Lightweight ORSI Salient Object Detection via Frequency and Mutual Assistance Attention

Gongyang Li, *Member, IEEE*, Shixiang Shi, Yong Wu, Weisi Lin, *Fellow, IEEE*, and Zhen Bai

***Abstract*—Lightweight Salient Object Detection in Optical Remote Sensing Image (ORSI-SOD) is expected to achieve a good balance between model complexity and detection accuracy. Existing lightweight ORSI-SOD methods usually adopt the MobileNet as the backbone, which greatly reduces the model complexity, but also restricts the detection accuracy. In this paper, we propose a novel lightweight *Fre*quency and *M*utual *A*ssistance Attention *Net*work, *i.e.*, *FreMaNet*, with a lightweight transformer backbone for ORSI-SOD. Our FreMaNet is built on the strategy of intra-level modeling and inter-level assistance. Frequency-domain Self-Attention (FreSA) and Mutual Assistance Channel Attention (MaCA) are responsible for intra-level modeling and inter-level assistance, respectively. Specifically, FreSA is arranged behind the backbone to further model global relationships within each level of features (*i.e.*, intra-level features). Different from the vanilla self-attention, FreSA achieves global relationship modeling through multiplication in the frequency domain, resulting in less computational load. Then, different levels of features (*i.e.*, inter-level features) assist and interact with each other in MaCA. MaCA first performs a simple fusion on the features of two adjacent levels, and then adopts parallel self-channel attention and assistance channel attention to adaptively achieve mutual assistance of features at different levels. With the cooperation of the above components and an efficient saliency decoder, our FreMaNet has only 4.91M parameters and 4.52G floating point operations for a 352×352 input. Extensive experiments on three datasets demonstrate that our lightweight FreMaNet achieves competitive performance compared to lightweight and normal-size ORSI-SOD methods. The code and results of our method are available at https://github.com/MathLee/FreMaNet.**

***Index Terms*—Optical remote sensing image, salient object detection, frequency-domain self-attention, mutual assistance channel attention.**

I. Introduction

Salient object detection is a fundamental task in the field of computer vision [1]–[6], which can pop out potential objects of interest. Recently, researchers have shifted their attention to Optical Remote Sensing Images (ORSIs), focusing on Salient Object Detection in ORSIs (ORSI-SOD) [7]–[13].

Gongyang Li and Shixiang Shi are with the School of Communication and Information Engineering, and the Key Laboratory of Specialty Fiber Optics and Optical Access Networks, Shanghai University, Shanghai 200444, China (e-mail: ligongyang@shu.edu.cn; shishixiang@shu.edu.cn).

Yong Wu is with Bionic Vision System Laboratory, State Key Laboratory of Transducer Technology, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai 200050, China (e-mail: yongwu@mail.sim.ac.cn).

Weisi Lin is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (email: wslin@ntu.edu.sg).

Zhen Bai is with the Department of Medical Equipment, the First Affiliated Hospital of Zhengzhou University, Zhengzhou 450052, China (e-mail: bz536476@163.com).

Corresponding authors: Yong Wu and Zhen Bai.

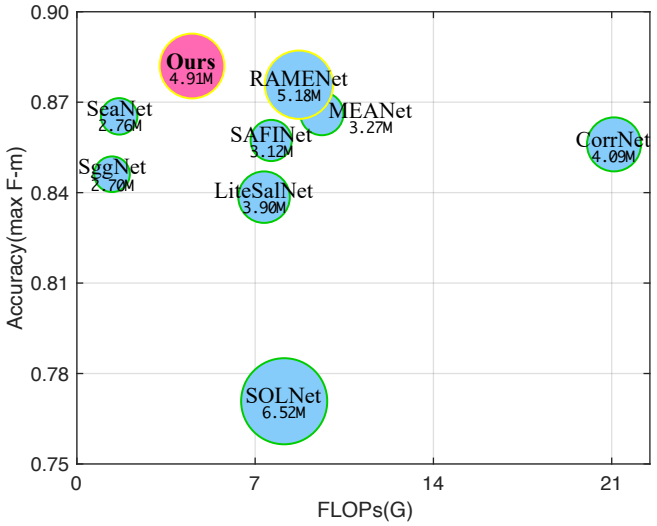

Fig. 1. The comparisons of accuracy, FLOPs, and parameter count of our FreMaNet and state-of-the-art lightweight ORSI-SOD methods [15]–[22] on the ORSI-4199 dataset [23]. ● is our FreMaNet, and ● is lightweight ORSI-SOD methods. The lightweight CNN-based method has a green edge, while the MobileViT-based method has a yellow edge. The radius of ●/● is the parameter count.

ORSI-SOD aims to pop out the attention-grabbing aircraft, ships, rivers, islands, cars, and buildings from ORSIs, showing significant value in natural resource management, post-disaster reconstruction, and urban planning [14]. Considering that the application scenarios of ORSI-SOD methods are usually on spacecraft and aircraft, researchers are concerned about the model complexity, and have proposed a series of lightweight ORSI-SOD methods [15]–[22].

At the beginning of ORSI-SOD, researchers focus on improving the detection accuracy. With the breakthrough of feature extraction backbones [24]–[27] and specifically designed architectures, numerous methods of normal size break through performance bottlenecks [10]–[13], [23], [28], [29]. Lightweight ORSI-SOD methods basically follow the same idea as normal-size ORSI-SOD methods, but use lighter backbones and design lighter architectures, significantly reducing model complexity. Most lightweight methods [15]–[21] adopt the lightweight Convolutional Neural Networks (CNNs) (such as the modified VGG [24], MobileNet [30], and RepVGG [31]) as the backbone. However, due to the limited feature extraction capabilities of lightweight CNNs, there is still significant room for improvement in the performance of these methods. RAMENet [22] breaks through this dilemma, and adopts MobileViT [32] as the backbone, improving the

detection accuracy. As shown in Fig. 1, RAMENet is the accuracy leader in lightweight methods.

These lightweight methods have multiple trade-offs in both architecture design and backbone to reduce model complexity. In the architecture design, SAFINet [18] and SOLNet [19] only explore information from single-level features. CorrNet [15], SeaNet [16], SggNet [20], LiteSalNet [21], and RAMENet [22] only consider adjacent-level relationships. MEANet [17] considers multi-level relationships, but only uses simple multiplication for lightweight fusion. However, these architecture designs create an information gap between high-level and low-level features. Moreover, we can also find from Fig. 1 that the computational load of existing lightweight ORSI-SOD methods is relatively high, with most of them exceeding 7.0G Floating Point Operations (FLOPs) except for SeaNet [16] and SggNet [20]. In more detail, we can summarize two issues with existing lightweight methods. First, although CNN-based lightweight methods [15]–[21] have relatively low model complexity, their detection accuracy is far from satisfactory. Second, although the MobileViT-based method (*i.e.*, RAMENet [22]) has high detection accuracy, it suffers from high model complexity. In other words, existing lightweight methods still have room for improvement in balancing model complexity and detection accuracy.

Motivated by the above observation, in this paper, we focus on improving the detection accuracy of CNN-based lightweight methods and reducing the high model complexity of the MobileViT-based method. With this idea, we replace the lightweight CNN backbone with MobileViT [32] and design lighter and more effective modules than RAMENet based on the strategy of intra-level modeling and inter-level assistance. As a result, we propose a novel lightweight *Fre*quency and *M*utual *A*ssistance Attention *Net*work, *i.e.*, *FreMaNet*, for ORSI-SOD with 4.91M parameters and 4.52G FLOPs. As shown in Fig. 1, our FreMaNet has higher detection accuracy than CNN-based lightweight methods [15]–[21], but with similar model complexity. It has lower model complexity but higher detection accuracy than the MobileViT-based RAMENet. Our FreMaNet achieves a better balance between model complexity and detection accuracy than existing lightweight ORSI-SOD methods.

Specifically, for the strategy of intra-level modeling and inter-level assistance, we propose Frequency-domain Self-Attention (FreSA) and Mutual Assistance Channel Attention (MaCA) in our FreMaNet. FreSA corresponds to intra-level modeling. Inspired by the Fourier transform [33], FreSA optimizes the vanilla self-attention [34] by transferring the modeling of global relationships in the spatial domain to the frequency domain. In the frequency domain, matrix multiplication in the spatial domain can be approximated by element-wise multiplication, greatly reducing computational load while maintaining good global modeling capabilities. Different from FreSA which processes single-level features, MaCA handles multi-level features simultaneously, corresponding to inter-level assistance. MaCA ingeniously achieves mutual assistance between multi-level features through parallel self-channel attention and assistance channel attention. Self-channel attention generates an attention map from the current-level features, while assistance channel attention generates an attention map from other-level features. Both attention maps modulate features together in an adaptive manner. In this way, FreSA and MaCA are extremely lightweight, *i.e.*, FreSA has 18.36K parameters and 47.93M FLOPs, while MaCA has 2.19K parameters and 0.65M FLOPs. Lightweight does not necessarily mean low detection accuracy. Our FreMaNet achieves competitive performance compared to normal-size ORSI-SOD methods.

Our main contributions are summarized as follows:

- We propose a novel lightweight ORSI-SOD method, namely *FreMaNet*, based on the strategy of intra-level modeling and inter-level assistance. FreMaNet adopts the lightweight MobileViT to improve the feature extraction ability of existing CNN-based lightweight ORSI-SOD methods. Meanwhile, it solves the issue of high model complexity of the existing MobileViT-based ORSI-SOD method. FreMaNet achieves a good balance between model complexity and detection accuracy, and provides a powerful and lightweight ORSI-SOD solution.
- We propose the Frequency-domain Self-Attention, *i.e.*, *FreSA*, to efficiently model the global relationships of intra-level features in the frequency domain. FreSA has comparable modeling capabilities to the vanilla self-attention, but it extremely reduces the computational load of vanilla self-attention from 2380.34M FLOPs to 47.93M FLOPs.
- We propose the Mutual Assistance Channel Attention, *i.e.*, *MaCA*, which not only considers the adjacent relationships, but also constructs the adaptive mutual assistance relationships. MaCA explores the mutual assistance relationships of inter-level features in the channel domain. Thereby, it has limited complexity, *i.e.*, 2.19K parameters and 0.65M FLOPs.

II. Related Work

A. Normal-size ORSI-SOD Method

Normal-size ORSI-SOD methods focus on the detection accuracy. In recent years, with the increasing attention of many researchers to ORSI-SOD, numerous high-performance normal-size ORSI-SOD methods have emerged. Here, we classify the existing normal-size ORSI-SOD methods into three categories based on the feature extraction backbone used. The first category adopts the CNN backbones, such as VGG [24], ResNet [25], and Res2Net [35]. The second category adopts the transformer backbones, such as Pyramid Vision Transformer (PVT) [27], Swin Transformer (SwinT) [26], ResT [36], and carefully designed transformer [37]. The last category adopts both CNN and transformer backbones, termed hybrid backbones. We introduce these three categories of methods one by one below.

For the first category method, LVNet [28] was the pioneer. It not only built the first dataset (*i.e.*, ORSSD dataset), but also proposed the first deep learning-based ORSI-SOD method. LVNet constructed a multi-scale feature pyramid to extract the multi-scale feature representations through multiple Multi-scale Convolution Units (M-CUs). Notably, its input was not

one ORSI, but multiple ORSIs of different sizes, which is conducive to directly extracting multi-scale information and is suitable for ORSIs containing variable sizes and multiple objects. VGG benefits from its clear structure and is a popular backbone in the first category method [12], [23], [29], [38]–[42]. Benefiting from the improved structure of ResNet, Res2Net has enabled many methods [43]–[46] to achieve performance breakthroughs. Of course, ResNet-based methods [47], [48] also occupy a part. In order to adapt to ORSIs, many effective strategies have also been applied to these methods, such as edge perception [23], [39], [40], [43], [45], [47], multi-scale input [28], [39], semantic exploration [12], [38], [44], [46], and adjacent fusion [12], [41].

Transformer [34] models the long-range dependencies, demonstrating powerful global feature extraction capability. For the second category method, GeleNet [10] and GLGCNet [11] were pioneers. Both adopted PVT [27] as the backbone, and embedded global and local feature fusion in their architectures. Following [10], [11], PRNet [49] and RoCAFENet [50] also used PVT, while PRNet employed the parallel refinement strategy and RoCAFENet developed multi-scale contextual attention. Different from the above methods, SwinT [26] was used in MTPNet [51]. MTPNet conducted SOD and edge detection together through task prompts and task tokens. TLCKDNet [52] combined the ResT encoder [36] with a large convolution kernel decoder, establishing a solution for highlighting different scales of objects. Sun *et al*. [37] did not use any existing transformers, but carefully designed a transformer backbone, named DPU-Former, which explores the global and local fusion in depth. Thanks to the powerful transformer backbones and corresponding architectures, the second category methods have significant performance improvements compared to the first category methods.

The third category method attempts to draw on the strengths of both CNN and transformer, directly extracting local and global features. For example, Wang *et al*. [53] stitched three CNN blocks with two transformer blocks, constructing a hybrid backbone for generating local and global features in sequence. This ingenious stitching broke through the limitations of both backbones and provided an all-around backbone. Differently, Zhao *et al*. [54] constructed a parallel dual-encoder structure, which uses a CNN encoder and a transformer encoder to simultaneously extract global and local features. This manner was straightforward and focused on the fusion of local and global features. The hybrid backbone injected new vitality into ORSI-SOD and achieved promising performance.

Undoubtedly, the aforementioned normal-size ORSI-SOD methods have improved detection accuracy, but they have been accompanied by high model complexity. They usually have 20M-110M parameters and 7G-480G FLOPs, restricting the practical application of existing normal-size methods. This is precisely the issue that our lightweight FreMaNet primarily addresses. Our lightweight FreMaNet adopts a lightweight backbone, develops two specialized lightweight attentions (*i.e*., FreSA and MaCA), and employs an efficient saliency decoder. In this way, our FreMaNet has only 4.91M parameters and 4.52G FLOPs, while demonstrating comparable or even superior performance to most normal-size ORSI-SOD methods.

B. Lightweight ORSI-SOD Method

As the name suggests, lightweight ORSI-SOD methods focus on the model complexity. They aim to strike a balance between detection accuracy and model complexity, facilitating the practical application of ORSI-SOD methods. Undoubtedly, this balance is a consideration for all computer vision tasks.

As pioneers, Li *et al*. [15] first concentrated on the lightweight ORSI-SOD. They first replaced regular convolutions of the last two blocks of VGG with depthwise separable convolutions, reducing the parameters of VGG from 14.72M to 3.22M. Then, they explored the adjacent-level relationships with lightweight modules, achieving a lightweight CorrNet. Subsequently, Li *et al*. [16] adopted a more lightweight backbone (*i.e*., MobileNetV2 [30]), and continuously explored the adjacent-level relationships, resulting in a more lightweight SeaNet. MobileNetV2 has been adopted by many lightweight ORSI-SOD methods [17], [18], [20], [21] due to its excellent feature extraction capability and moderate model complexity. SOLNet [19] took a different approach by using reparameterization technology to construct VGG [31], which accelerates the inference speed. However, these backbones still belong to CNNs and are not sufficient to extract representative features.

Differently, RAMENet [22] introduced a lightweight transformer (*i.e*., MobileViT [32]) into lightweight ORSI-SOD, significantly improving the performance of lightweight ORSI-SOD. Unfortunately, these methods either explore adjacent-level relationships [15], [16], [20]–[22], extract information from a single level [18], [19], or mine multi-level relationships through simple fusion [17], which is often insufficient. In addition, the computational load of most lightweight methods (including the powerful MobileViT-based RAMENet) is high.

To this end, we propose our FreMaNet to specifically address the high complexity of RAMENet (especially in terms of computational load) and the low performance of lightweight CNN-based ORSI-SOD methods. Following RAMENet [22], our FreMaNet also adopts the powerful yet lightweight MobileViT as backbone. But we embed the strategy of intra-level modeling and inter-level assistance in our FreMaNet. Concretely, we arrange two modules, *i.e*., FreSA and MaCA, with an extremely small complexity into our FreMaNet to efficiently model the global relationships of intra-level features and construct the adaptive mutual assistance relationships of inter-level features, respectively. As a result, compared to RAMENet, our FreMaNet has lower model complexity and better performance. Compared to lightweight CNN-based ORSI-SOD methods, our FreMaNet significantly improves performance while maintaining comparable model complexity.

III. Methodology

A. Network Overview

As illustrated in Fig. 2, our lightweight FreMaNet is built on the encoder-decoder architecture. It consists of a lightweight feature encoder, four FreSAs, a MaCA, and an efficient saliency decoder. Concretely, in the encoder, we adopt the lightweight yet powerful MobileViT-S [32] as the

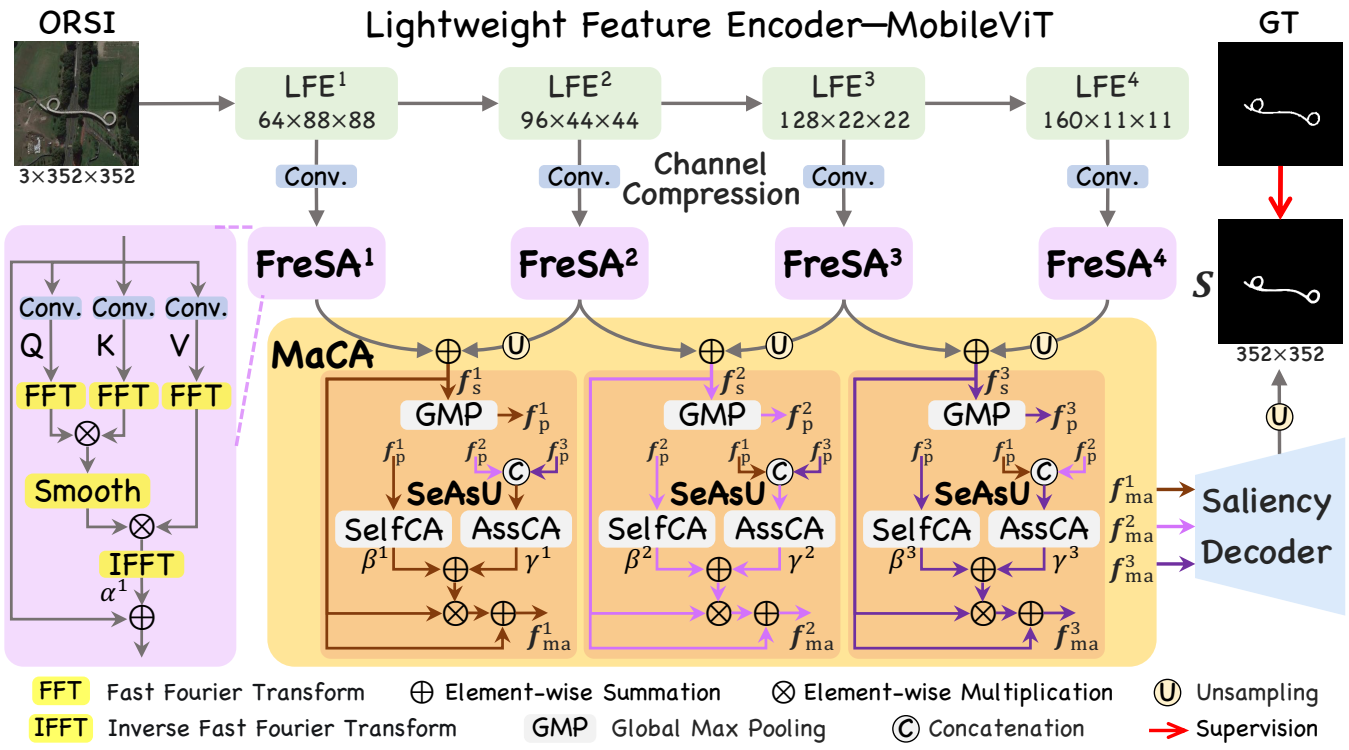

Fig. 2. The overall framework of the proposed FreMaNet. FreMaNet is built on the encoder-decoder architecture. Its input is a 3×352×352 ORSI. A lightweight feature encoder (*i.e.*, MobileViT-S [32]) is employed to extract four-level features, which are then compressed to a uniform number of channels. FreSA is arranged behind the encoder to model global relationships within each level of features. Then, MaCA receives the output of FreSA. MaCA first sums the features of adjacent levels, and then performs mutual assistance of features between different levels through three SeAsUs. Finally, an efficient saliency decoder closely follows MaCA to generate the saliency map $\boldsymbol{S}$.

backbone with a 3×352×352 ORSI as input. We denote the four stages of MobileViT-S as LFE^i $(i = 1, 2, 3, 4)$. The output basic features of LFE^i are denote as $\boldsymbol{f}_{\mathrm{b}}^i \in \mathbb{R}^{c_i \times h_i \times w_i}$, where $c_i \in \{64, 96, 128, 160\}$ and $h_i/w_i = \frac{352}{2^{i+1}}$. To achieve the goal of lightweight design and effective feature representation, we uniformly compress the number of channels of all four-level features $\{\boldsymbol{f}_{\mathrm{b}}^i\}_{i=1}^4$ to $c = 32$, obtaining $\{\boldsymbol{f}_{\mathrm{c}}^i \in \mathbb{R}^{c \times h_i \times w_i}\}_{i=1}^4$. Then, $\boldsymbol{f}_{\mathrm{b}}^i$ is fed into the corresponding FreSA^i to achieve the intra-level global relationshi modeling, generating $\boldsymbol{f}_{\mathrm{fre}}^i \in \mathbb{R}^{c \times h_i \times w_i}$. $\{\boldsymbol{f}_{\mathrm{fre}}^i\}_{i=1}^4$ are all fed into MaCA. MaCA takes adjacent relationships into account and constructs mutual assistance relationships. Therefore, MaCA adopts the summation operation to simply fuse the features of adjacent levels (*i.e.*, $\boldsymbol{f}_{\mathrm{fre}}^i$ and $\boldsymbol{f}_{\mathrm{fre}}^{i+1}$), generating $\{\boldsymbol{f}_{\mathrm{s}}^i \in \mathbb{R}^{c \times h_i \times w_i}\}_{i=1}^3$. $\{\boldsymbol{f}_{\mathrm{s}}^i\}_{i=1}^3$ are fed into the corresponding Self and Assistance Channel Attention Unit (SeAsU) to explore the mutual assistance relationships of inter-level features. SeAsU consists of a Self-Channel Attention (SelfCA) and an Assistance Channel Attention (AssCA). Three parallel SeAsUs generate $\{\boldsymbol{f}_{\mathrm{ma}}^i \in \mathbb{R}^{c \times h_i \times w_i}\}_{i=1}^3$ for saliency inference through adaptive fusion. Finally, an efficient saliency decoder receives and parses them to generate the saliency map $\boldsymbol{S} \in [0, 1]^{1 \times 352 \times 352}$.

B. Frequency-domain Self-Attention

Objects in ORSIs often exhibit significant differences in shape, size, boundary, orientation, texture, and other aspects [23], [28], [29]. Therefore, many ORSI-SOD methods [10], [15], [16] use the self-attention [34] to model the global relationships between pixels to better perceive objects in ORSIs. However, for the vanilla self-attention, its computational load and memory footprint are enormous. For example, for a feature map with an input size of $C \times W \times H$, the computational load of the vanilla self-attention is proportional to $(W \times H)^2$. In addition, it cannot run well on a regular GPU with 24GB memory. Fourier transform [33] shows us a promising way, that is, matrix multiplication in the spatial domain can be approximated by element-wise multiplication in the frequency domain. Therefore, we propose the Frequency-domain Self-Attention, which models global relationships via element-wise multiplication in the frequency domain, to replace the heavy self-attention. For a feature map with an input size of $C \times W \times H$, the computational load of our FreSA is proportional to $W \times H$, significantly reducing the computational load compared to vanilla self-attention. Our FreSA is also friendly to a regular GPU. In the following, we introduce FreSA in detail.

The structure of our FreSA is shown on the left of Fig. 2. Its input is $\boldsymbol{f}_{\mathrm{c}}^i$. Similar to self-attention [34], we first adopt convolutional layers to project $\boldsymbol{f}_{\mathrm{c}}^i$ to $\{\boldsymbol{f}_{\mathrm{q}}^i, \boldsymbol{f}_{\mathrm{k}}^i, \boldsymbol{f}_{\mathrm{v}}^i\} \in \mathbb{R}^{c \times h_i \times w_i}$, respectively. To reduce the model complexity, all convolutional layers here adopt a kernel size of only 1×1. Then, we adopt the fast Fourier transform to transform $\{\boldsymbol{f}_{\mathrm{q}}^i, \boldsymbol{f}_{\mathrm{k}}^i, \boldsymbol{f}_{\mathrm{v}}^i\}$ to the frequency domain, generating $\{\boldsymbol{f}_{\mathrm{fq}}^i, \boldsymbol{f}_{\mathrm{fk}}^i, \boldsymbol{f}_{\mathrm{fv}}^i\}$ as follows:

$$\boldsymbol{f}_{\mathrm{fq/fk/fv}}^i = \mathrm{FFT}(\boldsymbol{f}_{\mathrm{q/k/v}}^i), \tag{1}$$

where $\mathrm{FFT}(\cdot)$ is the fast Fourier transform.

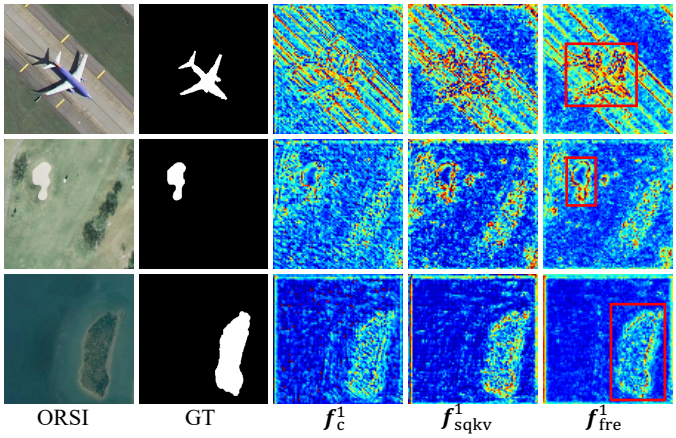

Fig. 3. Feature visualization in FreSA. Zoom-in for details.

Different from self-attention, we here generate the attention map $\boldsymbol{A}^i$ via the element-wise multiplication rather than matrix multiplication as follows:

$$\boldsymbol{A}^i = \boldsymbol{f}_{\mathrm{fq}}^i \otimes \boldsymbol{f}_{\mathrm{fk}}^i, \tag{2}$$

where $\otimes$ is the element-wise multiplication. To endow $\boldsymbol{A}^i$ with the capability of self-learning and self-adjustment, we employ the smooth layer on $\boldsymbol{A}^i$, generating $\hat{\boldsymbol{A}}^i$. Specifically, in the smooth layer, we first utilize two convolutional layers to process $\boldsymbol{A}^i$, and then apply the sigmoid function to constrain its values in $(0, 1)$. As for the attention transmission, we also adopt the element-wise multiplication to propagate $\hat{\boldsymbol{A}}^i$ to $\boldsymbol{f}_{\mathrm{fv}}^i$ as follows:

$$\boldsymbol{f}_{\mathrm{fqkv}}^i = \hat{\boldsymbol{A}}^i \otimes \boldsymbol{f}_{\mathrm{fv}}^i. \tag{3}$$

where $\boldsymbol{f}_{\mathrm{fqkv}}^i$ is the frequency-domain enhanced features.

Finally, we transform $\boldsymbol{f}_{\mathrm{fqkv}}^i$ to the spatial domain as follows:

$$\boldsymbol{f}_{\mathrm{sqkv}}^i = \mathrm{IFFT}(\boldsymbol{f}_{\mathrm{fqkv}}^i), \tag{4}$$

where $\boldsymbol{f}_{\mathrm{sqkv}}^i$ is the spatial-domain enhanced features and $\mathrm{IFFT}(\cdot)$ is the inverse fast Fourier transform. We also fuse $\boldsymbol{f}_{\mathrm{sqkv}}^i$ with the input $\boldsymbol{f}_{\mathrm{c}}^i$ through a residual connection with a learnable weight $\alpha^i \in (0, 1)$, obtaining the output features of FreSA, *i.e.*, $\boldsymbol{f}_{\mathrm{fre}}^i$.

Our FreSA implements the core attention generation (*i.e.*, Eq. 2) and attention propagation (*i.e.*, Eq. 3) via element-wise multiplication. Multiplying the amplitude spectrums and multiplying the phase spectrums in the frequency domain corresponds to a global modeling operation in the spatial domain, enabling more efficient modeling of global spatial relationships than vanilla self-attention. Therefore, our FreSA can serve as an efficient alternative to vanilla self-attention, and is highly friendly to lightweight ORSI-SOD. To intuitively demonstrate the effect of our FreSA, we visualize $\boldsymbol{f}_{\mathrm{c}}^1$, $\boldsymbol{f}_{\mathrm{sqkv}}^1$, and $\boldsymbol{f}_{\mathrm{fre}}^1$ in Fig. 3. As observed, the salient regions in $\boldsymbol{f}_{\mathrm{sqkv}}^1$ are more obvious than those in the input features $\boldsymbol{f}_{\mathrm{c}}^1$, highlighting salient regions in the output features $\boldsymbol{f}_{\mathrm{fre}}^1$ (marked by the red box).

C. Mutual Assistance Channel Attention

Through the modeling of global information using a lightweight transformer backbone and FreSA, $\boldsymbol{f}_{\mathrm{fre}}^i$ is rich in global information, but this information is limited to a single level. For ORSI-SOD, the interaction of features across different levels is highly necessary and is beneficial for performance improvement [10], [16], [20]–[22], [41]. This is because features at different levels characterize objects from different granularities. Thus, feature interaction can be interpreted as granularity interaction. For ORSIs that inherently contain objects with diverse sizes, categories, shapes, and orientations, granularity interaction is conducive to perceiving such variable objects, thus enabling good adaptation to ORSIs. Therefore, we propose the Mutual Assistance Channel Attention to handle multi-level features simultaneously and collaborate with FreSA. The structure of MaCA is illustrated at the bottom of Fig. 2. Its inputs are $\{\boldsymbol{f}_{\mathrm{fre}}^i\}_{i=1}^4$. MaCA achieves granularity interaction by considering the adjacent relationships and constructing the mutual assistance relationships. In the following, we introduce MaCA from two components, including the adjacent feature fusion and the self and assistance channel attention unit.

1) Adjacent Feature Fusion. Adjacent feature fusion can effectively integrate information of similar granularities without impairing the representation of objects due to excessive granularity differences. Here, we adopt the simple summation operation to fuse the adjacent features $\boldsymbol{f}_{\mathrm{fre}}^i$ and $\boldsymbol{f}_{\mathrm{fre}}^{i+1}$, generating $\boldsymbol{f}_{\mathrm{s}}^i$ as follows:

$$\boldsymbol{f}_{\mathrm{s}}^i = \boldsymbol{f}_{\mathrm{fre}}^i \oplus \mathrm{Up}_2(\boldsymbol{f}_{\mathrm{fre}}^{i+1}), \tag{5}$$

where $\oplus$ is the element-wise summation and $\mathrm{Up}_2(\cdot)$ is the $2\times$ upsampling operation.

2) Self and Assistance Channel Attention Unit. SeAsU is arranged behind the adjacent feature fusion. It achieves complementary enhancement by leveraging the mutual assistance of $\boldsymbol{f}_{\mathrm{s}}^1$, $\boldsymbol{f}_{\mathrm{s}}^2$, and $\boldsymbol{f}_{\mathrm{s}}^3$. Concretely, SeAsU consists of a self-channel attention and an assistance channel attention. We employ SeAsUs to process features at three levels simultaneously. In each SeAsU, we first perform the global max pooling layer on $\boldsymbol{f}_{\mathrm{s}}^1$, $\boldsymbol{f}_{\mathrm{s}}^2$, and $\boldsymbol{f}_{\mathrm{s}}^3$ to generate $\boldsymbol{f}_{\mathrm{p}}^1 \in \mathbb{R}^{c\times1\times1}$, $\boldsymbol{f}_{\mathrm{p}}^2 \in \mathbb{R}^{c\times1\times1}$, and $\boldsymbol{f}_{\mathrm{p}}^3 \in \mathbb{R}^{c\times1\times1}$, respectively.

Take the SeAsU for $\boldsymbol{f}_{\mathrm{p}}^1$ for an example, we adopt the SelfCA to extract the self channel attention map $\boldsymbol{A}_{\mathrm{self}}^1 \in \mathbb{R}^{c\times1\times1}$ from $\boldsymbol{f}_{\mathrm{p}}^1$ as follows:

$$\boldsymbol{A}_{\mathrm{self}}^1 = \mathrm{SelfCA}(\boldsymbol{f}_{\mathrm{p}}^1). \tag{6}$$

Meanwhile, we adopt the AssCA to extract the assistance channel attention map $\boldsymbol{A}_{\mathrm{ass}}^1 \in \mathbb{R}^{c\times1\times1}$ from $\boldsymbol{f}_{\mathrm{p}}^2$ and $\boldsymbol{f}_{\mathrm{p}}^3$ as follows:

$$\boldsymbol{A}_{\mathrm{ass}}^1 = \mathrm{AssCA}\left(\mathrm{Cat}(\boldsymbol{f}_{\mathrm{p}}^2, \boldsymbol{f}_{\mathrm{p}}^3)\right), \tag{7}$$

where $\mathrm{Cat}(\cdot)$ is the channel concatenation. Notably, the essence of SelfCA and AssCA is the channel attention [55]. They are the same, both consisting of two convolutional layers and a sigmoid function.

Then, to achieve effective and adaptive fusion, we assign two learnable weights $\beta^1 \in (0, 1)$ and $\gamma^1 \in (0, 1)$ to $\boldsymbol{A}_{\mathrm{self}}^1$

and $\boldsymbol{A}^1_{\text{ass}}$, respectively, and sum them to generate the joint channel attention map $\boldsymbol{A}^1_{\text{ca}} \in \mathbb{R}^{c\times1\times1}$ as follows:

$$\boldsymbol{A}^1_{\text{ca}} = \beta^1 \cdot \boldsymbol{A}^1_{\text{self}} + \gamma^1 \cdot \boldsymbol{A}^1_{\text{ass}}, \tag{8}$$

where we constrain the sum of β^1 and γ^1 to 1. $\boldsymbol{A}^1_{\text{ca}}$ is the core of mutual assistance.

Finally, we adopt $\boldsymbol{A}^1_{\text{ca}}$ to modulate $\boldsymbol{f}^1_{\text{s}}$, and then superimpose the modulated features with $\boldsymbol{f}^1_{\text{s}}$ via the residual connection, generating $\boldsymbol{f}^1_{\text{ma}}$. Following the same operations, we can obtain $\boldsymbol{f}^2_{\text{ma}}$ and $\boldsymbol{f}^3_{\text{ma}}$, as shown in Fig. 2. In this way, MaCA achieves the mutual assistance through adaptive fusion of two channel attention maps, which is flexible. Moreover, since MaCA is based on the channel attention and we have compressed the channel number of features, the parameters and computational load of MaCA are extremely limited.

D. Efficient Saliency Decoder and Hybrid Loss Function

Inspired by [10], we adopt the partial decoder [56] as our saliency decoder. The inputs of our saliency decoder are $\boldsymbol{f}^1_{\text{ma}} \in \mathbb{R}^{c\times88\times88}$, $\boldsymbol{f}^2_{\text{ma}} \in \mathbb{R}^{c\times44\times44}$, and $\boldsymbol{f}^3_{\text{ma}} \in \mathbb{R}^{c\times22\times22}$, which perfectly match the input sizes of the original partial decoder. Therefore, we do not make any structure modifications to the original partial decoder. We only replace the regular convolution layers of the original partial decoder with depthwise separable convolution layers for the sake of high efficiency, obtaining an efficient saliency decoder as shown in Fig. 2. By virtue of this decoder, we can parse salient objects and generate the saliency map $\boldsymbol{S} \in [0,1]^{1\times352\times352}$ using the simple $4\times$ upsampling operation.

To effectively train our FreMaNet, we develop a hybrid loss function in the training phase. Our hybrid loss function consists of three losses, including the weighted Binary Cross-Entropy (wBCE) loss, the weighted Intersection-over-Union (wIoU) loss, and the F-measure (Fm) loss [57]. They are pixel-aware, patch-aware, and measure-aware, respectively. We formulate our hybrid loss function L_{total} as follows:

$$L_{\text{total}} = \underbrace{L_{\text{wBCE}}(\boldsymbol{S},\boldsymbol{G})}_{\text{Pixel-aware}} + \underbrace{L_{\text{wIoU}}(\boldsymbol{S},\boldsymbol{G})}_{\text{Patch-aware}} + \underbrace{L_{\text{Fm}}(\boldsymbol{S},\boldsymbol{G})}_{\text{Measure-aware}}, \tag{9}$$

where $L_{\text{wbce}}(\cdot)$, $L_{\text{wiou}}(\cdot)$, and $L_{\text{fm}}(\cdot)$ correspond to wBCE loss, wIoU loss, and Fm loss, respectively, and $\boldsymbol{G} \in \{0,1\}^{1\times352\times352}$ is the Ground Truth (GT).

IV. Experiments

A. Experimental Setup

1) Datasets: We conduct comprehensive experiments on three ORSI-SOD datasets [23], [28], [29], which vary in dataset size and thus allow for a robust evaluation of ORSI-SOD methods. The ORSSD dataset [28][1] consists of 800 pairs of ORSI and GT, dividing the training set and testing set in a 3:1 ratio. The EORSSD dataset [29][2] is an extended dataset of the ORSSD dataset, containing 2,000 pairs of ORSI and GT which are split into training and testing sets at a ratio of 7:3. The ORSI-4199 dataset [23][3] consists of 4,199 pairs of ORSI and GT, making it the largest one among the three datasets. Among these pairs, 2,000 are used as the training set, and the remaining 2,199 serve as the test set. Notably, in the field of ORSI-SOD, these three datasets are evaluated separately rather than jointly.

2) Implementation Details: We achieve our FreMaNet using PyTorch [58] and an NVIDIA RTX 3090 GPU (24GB memory). All images are first resized to 352×352 for network input. We adopt rotation and flipping for data augmentation. In the training phase, the lightweight feature encoder of our FreMaNet is initialized using pre-trained parameters, while other layers are initialized using the Kaiming method [59]. Our FreMaNet is optimized by the AdamW optimizer for 100 epochs. The initial learning rate is set to $2e^{-4}$, which decreases to 10% of the original rate every 40 epochs. The batch size is set to 16, which results in a GPU memory occupancy of 10.41 GB during training.

3) Evaluation Metrics: We evaluate not only the detection accuracy but also the model complexity of all methods. For the detection accuracy, we adopt S-measure (S_α, α = 0.5) [60], maximum F-measure ($F^{\max}_\beta$, β^2 = 0.3) [61], maximum E-measure ($E^{\max}_\xi$) [62], and Mean Absolute Error (MAE, $\mathcal{M}$) from the evaluation tool[4] for assessment. The larger the values of the first three metrics, the better. The smaller the value of the last one, the better. For the model complexity, we adopt the parameter count, the computational load, and the inference speed (without I/O time) for assessment. The smaller the value of the first two metrics, the better. The larger the values of the last one, the better.

B. Comparison with State-of-the-arts

We comprehensively compare our FreMaNet with 32 state-of-the-art ORSI-SOD methods on three widely used datasets, including EORSSD, ORSSD, and ORSI-4199. In particular, 24 of these comparison methods are normal-size ORSI-SOD methods, such as LVNet [28], DAFNet [29], SARNet [38], MJRBM [23], EMFINet [39], MCCNet [40], HFANet [53], ERPNet [47], ACCoNet [41], GeleNet [10], GLGCNet [11], MIRGNet [42], TSCNet [12], RAGRNet [43], SFANet [44], TLCKDNet [52], PRNet [49], ADSTNet [54], BCARNet [48], MRBINet [45], RoCAFENet [50], MTPNet [51], DPU-Former [37], and SAANet [46], and the remaining 8 methods are lightweight ORSI-SOD methods, such as CorrNet [15], SeaNet [16], MEANet [17], SAFINet [18], SOLNet [19], SggNet [20], LiteSalNet [21], and RAMENet [22]. We obtain the saliency maps of the aforementioned methods through the available links and the source codes made public by the authors.

1) Quantitative and Model Complexity Comparison: We provide the quantitative and model complexity comparison results of our FreMaNet and other 32 state-of-the-art ORSI-SOD methods on three datasets in Tab. I. As shown in the upper part of Tab. I, our lightweight FreMaNet exhibits competitive performance compared to normal-sized ORSI-SOD

[1]https://li-chongyi.github.io/proj_optical_saliency.html
[2]https://github.com/rmcong/DAFNet_TIP20
[3]https://github.com/wchao1213/ORSI-SOD
[4]https://github.com/MathLee/MatlabEvaluationTools

TABLE I

QUANTITATIVE COMPARISONS ON PERFORMANCE AND MODEL COMPLEXITY WITH STATE-OF-THE-ART NORMAL-SIZE AND LIGHTWEIGHT ORSI-SOD METHODS ON EORSSD, ORSSD, AND ORSI-4199 DATASETS. ↓ INDICATES THAT THE SMALLER THE BETTER, WHILE ↑ IS THE OPPOSITE. WE HIGHLIGHT THE BEST RESULT AND THE SECOND BEST RESULT IN **RED** AND **BLUE**, RESPECTIVELY.

Methods	Backbone	Input size	Param↓ (M)	FLOPs↓ (G)	Speed↑ (fps)	EORSSD [29]				ORSSD [28]				ORSI-4199 [23]			
						$S_\alpha\uparrow$	$F_\beta^{\max}\uparrow$	$E_\xi^{\max}\uparrow$	$\mathcal{M}\downarrow$	$S_\alpha\uparrow$	$F_\beta^{\max}\uparrow$	$E_\xi^{\max}\uparrow$	$\mathcal{M}\downarrow$	$S_\alpha\uparrow$	$F_\beta^{\max}\uparrow$	$E_\xi^{\max}\uparrow$	$\mathcal{M}\downarrow$
Normal-size ORSI-SOD Methods																	
$LVNet_{19}$ [28]	M-CU	128^2	207	-	1.4	.8630	.7794	.9254	.0146	.8815	.8263	.9456	.0207	-	-	-	-
$DAFNet_{21}$ [29]	VGG	128^2	29.35	68.5	26	.9166	.8614	**.9861**	.0060	.9191	.8928	.9771	.0113	-	-	-	-
$SARNet_{21}$ [38]	VGG	336^2	25.91	129.7	47	.9240	.8719	.9620	.0099	.9134	.8850	.9557	.0187	-	-	-	-
$MJRBM_{22}$ [23]	VGG	352^2	43.54	95.7	32	.9197	.8656	.9646	.0099	.9204	.8842	.9623	.0163	.8593	.8493	.9311	.0374
$EMFINet_{22}$ [39]	VGG	256^2	107.26	480.9	25	.9290	.8720	.9711	.0084	.9366	.9002	.9737	.0109	.8675	.8584	.9340	.0330
$MCCNet_{22}$ [40]	VGG	256^2	67.65	112.8	95	.9327	.8904	.9755	.0066	.9437	.9155	.9800	.0087	.8746	.8690	.9413	.0316
$HFANet_{22}$ [53]	ResNet+PVT	448^2	60.53	68.3	26	.9380	.8876	.9740	.0070	.9399	.9112	.9770	.0092	.8767	.8700	.9431	.0314
$ERPNet_{23}$ [47]	ResNet	224^2	56.48	87.2	50	.9210	.8632	.9603	.0089	.9254	.8974	.9710	.0135	.8670	.8553	.9290	.0357
$ACCoNet_{23}$ [41]	VGG	256^2	102.55	179.95	81	.9290	.8837	.9727	.0074	.9437	.9149	.9796	.0088	.8775	.8686	.9412	.0314
$GeleNet_{23}$ [10]	PVT	352^2	25.45	11.66	30	.9376	.8923	.9828	.0064	.9469	.9254	.9860	.0079	**.8862**	.8842	.9544	**.0264**
$GLGCNet_{23}$ [11]	PVT	352^2	25.15	9.8	21	.9375	.8924	.9803	.0055	.9488	.9236	.9864	.0071	.8839	.8808	.9508	.0274
$MIRGNet_{24}$ [42]	VGG	256^2	78.72	136.2	46.9	.9383	**.8930**	.9789	.0056	.9455	.9192	.9812	.0081	-	-	-	-
$TSCNet_{24}$ [12]	VGG	256^2	103.56	116.82	59	.9376	.8900	.9765	.0061	.9428	.9198	.9850	.0081	.8783	.8771	.9486	.0295
$RAGRNet_{24}$ [43]	Res2Net	256^2	35.6	17.8	31	.9361	.8852	.9785	.0057	**.9507**	.9242	.9861	**.0066**	.8811	.8811	.9492	.0284
$SFANet_{24}$ [44]	Res2Net	256^2	25.1	7.7	-	.9349	.8833	.9769	.0058	.9453	.9192	.9830	.0077	.8761	.8710	.9447	.0292
$TLCKDNet_{24}$ [52]	ResT	256^2	50	-	32	.9350	.8843	.9788	.0056	.9421	.9114	.9794	.0082	-	-	-	-
$PRNet_{24}$ [49]	PVT	352^2	20.8	8.47	21	.9276	.8684	.9784	.0054	.9459	.9177	.9848	.0075	**.8873**	.8819	.9527	.0272
$ADSTNet_{24}$ [54]	Res2Net+T	256^2	62.09	27.72	40	.9311	.8804	.9769	.0065	.9379	.9124	.9807	.0086	.8710	.8698	.9433	.0318
$BCARNet_{25}$ [48]	ResNet	352^2	24	7	-	.9361	.8871	.9761	.0054	.9465	.9196	.9833	.0071	.8757	.8689	.9407	.0306
$MRBINet_{25}$ [45]	Res2Net	256^2	32.4	42.8	9.0	.9351	.8852	.9766	.0056	.9474	.9199	.9851	.0069	.8824	.8800	.9489	.0268
$RoCAFENet_{25}$ [50]	PVT	352^2	31.66	24.15	19.96	**.9437**	**.8983**	**.9844**	**.0053**	.9497	**.9272**	**.9873**	.0074	.8847	.8845	.9465	.0267
$MTPNet_{25}$ [51]	SwinT	352^2	55.0	-	91	.9385	**.8930**	.9808	**.0050**	**.9538**	**.9330**	**.9916**	**.0066**	**.8862**	**.8873**	**.9553**	.0268
$DPU\text{-}Former_{25}$ [37]	DPU-Former	352^2	44.20	32.51	-	**.9401**	**.8930**	.9816	.0056	.9412	.9263	.9868	**.0062**	.8833	**.8877**	**.9547**	**.0263**
$SAANet_{25}$ [46]	Res2Net	352^2	43.06	43.31	17	.9397	.8918	.9801	.0057	.9483	.9226	.9840	.0078	.8806	.8773	.9455	.0302
Lightweight ORSI-SOD Methods																	
$CorrNet_{22}$ [15]	LFE-VGG	256^2	4.09	21.09	100	.9289	.8778	.9696	.0083	.9380	.9129	.9790	.0098	.8623	.8560	.9330	.0366
$SeaNet_{23}$ [16]	MobileNetV2	288^2	2.76	1.66	96	.9208	.8649	.9710	.0073	.9260	.8942	.9767	.0105	.8772	.8653	.9426	.0308
$MEANet_{24}$ [17]	MobileNetV2	352^2	3.27	9.62	33	.9282	.8766	.9708	.0070	.9340	.9033	.9768	.0098	.8708	.8662	.9417	.0312
$SAFINet_{24}$ [18]	MobileNetV2	288^2	3.12	7.63	-	.9267	.8799	.9732	.0065	.9401	.9106	.9786	.0086	.8583	.8573	.9367	.0340
$SOLNet_{25}$ [19]	RepVGG-A0	256^2	6.52	8.14	161	.9171	.8609	.9623	.0078	.9284	.9012	.9734	.0111	.7901	.7708	.8848	.0592
$SggNet_{25}$ [20]	MobileNetV2	288^2	2.70	1.38	108	.9278	.8871	.9762	.0068	.9342	.9030	.9758	.0111	.8563	.8461	.9337	.0351
$LiteSalNet_{25}$ [21]	MobileNetV2	256^2	3.90	7.35	35	.9273	.8877	.9743	.0063	.9372	.9124	.9789	.0090	.8521	.8385	.9273	.0383
$RAMENet_{25}$ [22]	MobileViT-S	352^2	5.18	8.72	92	**.9419**	**.8971**	**.9822**	**.0050**	**.9489**	**.9229**	**.9862**	**.0072**	**.8776**	**.8758**	**.9503**	**.0277**
FreMaNet (Ours)	MobileViT-S	352^2	4.91	4.52	208	**.9428**	**.8962**	**.9844**	**.0048**	**.9496**	**.9226**	**.9858**	**.0067**	**.8810**	**.8820**	**.9538**	**.0270**

**We record Param, FLOPs, and Speed from the original published papers, so the precision of the decimal point is different.

TABLE II

QUANTITATIVE COMPARISONS ACROSS DIFFERENT OBJECT SCALES (SMALL, MEDIUM, LARGE) OF THE ORSI-4199 DATASET.

Methods	ORSI-4199-Small (1693)				ORSI-4199-Medium (279)				ORSI-4199-Large (227)			
	$S_\alpha\uparrow$	$F_\beta^{\max}\uparrow$	$E_\xi^{\max}\uparrow$	$\mathcal{M}\downarrow$	$S_\alpha\uparrow$	$F_\beta^{\max}\uparrow$	$E_\xi^{\max}\uparrow$	$\mathcal{M}\downarrow$	$S_\alpha\uparrow$	$F_\beta^{\max}\uparrow$	$E_\xi^{\max}\uparrow$	$\mathcal{M}\downarrow$
Lightweight ORSI-SOD Methods												
$CorrNet_{22}$ [15]	.8751	.8585	.9469	.0210	.8502	.9001	.8971	.0837	.7814	.7940	.8837	.0943
$SeaNet_{23}$ [16]	.8796	.8637	.9523	.0190	.8725	.9176	.9126	.0717	.8166	.8178	.9152	.0688
$MEANet_{24}$ [17]	.8777	.8636	.9505	.0198	.8702	.9147	.9094	.0720	.8198	**.8274**	.9200	.0657
$SAFINet_{24}$ [18]	.8630	.8530	.9436	.0228	.8672	.9144	.9114	.0749	.8131	.8202	.9190	.0679
$SOLNet_{25}$ [19]	.8018	.7688	.9010	.0383	.7768	.8394	.8470	.1285	.7186	.7396	.8390	.1294
$SggNet_{25}$ [20]	.8614	.8412	.9420	.0231	.8613	.9055	.9021	.0793	.8123	.8143	.9156	.0695
$LiteSalNet_{25}$ [21]	.8537	.8292	.9326	.0279	.8711	.9134	.9097	.0761	.8164	.8190	.9145	.0695
$RAMENet_{25}$ [22]	**.8803**	**.8728**	**.9568**	**.0193**	**.8951**	**.9354**	**.9327**	**.0576**	**.8364**	**.8272**	**.9253**	**.0541**
FreMaNet (Ours)	**.8866**	**.8816**	**.9608**	**.0176**	**8914**	**.9359**	**.9346**	**.0583**	**.8263**	.8229	**.9298**	**.0586**

methods. As for model complexity, our FreMaNet outperforms all normal-sized methods comprehensively in terms of the parameter count, the computational load, and the inference speed. Specifically, on the EORSSD dataset, our FreMaNet performs second only to RoCAFENet [50] and surpasses all 24 normal-sized ORSI-SOD methods in terms of the MAE metric, which is merely 0.0048. On the ORSSD dataset, our FreMaNet is comparable to GeleNet [10], BCARNet [48], and MRBINet [45]. However, the parameter count of our FreMaNet is only about 20% of that of GeleNet and BCARNet, and its computational load is only about 10% of that of MRBINet. On the ORSI-4199 dataset, our FreMaNet is comparable to TSCNet [12] and RAGRNet [43]. However, the parameter count of our FreMaNet is only about 5% of that of TSCNet, and its inference speed is 6× faster than that of RAGRNet.

As shown in the bottom part of Tab. I, our lightweight FreMaNet ranks first in 9 out of 12 quantitative metrics and is second only to RAMENet [22] in the remaining 3 metrics. As for model complexity, the parameter count of our FreMaNet is inferior due to the large parameter count in its backbone. But our FreMaNet leads in terms of computational load and inference speed, especially ranking first in the inference speed (*i.e.*, 208 fps). Overall, our FreMaNet achieves the predefined objectives. It has lower model complexity yet better performance than RAMENet, and superior performance while maintaining a comparable parameter count to lightweight CNN-based ORSI-SOD methods. Therefore, our FreMaNet achieves a better balance between detection accuracy and model complexity.

In addition, we report quantitative comparisons across dif-

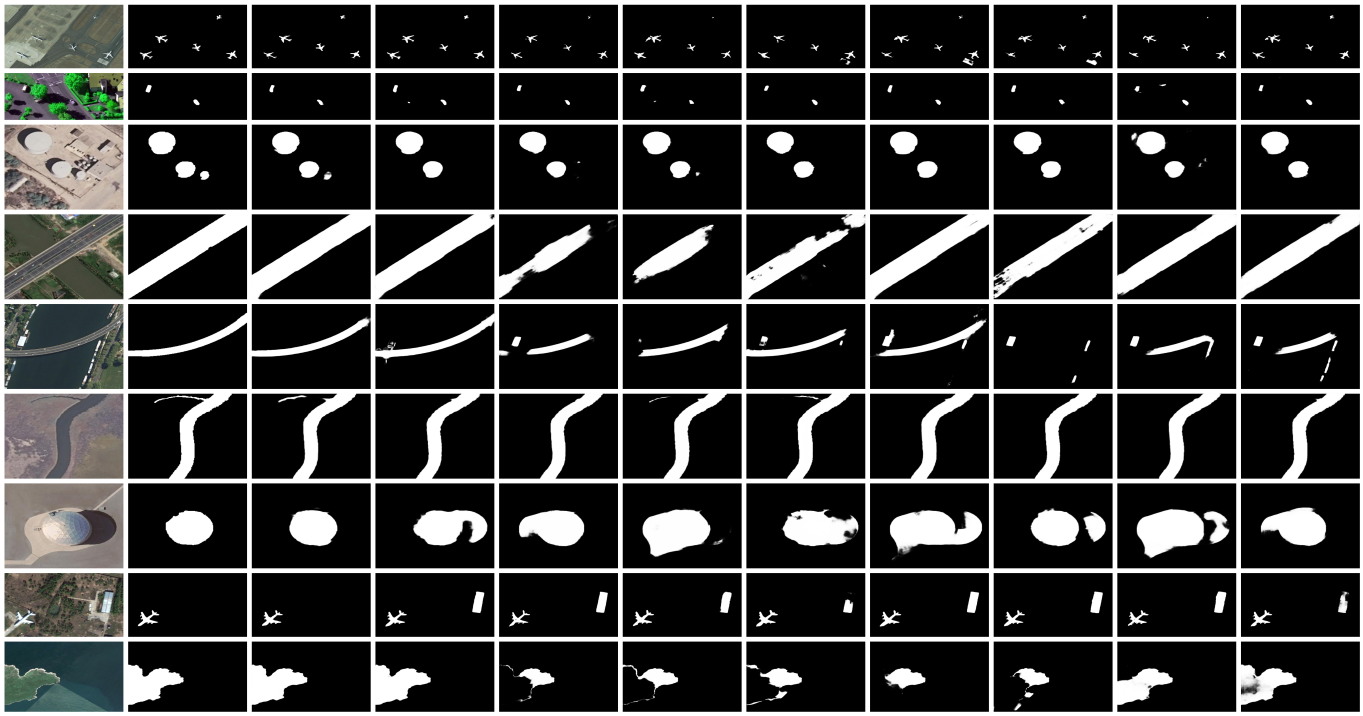

Fig. 4. Visual comparisons with eight lightweight ORSI-SOD methods.

TABLE III
ABLATION STUDIES ON EVALUATING THE INDIVIDUAL CONTRIBUTION OF EACH ATTENTION IN FREMANET. THE BEST ONE IN EACH COLUMN IS **BOLD**.

No.	Models	Param (K)	FLOPs (M)	EORSSD [29]			
				$S_\alpha \uparrow$	$F_\beta^{\max} \uparrow$	$E_\xi^{\max} \uparrow$	$\mathcal{M} \downarrow$
1	Base (*w/ MobileViT-S*)	4891.13	4475.57	0.9362	0.8867	0.9775	0.0062
2	Base+FreSA	4909.49 **+18.36**	4523.50 **+47.93**	0.9400	0.8936	0.9828	0.0050
3	Base+MaCA	4893.32 **+2.19**	4476.22 **+0.65**	0.9390	0.8908	0.9820	0.0051
4	Base* (*w/ MobileNetV2*)	2268.74	1813.04	0.9115	0.8549	0.9563	0.0101
5	Base+vanilla self-attention [34]	4896.42 **+5.39**	6855.91 **+2380.34**	0.9401	0.8939	0.9810	**0.0048**
6	Base+FreSA+MaCA (Ours)	4911.68 **+20.55**	4524.15 **+48.58**	**0.9428**	**0.8962**	**0.9844**	**0.0048**

ferent object scales (*e.g.*, small, medium, and large) of the ORSI-4199 dataset in Tab. II. As described in the ORSI-4199 dataset [23], the small object is that the ratio of the salient object to the entire image is not larger than 0.3. The medium object is that the ratio of the salient object to the entire image is between 0.3 and 0.5. The large object is that the ratio of the salient object to the entire image is not less than 0.5. The ORSI-4199 dataset contains 1693 ORSIs with small objects, 279 ORSIs with medium objects, and 227 ORSIs with large objects. As is well known, small objects are a unique characteristic of ORSIs. As reported in Tab. II, our FreMaNet achieves the best performance on small objects among all lightweight methods, demonstrating good adaptability to ORSIs. Moreover, the performance of our FreMaNet on medium and large objects is also competitive. The performance on three different scale object synthetically creates the superiority of our FreMaNet among all lightweight methods on the ORSI-4199 dataset as reported in Tab. I.

2) Visual Comparison: We visually compare the saliency maps generated by our FreMaNet with those generated by eight lightweight ORSI-SOD methods. In Fig. 4, we show nine cases of three challenging ORSI scenes, including multiple objects with different sizes and different object types (*i.e.*, the first three cases), multiple orientations with straight lines and arcs (*i.e.*, the middle three cases), and boundaries with complex geometric structures (*i.e.*, the last three cases). For the first scene, some methods, such as RAMENet, SggNet, and SeaNet, tend to produce false negatives (*i.e.*, missed detection) and false positives (*i.e.*, wrong detection), while our FreMaNet successfully highlights all objects. For the second scene, since objects span the entire ORSI with different orientations, some methods, such as LiteSalNet, SOLNet, and CorrNet, fail to segment the full extent of objects, while our FreMaNet fully perceives objects with different orientations. For the last scene, all eight lightweight ORSI-SOD methods fail to yield satisfactory saliency maps, as they are disturbed

by shadows and similar backgrounds. In contrast, the saliency maps generated by our FreMaNet are not only complete but also possess fine boundaries. In summary, with the strategy of intra-level modeling and inter-level assistance, our FreMaNet can produce promising saliency maps.

C. Ablation Studies

We conduct comprehensive ablation studies on each component of our FreMaNet using the same hyperparameters as those in the experimental setup on the EORSSD dataset to evaluate their contributions. Specifically, we assess our FreMaNet from the following three aspects, including 1) the individual contribution of each attention in FreMaNet, 2) the rationality of SelfCA and AssCA in SeAsU of MaCA, and 3) the effectiveness of the hybrid loss function.

1) The Individual Contribution of Each Attention in FreMaNet: FreSA and MaCA are the core of our FreMaNet. We conduct a detailed evaluation of their individual contribution. We provide the quantitative results and the model complexity in Tab. III. For Base (*w/ MobileViT-S*) in Tab. III, we keep the MobileViT-S, channel compression, and the saliency decoder, directly remove four FreSAs and a MaCA, sum the adjacent $\boldsymbol{f}_{\mathrm{c}}^{i}$ and $\boldsymbol{f}_{\mathrm{c}}^{i+1}$, and input them into the saliency decoder.

By comparing No. 1 and No. 6 in Tab. III, we find that FreSA and MaCA only add 20.55K parameters and 48.58M FLOPs of computational load to Base, while improving $F_{\beta}^{\max}$ by nearly 1%. On the other hand, due to the large number of parameters inherent in Base, our FreMaNet cannot further reduce the parameter count to the level of lightweight CNN-based ORSI-SOD methods [16]–[18], [20], [21], which is only 2-4M. Embedding FreSA or MaCA alone into Base does improve the model performance while only increasing the model complexity minimally. Specifically, the four-level FreSAs only have 18.36K parameters and 47.93M FLOPs. Since MaCA operates at the channel domain, it introduces even less complexity, with an increase of merely 2.19K parameters and 0.65M FLOPs.

We also provide a variant that replaces the backbone of Base from MobileViT-S to MobileNetV2, denoted by Base* (*w/ MobileNetV2*). As reported in No.4 of Tab. III, switching the backbone from MobileNetV2 to MobileViT-S greatly improves the performance, which is even more obvious than the improvement of two modules. The reason behind this is that MobileViT-S combines the local feature extraction ability of CNN with the global modeling ability of the transformer. This local and global hybrid feature representation is exactly what is urgently needed for SOD tasks [22], while MobileNetV2 only provides the local features. Even though Base (*w/ MobileViT-S*) has achieved a good performance, our FreSA and MaCA still improve its performance, which further proves the strength of our modules.

In addition, we provide a variant that replaces our FreSA with the vanilla self-attention [34], as reported in No.5 of Tab. III. Our FreSA can run smoothly with a batch size of 16 under 24 GB of GPU memory, while the vanilla self-attention will encounter an out-of-memory error with a batch size of 16. Therefore, we reduce the batch size of No.5 to 8 to enable training. In terms of performance, our FreSA is slightly inferior to the vanilla self-attention. For example, it is only 0.0001 lower in S_{α}, 0.0003 lower in $F_{\beta}^{\max}$, and 0.0002 lower in $\mathcal{M}$, while it is slightly higher by 0.0018 in $E_{\xi}^{\max}$. However, our FreSA drastically reduces the computational load of the vanilla self-attention from 2380.34M FLOPs to 47.93M FLOPs, and requires less GPU memory for running, which indicates that our FreSA is more device-friendly. By the way, in our complete model (No. 6), the learnable weights α^1, α^2, α^3, and α^4 in four FreSAs are 0.4674, 0.4494, 0.4414, and 0.4431, respectively.

TABLE IV
ABLATION STUDIES ON EVALUATING THE RATIONALITY OF SELFCA AND ASSCA IN SEASU OF MACA. THE BEST ONE IN EACH COLUMN IS **BOLD**.

No.	SelfCA	AssCA	EORSSD [29]			
			$S_{\alpha}\uparrow$	$F_{\beta}^{\max}\uparrow$	$E_{\xi}^{\max}\uparrow$	$\mathcal{M}\downarrow$
1	0.5	0.5	**0.9392**	0.8907	0.9800	0.0054
2	1	-	0.9384	0.8896	0.9795	0.0053
3	-	1	0.9381	0.8907	0.9814	0.0057
4	*Concat&CA*		0.9384	**0.8910**	0.9811	0.0053
5	β	γ	0.9390	0.8908	**0.9820**	**0.0051**

2) The Rationality of SelfCA and AssCA in SeAsU of MaCA: SeAsU is the core of our MaCA, while the parallel SelfCA and AssCA are the core of SeAsU. To investigate the rationality of SelfCA and AssCA in SeAsU of MaCA, we provide four variants: 1) modifying the adaptive weights $\{\beta, \gamma\}$ to the fixed weights $\{0.5, 0.5\}$, 2) removing the AssCA and only keeping SelfCA, 3) removing the SelfCA and only keeping AssCA, and 4) merging two attention branches into one attention branch and directly concatenating $\{\boldsymbol{f}_{\mathrm{p}}^1, \boldsymbol{f}_{\mathrm{p}}^2, \boldsymbol{f}_{\mathrm{p}}^3\}$ to generate channel attention map (*i.e.*, *Concat&CA*). Take the first SeAsU as an example, we illustrate the detailed structures of its four variants (*i.e.*, No. 1-No. 4) and its original version (*i.e.*, No. 5) in Fig. 5 to intuitively show the structure differences. We report the quantitative ablation results in Tab. IV.

We observe that the performance of our adaptive fusion of SelfCA and AssCA (*i.e.*, No. 5) is the most balanced across the four metrics, and is leading overall. Specifically, compared to the variant of fixed weights, our original SeAsU outperforms it on most metrics, which demonstrates that adaptive learning is more flexible and more suitable for deep learning methods than the indiscriminate fusion. Compared to the single SelfCA and the single AssCA, our original SeAsU outperforms them comprehensively, which indicates the inseparability of SelfCA and AssCA and the rationality of our mutual assistance. Compared to the last variant *Concat&CA*, our original SeAsU outperforms it in three metrics, which demonstrates that our elaborately designed adaptive mutual assistance structure is superior to the direct and simple fusion manner.

Specifically, we present the learnable weights $\{\beta^1, \gamma^1\}$, $\{\beta^2, \gamma^2\}$, and $\{\beta^3, \gamma^3\}$, which are $\{0.4662, 0.5338\}$, $\{0.4626, 0.5374\}$, and $\{0.4933, 0.5067\}$, respectively. The learned weights may imply that the assistance channel attention is more vital than the self channel attention in

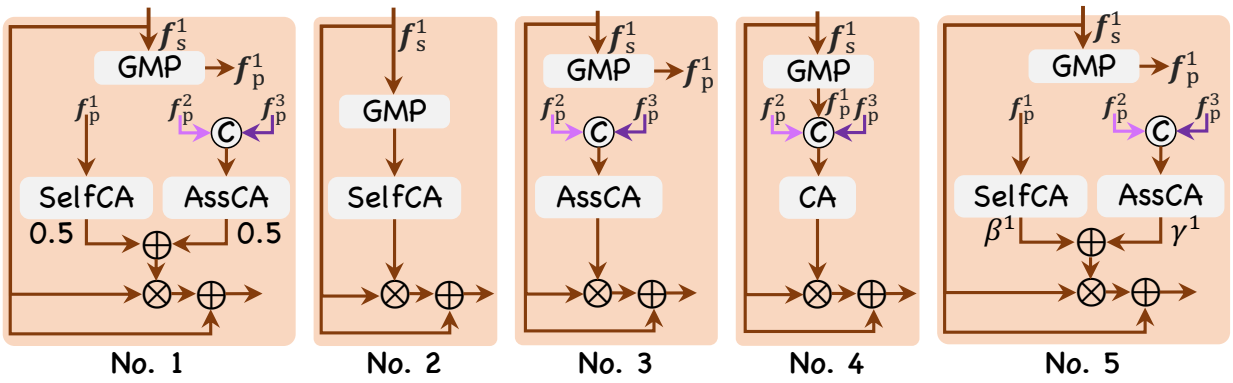

Fig. 5. Structures of four SeAsU variants (*i.e.*, No. 1-No. 4) and the complete SeAsU (*i.e.*, No. 5).

TABLE V
ABLATION STUDIES ON EVALUATING THE EFFECTIVENESS OF THE HYBRID LOSS FUNCTION. THE BEST ONE IN EACH COLUMN IS **BOLD**.

No.	L_{wBCE}	L_{wIoU}	L_{Fm}	EORSSD [29]			
				$S_\alpha \uparrow$	$F_\beta^{\max} \uparrow$	$E_\xi^{\max} \uparrow$	$\mathcal{M} \downarrow$
1	✓			0.9363	0.8823	0.9801	0.0056
2	✓	✓		0.9404	0.8944	0.9836	**0.0045**
3	✓		✓	0.9365	0.8950	0.9824	0.0054
4	✓	✓	✓	**0.9428**	**0.8962**	**0.9844**	0.0048

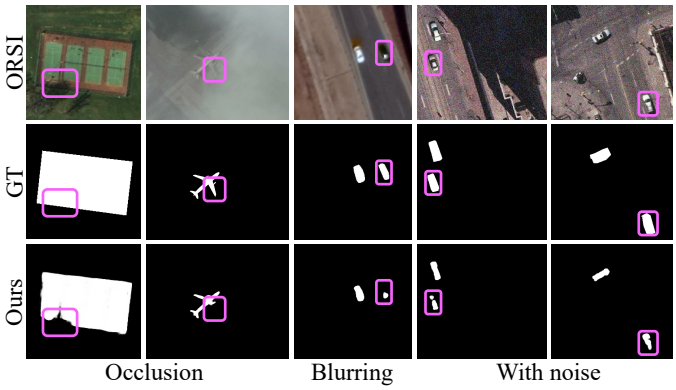

Fig. 6. Some failure cases of our FreMaNet. Zoom-in for details.

feature enhancement. This is a bit counterintuitive. But it also reveals a truth that the data-driven model should let the weights learn from the data, rather than setting a fixed empirical parameter, which further highlights the rationality of our adaptive fusion in SeAsU of MaCA.

3) The Effectiveness of the Hybrid Loss Function: Our hybrid loss function comprises of wBCE loss, wIoU loss, and Fm loss, supervising our FreMaNet at the pixel level, patch level, and measurement level. We design three stacked variants to demonstrate the effectiveness of combining these three losses. We provide the quantitative results in Tab. V. Both L_{wIoU} and L_{Fm} can independently improve the performance of our FreMaNet. As L_{wIoU} and L_{Fm} are successively stacked on L_{wBCE}, the performance of our FreMaNet shows an upward trend. Notably, L_{wBCE} and L_{wIoU} are a commonly used combination of loss functions at present [10], [12], [15], [16], which have stronger supervision capability than L_{wBCE} alone in previous practices. Furthermore, the subsequent introduction of L_{Fm} can further enhance the effectiveness of network training without increasing any parameters.

D. Limitations and Future Work

Although our FreMaNet achieves state-of-the-art performance on three datasets, it still exhibits certain limitations in extreme environments and imaging conditions. Specifically, our FreMaNet produces suboptimal saliency maps in scenarios involving shadow and cloud occlusions, blurring, and images with noise, as shown in Fig. 6. The occluded part is missing in the first two cases, while the objects in the last three cases are incomplete. The underlying reason is that heavy occlusions inherently discard critical semantic information, while blur and noise corrupt the high-frequency structural details. Since our FreMaNet explicitly relies on frequency-domain enhancement, it is highly sensitive to the integrity of high-frequency components, which encapsulate crucial structural details such as edges and textures. This leads to severe failures of our FreMaNet in the last three cases, where it is unable to completely segment the entire vehicle.

In future work, we plan to address these challenges from two perspectives. First, we will explore robust representation learning by incorporating image restoration modules (*e.g.*, denoising and deblurring mechanisms) as a preprocessing or joint-learning step to enhance feature representations. Second, for severe occlusions, we intend to introduce multi-modal data fusion (such as integrating SAR imagery, which is invariant to clouds and lighting) to reconstruct the missing contents, thereby further pushing the boundaries of the proposed method in complex and real-world applications.

V. CONCLUSION

In this paper, we propose a lightweight ORSI-SOD solution, termed FreMaNet. Our FreMaNet aims at addressing the high complexity of lightweight MobileViT-based methods and the low performance of lightweight CNN-based methods. Following the strategy of intra-level modeling and inter-level assistance, we successively insert FreSA and MaCA with

extremely limited complexity into the lightweight encoder-decoder architectures. FreSA achieves more efficient global modeling of single-level features in the frequency domain through multiplication than the vanilla self-attention. MaCA first captures adjacent relationships. Then, it constructs the adaptive mutual assistance relationships of features at different levels in the channel domain. These two attentions work together with a powerful yet lightweight MobileViT and an efficient saliency decoder, endowing our FreMaNet with only 4.91M parameters, 4.52G FLOPs, and an inference speed of 208 fps. Moreover, its performance is not inferior to any lightweight ORSI-SOD methods, and it is even highly competitive compared to normal-size ORSI-SOD methods. Overall, our lightweight FreMaNet achieves a better balance between model complexity and detection accuracy than existing lightweight ORSI-SOD methods.

REFERENCES

[1] A. Borji, M.-M. Cheng, Q. Hou, H. Jiang, and J. Li, "Salient object detection: A survey," *Comput. Vis. Media*, vol. 5, no. 2, pp. 117–150, Jun. 2019.

[2] W. Wang, J. Shen, J. Xie, M.-M. Cheng, H. Ling, and A. Borji, "Revisiting video saliency prediction in the deep learning era," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 220–237, Jan. 2021.

[3] G. Li, Z. Liu, and H. Ling, "ICNet: Information conversion network for RGB-D based salient object detection," *IEEE Trans. Image Process.*, vol. 29, pp. 4873–4884, Mar. 2020.

[4] G. Li, Z. Liu, M. Chen, Z. Bai, W. Lin, and H. Ling, "Hierarchical alternate interaction network for RGB-D salient object detection," *IEEE Trans. Image Process.*, vol. 30, pp. 3528–3542, Mar. 2021.

[5] G. Li, Z. Liu, L. Ye, Y. Wang, and H. Ling, "Cross-modal weighting network for RGB-D salient object detection," in *Proc. ECCV*, Aug. 2020, pp. 665–681.

[6] M. Song, L. Li, X. Yu, and C. Chen, "Pushing the boundaries of salient object detection: A denoising-driven approach," *IEEE Trans. Image Process.*, vol. 34, pp. 3903–3917, Jun. 2025.

[7] X. Hu, F. Sun, X. Zhang, C. Jia, and S. Ma, "DINet: Depth-guided and iterative refinement network for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–13, 2025.

[8] G. Xing, M. Wang, F. Wang, F. Sun, and H. Li, "Lightweight edge-aware mamba-fusion network for weakly supervised salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–13, 2025.

[9] J. Han, J. Sun, F. Wang, F. Sun, and H. Li, "ORSIDiff: Diffusion model for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–15, 2025.

[10] 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.

[11] Z. Bai, G. Li, and Z. Liu, "Global-local-global context-aware network for salient object detection in optical remote sensing images," *ISPRS J. Photogramm. Remote Sens.*, vol. 198, pp. 184–196, Apr. 2023.

[12] 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.

[13] Y. Xie, S. Liu, H. Chen, S. Cao, H. Zhang, D. Feng, Q. Wan, J. Zhu, and Q. Zhu, "Localization, balance, and affinity: A stronger multifaceted collaborative salient object detector in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–17, 2025.

[14] K. Wu, Y. Zhang, L. Ru *et al.*, "A semantic-enhanced multi-modal remote sensing foundation model for Earth observation," *Nat. Mach. Intell.*, vol. 7, pp. 1235–1249, Aug. 2025.

[15] G. Li, Z. Liu, Z. Bai, W. Lin, and H. Ling, "Lightweight salient object detection in optical remote sensing images via feature correlation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–12, 2022.

[16] G. Li, Z. Liu, X. Zhang, and W. Lin, "Lightweight salient object detection in optical remote-sensing images via semantic matching and edge alignment," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–12, 2023.

[17] B. Liang and H. Luo, "MEANet: An effective and lightweight solution for salient object detection in optical remote sensing images," *Expert Syst. Appl.*, vol. 238, p. 121778, Mar. 2024.

[18] H. Luo, J. Wang, and B. Liang, "Spatial attention feedback iteration for lightweight salient object detection in optical remote sensing images," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 13 809–13 823, Jul. 2024.

[19] Z. Li, Y. Miao, X. Li, W. Li, J. Cao, Q. Hao, D. Li, and Y. Sheng, "Speed-oriented lightweight salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–14, 2025.

[20] J. Liu, J. He, H. Chen, R. Yang, and Y. Huang, "A lightweight semantic- and graph-guided network for advanced optical remote sensing image salient object detection," *Remote Sens.*, vol. 17, no. 5, p. 816, Feb. 2025.

[21] Z. Ai, H. Luo, and J. Wang, "A lightweight multistream framework for salient object detection in optical remote sensing," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–15, Mar. 2025.

[22] J. Han, F. Sun, Y. Hou, J. Sun, and H. Li, "Exploring a lightweight and efficient network for salient object detection in ORSI," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–14, Jul. 2025.

[23] Z. Tu, C. Wang, C. Li, M. Fan, H. Zhao, and B. Luo, "ORSI salient object detection via multiscale joint region and boundary model," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, 2022.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, May 2015, pp. 1–14.

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

[26] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE ICCV*, Oct. 2021, pp. 9992–10 002.

[27] 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.

[28] C. Li, R. Cong, J. Hou, S. Zhang, Y. Qian, and S. Kwong, "Nested network with two-stream pyramid for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 9156–9166, Nov. 2019.

[29] Q. Zhang, R. Cong, C. Li, M.-M. Cheng, Y. Fang, X. Cao, Y. Zhao, and S. Kwong, "Dense attention fluid network for salient object detection in optical remote sensing images," *IEEE Trans. Image Process.*, vol. 30, pp. 1305–1317, 2021.

[30] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE CVPR*, Jun. 2018, pp. 4510–4520.

[31] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, and J. Sun, "RepVGG: Making VGG-style ConvNets great again," in *Proc. IEEE CVPR*, Jun. 2021, pp. 13 728–13 737.

[32] S. Mehta and M. Rastegari, "MobileViT: Light-weight, general-purpose, and mobile-friendly vision transformer," in *Proc. ICLR*, Apr. 2021, pp. 1–13.

[33] J. B. J. baron de Fourier, *Théorie analytique de la chaleur*. Firmin Didot, 1822.

[34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NeurIPS*, Dec. 2017, pp. 6000–6010.

[35] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2Net: A new multi-scale backbone architecture," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 2, pp. 652–662, Feb. 2021.

[36] Q.-L. Zhang and Y.-B. Yang, "ResT: An efficient transformer for visual recognition," *Proc. NeurIPS*, vol. 34, pp. 15 475–15 485, 2021.

[37] Y. Sun, J. Yan, J. Qian, C. Xu, J. Yang, and L. Luo, "Dual-perspective united transformer for object segmentation in optical remote sensing images," *Proc. IJCAI*, 2025.

[38] Z. Huang, H. Chen, B. Liu, and Z. Wang, "Semantic-guided attention refinement network for salient object detection in optical remote sensing images," *Remote Sens.*, vol. 13, no. 11, p. 2163, May 2021.

[39] X. Zhou, K. Shen, Z. Liu, C. Gong, J. Zhang, and C. Yan, "Edge-aware multiscale feature integration network for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, 2022.

[40] G. Li, Z. Liu, W. Lin, and H. Ling, "Multi-content complementation network for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, 2022.

[41] G. Li, Z. Liu, D. Zeng, W. Lin, and H. Ling, "Adjacent context coordination network for salient object detection in optical remote

sensing images," *IEEE Trans. Cybern.*, vol. 53, no. 1, pp. 526–538, Jan. 2023.
[42] J. Zhao, Y. Jia, L. Ma, and L. Yu, "Multilevel interactive reverse-guided network for salient object detection in optical remote sensing images," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 12 983–12 999, Jul. 2024.
[43] ——, "Recurrent adaptive graph reasoning network with region and boundary interaction for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–20, Jul. 2024.
[44] Y. Quan, H. Xu, R. Wang, Q. Guan, and J. Zheng, "ORSI salient object detection via progressive semantic flow and uncertainty-aware refinement," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–13, Jan. 2024.
[45] Y. Jia, J. Zhao, L. Ma, and L. Yu, "Multistrategy region and boundary interaction network for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–16, Jul. 2025.
[46] Y. Ge, T. Liang, J. Ren, M. He, H. Bi, and Q. Zhang, "Semantic awareness aggregation for salient object detection in remote sensing images," *Eng. Appl. Artif. Intell.*, vol. 160, p. 111837, Nov. 2025.
[47] X. Zhou, K. Shen, L. Weng, R. Cong, B. Zheng, J. Zhang, and C. Yan, "Edge-guided recurrent positioning network for salient object detection in optical remote sensing images," *IEEE Trans. Cybern.*, vol. 53, no. 1, pp. 539–552, Jan. 2023.
[48] Y. Gu, S. Chen, X. Sun, J. Ji, Y. Zhou, and R. Ji, "Optical remote sensing image salient object detection via bidirectional cross-attention and attention restoration," *Pattern Recognit.*, vol. 164, pp. 1–12, Aug. 2025.
[49] S. Gu, Y. Song, Y. Zhou, Y. Bai, X. Yang, and Y. He, "PRNet: Parallel refinement network with group feature learning for salient object detection in optical remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 21, pp. 1–5, May 2024.
[50] Y. Wang, P. Zheng, R. Zhao, and L. Wang, "Robust salient object detection in optical remote sensing images via multiscale contextual attention and feature enhancement," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, pp. 1–20, Sept. 2025.
[51] H. Luo, J. He, and S. Yang, "Prompt-driven multi-task learning with task tokens for ORSI salient object detection," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, pp. 1–15, Sept. 2025.
[52] P. Dong, B. Wang, R. Cong, H.-H. Sun, and C. Li, "Transformer with large convolution kernel decoder network for salient object detection in optical remote sensing images," *Comput. Vis. Image Und.*, vol. 240, p. 103917, Mar. 2024.
[53] Q. Wang, Y. Liu, Z. Xiong, and Y. Yuan, "Hybrid feature aligned network for salient object detection in optical remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, 2022.
[54] J. Zhao, Y. Jia, L. Ma, and L. Yu, "Adaptive dual-stream sparse transformer network for salient object detection in optical remote sensing images," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 5173–5192, Feb. 2024.
[55] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. ECCV*, Sept. 2018, pp. 3–19.
[56] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *Proc. IEEE CVPR*, Jun. 2019, pp. 3902–3911.
[57] K. Zhao, S. Gao, W. Wang, and M.-M. Cheng, "Optimizing the F-measure for threshold-free salient object detection," in *Proc. IEEE ICCV*, Oct. 2019, pp. 8849–8857.
[58] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. NeurIPS*, Dec. 2019, pp. 8024–8035.
[59] 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.
[60] M.-M. Cheng and D.-P. Fan, "Structure-measure: A new way to evaluate foreground maps," *Int. J. Comput. Vis.*, vol. 129, no. 9, pp. 2622–2638, Sept. 2021.
[61] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE CVPR*, Jun. 2009, pp. 1597–1604.
[62] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, and A. Borji, "Enhanced-alignment measure for binary foreground map evaluation," in *Proc. IJCAI*, Jul. 2018, pp. 698–704.