieving better model performance with fewer parameters. Second, we plan to make appropriate modifications to large segmentation models, such as SAM [26] and SEEM [56], rather than directly fine-tuning them to improve the performance of NRSD segmentation task and break through performance bottlenecks.

V. Conclusion

In this paper, we propose the first transformer-based solution, termed OCINet, for NRSD segmentation. The core of OCINet lies in the global-local-global strategy and the cross-scale interaction scheme. These two cores are effective in handling the unique scenes of NRSD segmentation. Following the strategy, we adopt PVT as the global feature extractor, CCIM and CSIM for local enhancement, and CPIM for global enhancement in our OCINet. Following the scheme, we sequentially perform channel, space, and pixel interactions on features at adjacent scales in our OCINet. By integrating all components, our OCINet consistently outperforms 19 state-of-the-art methods on the challenging NRSD-MN dataset. Comprehensive ablation experiments further validate the effectiveness of our modules, strategy, and scheme for NRSD segmentation.

References

[1] N. Neogi, D. K. Mohanta, and P. K. Dutta, "Review of vision-based steel surface inspection systems," *EURASIP J. Image. Video. Process.*, pp. 1–19, Nov. 2014.

[2] Q. Luo, X. Fang, L. Liu, C. Yang, and Y. Sun, "Automated visual defect detection for flat steel surface: A survey," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 3, pp. 626–644, Mar. 2020.

[3] X. Ni, Z. Ma, J. Liu, B. Shi, and H. Liu, "Attention network for rail surface defect detection via consistency of intersection-over-union(iou)-guided center-point estimation," *IEEE Trans. Industr. Inform.*, vol. 18, no. 3, pp. 1694–1705, Mar. 2022.

[4] S. Ma, K. Song, M. Niu, H. Tian, Y. Wang, and Y. Yan, "Shape-consistent one-shot unsupervised domain adaptation for rail surface defect segmentation," *IEEE Trans. Industr. Inform.*, vol. 19, no. 9, pp. 9667–9679, Sept. 2023.

[5] D. Zhang, K. Song, J. Xu, Y. He, M. Niu, and Y. Yan, "MCnet: Multiple context information segmentation network of no-service rail surface defects," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–9, Jan. 2021.

[6] D. Zhang, K. Song, J. Xu, H. Dong, and Y. Yan, "An image-level weakly supervised segmentation method for no-service rail surface defect with size prior," *Mech. Syst. Signal Process.*, vol. 165, pp. 1–14, Feb. 2022.

[7] G. Li, C. Han, and Z. Liu, "No-service rail surface defect segmentation via normalized attention and dual-scale interaction," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–10, Jul. 2023.

[8] I. Ahmed, G. Jeon, and F. Piccialli, "From artificial intelligence to explainable artificial intelligence in industry 4.0: A survey on what, how, and where," *IEEE Trans. Industr. Inform.*, vol. 18, no. 8, pp. 5031–5042, Aug. 2022.

[9] C. Han, G. Li, and Z. Liu, "Two-stage edge reuse network for salient object detection of strip steel surface defects," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–12, Aug. 2022.

[10] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE ICCV*, Oct. 2013, pp. 3166–3173.

[11] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE CVPR*, Jun. 2015, pp. 3431–3440.

[12] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[13] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE CVPR*, Jun. 2016, pp. 770–778.

[15] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE CVPR*, Jul. 2017, pp. 2261–2269.

[16] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "PVT v2: Improved baselines with pyramid vision transformer," *Comput. Vis. Media*, vol. 8, pp. 415–424, Sept. 2022.

[17] A. Kaseb, M. Khaled, and O. Galal, "Convolutional neural networks for semantic segmentation: A recent survey," in *Proc. ACIT*, Nov. 2022, pp. 1–7.

[18] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3523–3542, Jul. 2022.

[19] X. Li, H. Ding, H. Yuan, W. Zhang, J. Pang, G. Cheng, K. Chen, Z. Liu, and C. C. Loy, "Transformer-based visual segmentation: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 10 138–10 163, Dec. 2024.

[20] M. Zhuge, D.-P. Fan, N. Liu, D. Zhang, D. Xu, and L. Shao, "Salient object detection via integrity learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3738–3752, Mar. 2023.

[21] G. Li, Z. Bai, Z. Liu, X. Zhang, and H. Ling, "Salient object detection in optical remote sensing images driven by transformer," *IEEE Trans. Image Process.*, vol. 32, pp. 5257–5269, Sept. 2023.

[22] G. Li, Z. Bai, and Z. Liu, "Texture-semantic collaboration network for ORSI salient object detection," *IEEE Trans. Circuits Syst. II-Express Briefs*, vol. 71, no. 4, pp. 2464–2468, Apr. 2024.

[23] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proc. IEEE CVPR*, Jun. 2019, pp. 3146–3154.

[24] E. Xie, W. Wang, Z. Yu, A. Anandkuma, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Proc. NeurIPS*, Dec. 2021, pp. 12 077–12 090.

[25] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, "Segmenter: Transformer for semantic segmentation," in *Proc. IEEE ICCV*, Oct. 2021, pp. 7242–7252.

[26] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollr, and R. Girshick, "Segment anything," in *Proc. IEEE ICCV*, Oct. 2023, pp. 3992–4003.
[27] C. Lu, D. de Geus, and G. Dubbelman, "Content-aware token sharing for efficient semantic segmentation with vision transformers," in *Proc. IEEE CVPR*, Jun. 2023, pp. 23 631–23 640.
[28] N. Norouzi, S. Orlova, D. De Geus, and G. Dubbelman, "ALGM: Adaptive local-then-global token merging for efficient semantic segmentation with plain vision transformers," in *Proc. IEEE CVPR*, Jun. 2024, pp. 15 773–15 782.
[29] X. Shi, Z. Yin, G. Han, W. Liu, L. Qin, Y. Bi, and S. Li, "BSSNet: A real-time semantic segmentation network for road scenes inspired from autoencoder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 5, pp. 3424–3438, May 2024.
[30] L. Yang, Y. Bai, F. Ren, C. Bi, and R. Zhang, "LCFNets: Compensation strategy for real-time semantic segmentation of autonomous driving," *IEEE Trans. Intell. Veh.*, vol. 9, no. 4, pp. 4715–4729, Apr. 2024.
[31] J. Fan, B. Gao, Q. Ge, Y. Ran, J. Zhang, and H. Chu, "SegTransConv: Transformer and cnn hybrid method for real-time semantic segmentation of autonomous vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 2, pp. 1586–1601, Feb. 2024.
[32] W. Wang, G. Sun, and L. Van Gool, "Looking beyond single images for weakly supervised semantic segmentation learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 3, pp. 1635–1649, Mar. 2024.
[33] J. Wang, K. Song, D. Zhang, M. Niu, and Y. Yan, "Collaborative learning attention network based on RGB image and depth image for surface defect inspection of no-service rail," *IEEE/ASME Trans. Mechatron.*, vol. 27, no. 6, pp. 4874–4884, Sept. 2022.
[34] W. Zhou and J. Hong, "FHENet: Lightweight feature hierarchical exploration network for real-time rail surface defect inspection in RGB-D images," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–8, Jan. 2023.
[35] L. Huang and A. Gong, "Surface defect detection for no-service rails with skeleton-aware accurate and fast network," *IEEE Trans. Industr. Inform.*, vol. 20, no. 3, pp. 4571–4581, Mar. 2024.
[36] G. Song, K. Song, and Y. Yan, "EDRNet: Encoder-decoder residual network for salient object detection of strip steel surface defects," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 12, pp. 9709–9719, Dec. 2020.
[37] X. Zhou, H. Fang, X. Fei, R. Shi, and J. Zhang, "Edge-aware multi-level interactive network for salient object detection of strip steel surface defects," *IEEE Access*, vol. 9, pp. 149 465–149 476, Nov. 2021.
[38] T. Ding, G. Li, Z. Liu, and Y. Wang, "Cross-scale edge purification network for salient object detection of steel defect images," *Meas.*, vol. 199, pp. 1–11, Aug. 2022.
[39] K. Shen, X. Zhou, and Z. Liu, "MINet: Multiscale interactive network for real-time salient object detection of strip steel surface defects," *IEEE Trans. Industr. Inform.*, vol. 20, no. 5, pp. 7842–7852, May 2024.
[40] X. Zhou, H. Fang, Z. Liu, B. Zheng, Y. Sun, J. Zhang, and C. Yan, "Dense attention-guided cascaded network for salient object detection of strip steel surface defects," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–14, Mar. 2022.
[41] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. ECCV*, Sep. 2018, pp. 3–19.
[42] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, Oct. 2015, pp. 234–241.
[43] H. Cao, Y. Wang, J. Chen, D. Jiang4, X. Zhang, Q. Tian, and M. Wang, "Swin-Unet: Unet-like pure transformer for medical image segmentation," in *Proc. ECCVW*, Aug. 2022, pp. 205–218.
[44] J. Chen, J. Mei, X. Li, Y. Lu, Q. Yu, Q. Wei, X. Luo, Y. Xie, E. Adeli, Y. Wang *et al.*, "TransUNet: Rethinking the U-Net architecture design for medical image segmentation through the lens of transformers," *Med. Image Anal.*, vol. 97, p. 103280, Oct. 2024.
[45] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," in *Proc. AAAI*, vol. 38, May 2015, pp. 562–570.
[46] P.-T. de Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Ann. Oper. Res.*, vol. 134, no. 1, pp. 19–67, Feb. 2005.
[47] G. Máttyus, W. Luo, and R. Urtasun, "DeepRoadMapper: Extracting road topology from aerial images," in *Proc. IEEE ICCV*, Oct. 2017, pp. 3458–3466.
[48] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. NeurIPS*, Dec. 2019, pp. 8024–8035.
[49] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE ICCV*, Dec. 2015, pp. 1026–1034.
[50] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, May 2015, pp. 1–15.
[51] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps," in *Proc. IEEE CVPR*, Jun. 2014, pp. 248–255.
[52] M. Everingham, A. Zisserman *et al.*, "The 2005 pascal visual object classes challenge," in *Proc. Machine Learning Challenges Workshop*, vol. 3944, 2006, pp. 117–176.
[53] L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.
[54] Z. Ni, X. Chen, Y. Zhai, Y. Tang, and Y. Wang, "Context-guided spatial feature reconstruction for efficient semantic segmentation," in *Proc. ECCV*, Sept. 2024, pp. 239–255.
[55] Y. Li, X. Li, Y. Dai, Q. Hou, L. Liu, Y. Liu, M.-M. Cheng, and J. Yang, "LSKNet: A foundation lightweight backbone for remote sensing," *Int. J. Comput. Vis.*, vol. 133, pp. 1410–1431, Mar. 2025.
[56] X. Zou, J. Yang, H. Zhang, F. Li, L. Li, J. Wang, J. G. Lijuan Wang, and Y. J. Lee, "Segment everything everywhere all at once," in *Proc. NeurIPS*, Dec. 2023, pp. 19 769–19 782.

Gongyang Li received the Ph.D. degree from Shanghai University, Shanghai, China, in 2022. He is currently an Associate Professor at the School of Communication and Information Engineering, Shanghai University, Shanghai, China. From 2021 to 2022, he was a Visiting Ph.D. Student at the School of Computer Science and Engineering, Nanyang Technological University, Singapore. From 2022 to 2024, he worked as a Postdoctor at Shanghai University, Shanghai, China. His research interests include multi-model image processing, saliency detection, and image/video segmentation.

Xiaofei Zhou received the Ph.D. degree in signal and information processing from Shanghai University, Shanghai, China, in 2018. He is currently an Associate Professor with the School of Automation, Hangzhou Dianzi University, Hangzhou, China. His research interests include saliency detection, video segmentation, image enhancement, and defect detection.

Hongyun Li received her M.S. degree from Yunnan Normal University, Kunming, China, in 2015. She is currently an Associate Professor at the Industrial School of Joint Innovation, Quanzhou Vocational and Technical University, Quanzhou, China. From 2015 to 2017, she served as a Teaching and Research Fellow at Henan Institute of Education, Zhengzhou, China. Her research interests include computer application technology, image/video restoration and segmentation, and image enhancement.