Texture-Semantic Collaboration Network for ORSI Salient Object Detection

Gongyang Li, Zhen Bai, and Zhi Liu, *Senior Member, IEEE*

***Abstract*—Salient object detection (SOD) in optical remote sensing images (ORSIs) has become increasingly popular recently. Due to the characteristics of ORSIs, ORSI-SOD is full of challenges, such as multiple objects, small objects, low illuminations, and irregular shapes. To address these challenges, we propose a concise yet effective *Texture-Semantic Collaboration Network* (TSCNet) to explore the collaboration of texture cues and semantic cues for ORSI-SOD. Specifically, TSCNet is based on the generic encoder-decoder structure. In addition to the encoder and decoder, TSCNet includes a vital Texture-Semantic Collaboration Module (TSCM), which performs valuable feature modulation and interaction on basic features extracted from the encoder. The main idea of our TSCM is to make full use of the texture features at the lowest level and the semantic features at the highest level to achieve the expression enhancement of salient regions on features. In the TSCM, we first enhance the position of potential salient regions using semantic features. Then, we render and restore the object details using the texture features. Meanwhile, we also perceive regions of various scales, and construct interactions between different regions. Thanks to the perfect combination of TSCM and generic structure, our TSCNet can take care of both the position and details of salient objects, effectively handling various scenes. Extensive experiments on three datasets demonstrate that our TSCNet achieves competitive performance compared to 14 state-of-the-art methods. The code and results of our method are available at https://github.com/MathLee/TSCNet.**

***Index Terms*—Salient object detection, optical remote sensing image, texture features, semantic features.**

I. Introduction

SALIENT object detection (SOD) focuses on extracting the most attractive objects/regions in a scene [1]–[3]. Recently, SOD in optical remote sensing images (ORSIs) has become a shining topic in the SOD community, and aims to capture the most attention-grabbing ships, airplanes, cars, buildings, islands, rivers, *etc.*, in ORSIs. ORSI-SOD has great applications in urban planning, land resource evaluation, environmental monitoring, and agricultural production [4]–[7].

Researchers have made remarkable achievements in SOD in natural scene images (NSIs) [8]–[14]. However, due to the differences in scenes and shooting between NSIs and ORSIs, NSI-SOD methods may not always be able to handle challenging scenes in ORSIs well. Therefore, many efforts have been made in ORSI-SOD, resulting in some effective specialized solutions. Among existing specialized ORSI-SOD methods, some methods focus on mining informative clues from features at a single level [15], [16] (called the single-level type), while some focus on extracting contextual clues from features at adjacent levels [4], [17]–[19] (called the adjacent-level type). Differently, some methods focus on the details of salient objects, and explore the edge information for ORSI-SOD [20], [21] (called the detail type). While some focus on the position of salient objects, and explore the global semantic information for ORSI-SOD [22], [23] (called the position type). Although these four types of methods have promoted the development of ORSI-SOD, each type of method has its own drawbacks. Obviously, the single-level type ignores contextual information. The adjacent-level type only utilizes information from nearby adjacent levels, ignoring information from longer distances (*i.e.*, levels). The detail type and the position type only consider edge information or position information, and both types of methods are suboptimal.

Inspired by the above observations, we integrate the advantages of four types of ORSI-SOD methods to alleviate their problems. Concretely, we utilize the contextual information from further distances (*i.e.*, levels), and take into account both detail and position information. We believe that the above information is essential to handle various complex and ever-changing scenes in ORSIs. Based on this idea, we propose a concise yet effective Texture-Semantic Collaboration Network (TSCNet) for ORSI-SOD, which makes an attempt to explore the collaboration of texture cues and semantic cues. Similar to previous ORSI-SOD methods, our TSCNet also follows the classic encoder-decoder structure [24]. In TSCNet, we propose the key Texture-Semantic Collaboration Module (TSCM) to modulate current features using semantic features and texture features, enabling current features to obtain valuable position and detail information. We also introduce the transformer [25] into TSCM to establish region-level connections, which is effective in handling scenes of multiple objects. In this way, our TSCNet can handle various challenging scenes of ORSIs. The saliency map generated by our TSCNet can accurately locate salient objects and finely outline the details of salient objects, making our TSCNet a competitive detector.

Our main contributions are threefold:

- We explore the collaboration of texture cues and semantic cues for ORSI-SOD, and propose a novel *TSCNet*, which takes into account both position highlighting and detail rendering of salient objects in ORSIs.

This work was supported in part by the National Natural Science Foundation of China under Grant 62171269 and 62376148, and in part by the China Postdoctoral Science Foundation under Grant 2022M722037. *(Corresponding author: Zhi Liu.)*

The authors are with Key Laboratory of Specialty Fiber Optics and Optical Access Networks, Joint International Research Laboratory of Specialty Fiber Optics and Advanced Communication, Shanghai Institute for Advanced Communication and Data Science, Shanghai University, Shanghai 200444, China, and School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China. Gongyang Li and Zhi Liu are also with Wenzhou Institute of Shanghai University, Wenzhou 325000, China (email: ligongyang@shu.edu.cn; bz536476@163.com; liuzhisjtu@163.com).

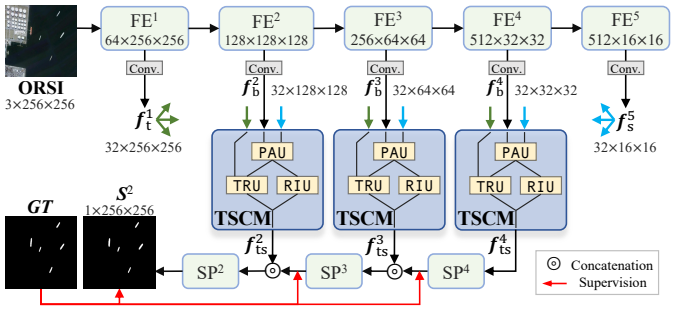

Fig. 1. Architecture of the proposed TSCNet.

- We propose a *Texture-Semantic Collaboration Module* to coordinate the lowest-level texture features and the highest-level semantic features to perform valuable modulation and interaction on other levels of features, enhancing the expression of salient regions.
- We evaluate the proposed TSCNet on three public ORSI-SOD datasets, *i.e.*, EORSSD, ORSSD, and ORSI-4199. Comprehensive experiments show that our TSCNet is competitive and that our key module is effective.

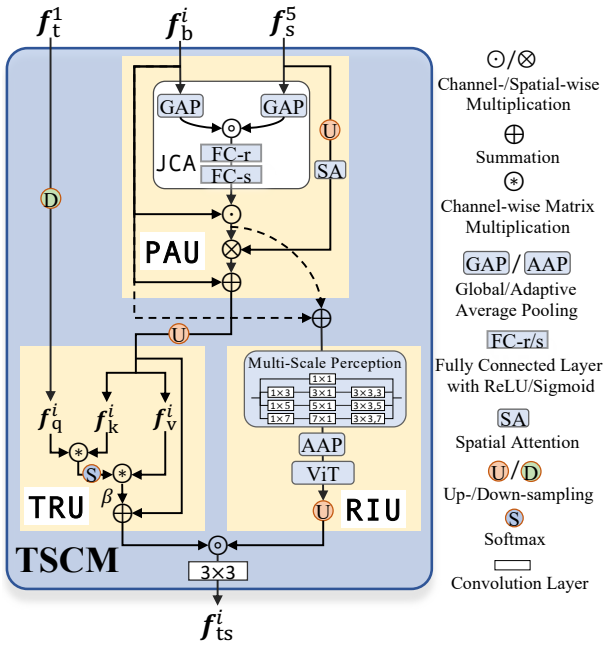

Fig. 2. Illustration of the TSCM, which consists of PAU, TRU, and RIU.

II. Proposed Method

A. Network Overview

As illustrated in Fig. 1, the proposed TSCNet is based on the encoder-decoder structure, and consists of the encoder, the Texture-Semantic Collaboration Module (TSCM), and the decoder. We adopt the VGG [26] as our encoder with the input size of $3 \times 256 \times 256$, also known as the feature extractor (FE), and denote its basic block as FE^i $(i = 1, 2, 3, 4, 5)$. We use a convolution layer after each basic block to compress the channel number of features, resulting in five-level features. This operation can significantly reduce computational complexity. Here, we denote the features of FE^1 as $\boldsymbol{f}_{\text{t}}^1 \in \mathbb{R}^{32\times256\times256}$ (*i.e.*, texture features), the features of FE^5 as $\boldsymbol{f}_{\text{s}}^5 \in \mathbb{R}^{32\times16\times16}$ (*i.e.*, semantic features), and the other features as $\boldsymbol{f}_{\text{b}}^i \in \mathbb{R}^{c_i\times h_i\times w_i}$ $(i = 2, 3, 4)$, where h_i and w_i are $\frac{256}{2^{i-1}}$, and c_i is 32. Then, we transfer the valuable cues of $\boldsymbol{f}_{\text{t}}^1$ and $\boldsymbol{f}_{\text{s}}^5$ to $\boldsymbol{f}_{\text{b}}^i$ using the TSCM. In the TSCM, we first adopt Position Anchoring Unit (PAU) to explore the semantic features to enhance the position of salient regions in both channel and spatial dimensions. Next, we adopt Texture Rendering Unit (TRU) to explore the texture features to render and restore the object details, and adopt Region Interaction Unit (RIU) to construct interactions between different regions. In this way, we get $\boldsymbol{f}_{\text{ts}}^i \in \mathbb{R}^{c_i\times2h_i\times2w_i}$ $(i = 2, 3, 4)$ from TSCM. Finally, we gradually infer the saliency map $\boldsymbol{S}^2 \in \mathbb{R}^{1\times256\times256}$ in the decoder using the saliency prediction block denoted as SP^i $(i = 2, 3, 4)$. Notably, the deep supervision is introduced in the training phase to accelerate network convergence.

B. Texture-Semantic Collaboration Module

As we all know, in convolutional neural networks, as the number of convolution layers increases, the texture information of objects will gradually be lost, and the semantic information (*i.e.*, position information) will dominate. In other words, the low-level features contain detailed texture cues, while the high-level features contain sufficient semantic cues. Due to the complexity and variability of ORSIs, utilizing texture cues and semantic cues simultaneously is beneficial for ORSI-SOD. Therefore, we propose TSCM to coordinate the lowest-level texture features, *i.e.*, $\boldsymbol{f}_{\text{t}}^1$, and the highest-level semantic features, *i.e.*, $\boldsymbol{f}_{\text{s}}^5$, to perform valuable modulation and interaction on $\boldsymbol{f}_{\text{b}}^i$. We illustrate the detailed structure of TSCM in Fig. 2, which consists of PAU, TRU, and RIU.

1) Position Anchoring Unit. As shown at the top of Fig. 2, the inputs of PAU are $\boldsymbol{f}_{\text{b}}^i$ and $\boldsymbol{f}_{\text{s}}^5$. Obviously, in PAU, we aim at exploring the semantic information of $\boldsymbol{f}_{\text{s}}^5$ to anchor salient objects in both channel and spatial dimensions using the attention mechanism [27].

Differently from traditional channel attention (CA), we utilize $\boldsymbol{f}_{\text{s}}^5$ to assist $\boldsymbol{f}_{\text{b}}^i$ to achieve accurate channel-wise feature enhancement, which we call joint channel attention (JCA). We first extremely compress $\boldsymbol{f}_{\text{b}}^i$ and $\boldsymbol{f}_{\text{s}}^5$ to the global representation with the size of $32 \times 1 \times 1$. Then, we concatenate them and adopt two fully connected layers with suitable activation functions to generate the joint channel attention map with the size of $32\times1\times1$. Next, similar to traditional CA, we modulate $\boldsymbol{f}_{\text{b}}^i$ using the joint channel attention map to highlight important channels, generating $\boldsymbol{f}_{\text{jca}}^i \in \mathbb{R}^{c_i\times h_i\times w_i}$. We formulate JCA as follows:

$$\boldsymbol{f}_{\text{jca}}^i = \text{FC}_{\text{s}}\Big(\text{FC}_{\text{r}}\big(\text{GAP}(\boldsymbol{f}_{\text{b}}^i) \circledcirc \text{GAP}(\boldsymbol{f}_{\text{s}}^5)\big)\Big) \odot \boldsymbol{f}_{\text{b}}^i, \quad (1)$$

where $\circledcirc$ is the concatenation operator, $\odot$ is the channel-wise multiplication, $\text{GAP}(\cdot)$ is the global average pooling layer, and $\text{FC}_{\text{r}}(\cdot)/\text{FC}_{\text{s}}(\cdot)$ is the fully connected layer with ReLU/sigmoid activation function. With the help of semantic cues, our JCA is more sensitive to informative channels than traditional CA.

At the rest of PAU, we focus on the spatial enhancement of $\boldsymbol{f}^i_{\rm jca}$ to anchor the position of salient objects, which is a supplement to JCA. We extract the spatial attention map from $\boldsymbol{f}^5_{\rm s}$, and anchor salient regions of $\boldsymbol{f}^i_{\rm jca}$ through the spatial-wise multiplication. Moreover, we adopt the residual connection to fuse $\boldsymbol{f}^i_{\rm b}$ and the effectively enhanced features, generating the output features of PAU, *i.e.*, $\boldsymbol{f}^i_{\rm pau} \in \mathbb{R}^{c_i \times h_i \times w_i}$.

2) Texture Rendering Unit. The position of salient objects is well highlighted in the PAU. We turn our attention to the texture of salient objects. In TRU, we aim to achieve the feature super-resolution of $\boldsymbol{f}^i_{\rm pau}$ with the assistance of $\boldsymbol{f}^1_{\rm t}$, that is, we not only restore the texture of $\boldsymbol{f}^i_{\rm pau}$, but also enlarge the size of $\boldsymbol{f}^i_{\rm pau}$.

We first up-sample $\boldsymbol{f}^i_{\rm pau}$ from $32\times h_i \times w_i$ to $32\times 2h_i \times 2w_i$, generating $\hat{\boldsymbol{f}}^i_{\rm pau}$. Meanwhile, we down-sample $\boldsymbol{f}^1_{\rm t}$ from $32\times 256\times 256$ to $32\times 2h_i \times 2w_i$, generating $\hat{\boldsymbol{f}}^1_{\rm t}$. Next, according to the standard self-attention mechanism [28], we generate the corresponding $\boldsymbol{f}^i_{\rm q}$ and $\{\boldsymbol{f}^i_{\rm k}, \boldsymbol{f}^i_{\rm v}\}$ from $\hat{\boldsymbol{f}}^1_{\rm t}$ and $\hat{\boldsymbol{f}}^i_{\rm pau}$, respectively. We model correlations between $\boldsymbol{f}^i_{\rm q}$ and $\boldsymbol{f}^i_{\rm k}$, and transfer them to $\boldsymbol{f}^i_{\rm v}$ to render the texture of features, generating $\boldsymbol{f}^i_{\rm sp}$ with the size of $2h_i \times 2w_i$. Finally, we also adopt the residual connection to fuse $\boldsymbol{f}^i_{\rm sp}$ and $\hat{\boldsymbol{f}}^i_{\rm pau}$ with a coefficient of β, generating the output features of TRU, *i.e.*, $\boldsymbol{f}^i_{\rm tru} \in \mathbb{R}^{c_i \times 2h_i \times 2w_i}$.

Notably, the matrix multiplication used in our TRU is the channel-wise matrix multiplication, which is different from that in the standard self-attention mechanism. Compared to the standard one, our channel-wise matrix multiplication can significantly reduce computational complexity and memory usage, enabling the use of self-attention attention at multiple levels. As is well known, for $\{\boldsymbol{f}^1, \boldsymbol{f}^2\} \in \mathbb{R}^{c\times h\times w}$, the standard one performs matrix multiplication at the size of $(hw\times c)\times(c\times hw)$, generating features with size of $hw\times hw$. Differently, our TRU only performs the matrix multiplication for two corresponding channels at the size of $(h\times w)\times(w\times h)$, and there are c channels in total, generating features with the size of $c\times(h\times w)$[1].

3) Region Interaction Unit. Similar to TRU, RIU can also perform the function of rendering textures, but it reconstructs textures by establishing region-level connections between different regions. The input of RIU is the summation of $\boldsymbol{f}^i_{\rm jca}$ and $\boldsymbol{f}^i_{\rm b}$, denoted as $\boldsymbol{f}^i_{\rm in} \in \mathbb{R}^{c_i\times h_i\times w_i}$. According to the fact that salient objects in ORSIs have various sizes and shapes, we first adopt the multi-scale perception operation to perceive regions of various scales. The multi-scale perception operation has four parallel dilated convolution branches [29], which can be formulated as follows:

$$\boldsymbol{f}^{i,j}_{\rm br} = \begin{cases} {\rm C}_{1\times1}(\boldsymbol{f}^i_{\rm in}), & j=1, \\ {\rm C}_{3\times3,{\rm k}}({\rm C}_{{\rm k}\times1}({\rm C}_{1\times{\rm k}}(\boldsymbol{f}^i_{\rm in}))), & j=2,3,4; k=2j-1, \end{cases} \tag{2}$$

where $\boldsymbol{f}^{i,j}_{\rm br} \in \mathbb{R}^{c_i\times h_i\times w_i}$ is the output of j-th branch and ${\rm C}_{{\rm k}_1\times{\rm k}_2,{\rm r}}(\cdot)$ is the convolution layer with kernel size of ${\rm k}_1\times{\rm k}_2$ and dilated rate of r. The output of these four branches is fused through the concatenation and a convolution layer, generating $\boldsymbol{f}^i_{\rm msp} \in \mathbb{R}^{c_i\times h_i\times w_i}$.

Moreover, we adopt the effective transformer (*i.e.*, ViT [25]) to comprehensively model the long-range texture dependencies of different regions through the multi-head self-attention mechanism. Notably, we set the input size of ViT as $32\times32\times32$ and the patch size as 1×1. This is because that $\boldsymbol{f}^4_{\rm msp}$, $\boldsymbol{f}^3_{\rm msp}$, and $\boldsymbol{f}^2_{\rm msp}$ are with various sizes, if we set the input size and patch size of ViT to the fixed size, we can reconstruct texture at different levels on the same size, thus maintaining texture consistency. Therefore, as depicted in RIU of Fig. 2, we insert an adaptive average pooling layer between the multi-scale perception and ViT to compress $\boldsymbol{f}^i_{\rm msp}$ from $h_i\times w_i$ to 32×32. Then, we enhance the region interaction using ViT, and up-sample the output of ViT to $c_i\times2h_i\times2w_i$, generating the output features of RIU, *i.e.*, $\boldsymbol{f}^i_{\rm riu} \in \mathbb{R}^{c_i\times2h_i\times2w_i}$.

Finally, we adopt the concatenation operator and the convolution layer to fuse $\boldsymbol{f}^i_{\rm tru}$ and $\boldsymbol{f}^i_{\rm riu}$, generating the output features of TSCM, *i.e.*, $\boldsymbol{f}^i_{\rm ts} \in \mathbb{R}^{c_i\times2h_i\times2w_i}$. Through the collaboration of these three units, our TSCM makes full use of texture and semantic features to enhance the expression of salient regions in a comprehensive manner.

C. Loss Function

The basic SP block of our encoder generally consists of two convolution layers, a dropout layer, and a deconvolution layer, while the last SP block (*i.e.*, SP^2) consists of three convolution layers. Due to the introduction of deep supervision into the network training, in addition to the last SP block generating the final saliency map $\boldsymbol{S}^2$, the other two SP blocks also generate two lateral saliency maps, *i.e.*, $\boldsymbol{S}^3 \in \mathbb{R}^{1\times256\times256}$ and $\boldsymbol{S}^4 \in \mathbb{R}^{1\times128\times128}$. For these three saliency maps, we impose the hybrid loss function, including the binary cross-entropy (BCE) loss and intersection-over-union (IoU) loss, to each one. Thus, we formulate the total loss function $L_{\rm total}$ as follows:

$$L_{\rm total} = \sum_{i=2}^{4} \big(\ell^i_{\rm bce}({\rm up}(\boldsymbol{S}^i), \boldsymbol{GT}) + \ell^i_{\rm iou}({\rm up}(\boldsymbol{S}^i), \boldsymbol{GT})\big), \tag{3}$$

where $\ell^i_{\rm bce}(\cdot)$ and $\ell^i_{\rm iou}(\cdot)$ are BCE loss and IoU loss, respectively, $\boldsymbol{GT} \in \{0,1\}^{1\times256\times256}$ is the binary ground truth (GT) map, and ${\rm up}(\cdot)$ is the up-sampling operation if the sizes of $\boldsymbol{S}^i$ and $\boldsymbol{GT}$ do not match.

III. Experiments

A. Experimental Setup

1) Datasets. We evaluate our TSCNet on three datasets, *i.e.*, ORSSD [4], EORSSD [5], and ORSI-4199 [21]. ORSSD has 800 images, of which 600 images are for training and 200 images are for testing. EORSSD has 2000 images, of which 1400 images are for training and 600 images are for testing. ORSI-4199 has 4199 images, of which 2000 images are for training and 2199 images are for testing.

2) Evaluation Metrics. We use four quantitative evaluation metrics for performance measurement, including S-measure (S_α, α = 0.5) [30], mean F-measure (F_β, β^2 = 0.3) [31], mean E-measure (E_ξ) [32], and mean absolute error ($\mathcal{M}$).

3) Implementation Details. We conduct all experiments using the PyTorch on a computer with an NVIDIA RTX 3090 GPU (24GB memory). During the training phase, all images and ground truths are resized to 256×256, and then flipping

[1]In our TSCNet, since h is equal to w, $h\times h$ is the same as $h\times w$.

TABLE I
QUANTITATIVE COMPARISONS WITH STATE-OF-THE-ART METHODS ON EORSSD, ORSSD, AND ORSI-4199 DATASETS. ↑ INDICATES THAT THE HIGHER THE BETTER, WHILE ↓ IS THE OPPOSITE. WE MARK THE RESULTS THAT ARE BETTER THAN OUR METHOD IN **BLUE**.

Methods	EORSSD [5]				ORSSD [4]				ORSI-4199 [21]			
	$S_\alpha \uparrow$	$F_\beta^{\text{mean}} \uparrow$	$E_\xi^{\text{mean}} \uparrow$	$\mathcal{M} \downarrow$	$S_\alpha \uparrow$	$F_\beta^{\text{mean}} \uparrow$	$E_\xi^{\text{mean}} \uparrow$	$\mathcal{M} \downarrow$	$S_\alpha \uparrow$	$F_\beta^{\text{mean}} \uparrow$	$E_\xi^{\text{mean}} \uparrow$	$\mathcal{M} \downarrow$
SUCA [8]	0.8988	0.7949	0.9277	0.0097	0.8989	0.8237	0.9400	0.0145	**0.8794**	0.8590	0.9356	0.0304
PA-KRN [9]	0.9192	0.8358	0.9536	0.0104	0.9239	0.8727	0.9620	0.0139	0.8491	0.8324	0.9168	0.0382
VST [10]	0.9208	0.8263	0.9442	0.0067	0.9365	0.8817	0.9621	0.0094	**0.8790**	0.8524	0.9348	**0.0281**
LVNet [4]	0.8630	0.7328	0.8801	0.0146	0.8815	0.7995	0.9259	0.0207	-	-	-	-
DAFNet [5]	0.9166	0.7845	0.9291	**0.0060**	0.9191	0.8511	0.9539	0.0113	-	-	-	-
MJRBM [21]	0.9197	0.8239	0.9350	0.0099	0.9204	0.8566	0.9415	0.0163	0.8593	0.8309	0.9102	0.0374
EMFINet [20]	0.9290	0.8486	0.9604	0.0084	0.9366	0.8856	0.9671	0.0109	0.8675	0.8479	0.9257	0.0330
ERPNet [16]	0.9210	0.8304	0.9401	0.0089	0.9254	0.8745	0.9566	0.0135	0.8670	0.8374	0.9149	0.0357
ACCoNet [19]	0.9290	0.8552	0.9653	0.0074	**0.9437**	0.8971	0.9754	0.0088	0.8775	0.8620	0.9342	0.0314
CorrNet [17]	0.9289	0.8620	0.9646	0.0083	0.9380	0.9002	0.9746	0.0098	0.8623	0.8513	0.9206	0.0366
MCCNet [15]	0.9327	0.8604	0.9685	0.0066	**0.9437**	**0.9054**	0.9758	0.0087	0.8746	0.8630	0.9348	0.0316
GPNet [22]	0.9233	0.8447	0.9617	0.0085	0.9185	0.8683	0.9590	0.0125	0.8573	0.8396	0.9184	0.0384
HFANet [18]	0.9380	0.8681	0.9679	0.0070	0.9399	0.8981	0.9712	0.0092	0.8767	0.8624	0.9336	0.0314
SeaNet [23]	0.9208	0.8519	0.9651	0.0073	0.9260	0.8772	0.9722	0.0105	0.8722	0.8591	0.9363	0.0308
TSCNet (Ours)	**0.9383**	**0.8740**	**0.9717**	**0.0061**	**0.9428**	**0.9030**	**0.9804**	**0.0081**	**0.8783**	**0.8703**	**0.9418**	**0.0295**

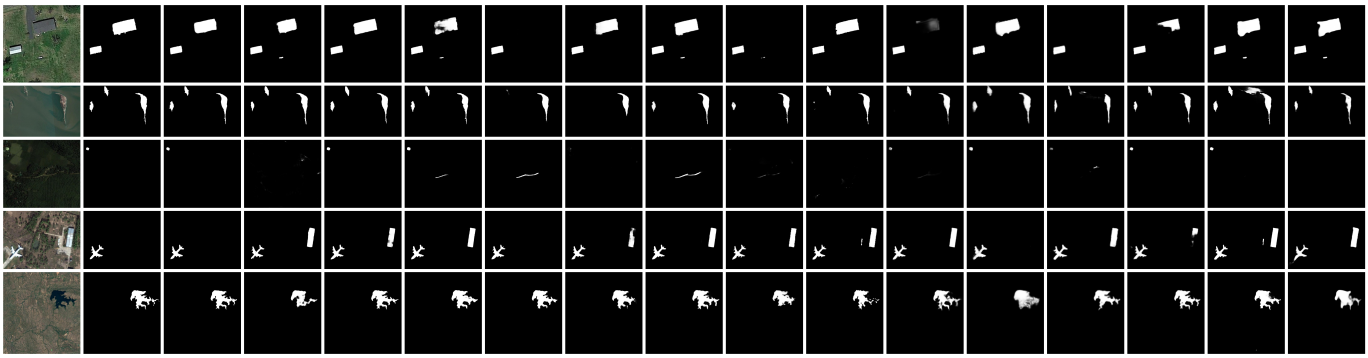

Fig. 3. Qualitative comparison with 14 state-of-the-art methods on representative scenes in ORSIs.

and rotation are adopted for data augmentation. We use the Adam optimizer for parameter updating with a base learning rate of $1e^{-4}$ and a batch size of 4. The learning rate drops to 1/10 every 30 epochs. On each dataset, we train our TSCNet for 70 epochs on its training set, and then test the trained TSCNet on its testing set.

B. Comparison with State-of-the-arts

We compare our TSCNet with 14 state-of-the-art SOD methods for NSIs and ORSIs. They are SUCA [8], PA-KRN [9] and VST [10] used for NSI-SOD, and LVNet [4], DAFNet [5], MJRBM [21], EMFINet [20], ERPNet [16], ACCoNet [19], CorrNet [17], MCCNet [15], GPNet [22], HFANet [18] and SeaNet [23] used for ORSI-SOD.

We report the quantitative performance of all methods in Tab. I. Among all 12 evaluation metrics, our method outperforms all compared methods in seven evaluation metrics, ranks second on four evaluation metrics, and ranks third on one evaluation metric. Moreover, we show the qualitative comparison on three challenging ORSI scenes in Fig. 3, including multiple objects, small objects, and irregular shapes. We can clearly observe that our method handles these challenging scenes well, and the saliency maps generated by our method are more accurate and complete than those generated by other competitors. The above quantitative and qualitative comparisons indicate that our method is a competitive saliency detector for ORSIs.

TABLE II
ABLATION RESULTS OF EVALUATING THE CONTRIBUTION OF EACH COMPONENT IN TSCNET. THE BEST ONE IN EACH COLUMN IS **BOLD**.

No.	Baseline	TSCM			EORSSD [5]			
		PAU	TRU	RIU	$S_\alpha \uparrow$	$F_\beta^{\text{mean}} \uparrow$	$E_\xi^{\text{mean}} \uparrow$	$\mathcal{M} \downarrow$
1	✓				0.8863	0.8035	0.9256	0.0184
2	✓	✓			0.9072	0.8260	0.9465	0.0113
3	✓	✓	✓		0.9146	0.8361	0.9577	0.0095
4	✓	✓		✓	0.9267	0.8554	0.9654	0.0078
5	✓	✓	✓	✓	**0.9383**	**0.8740**	**0.9717**	**0.0061**

C. Ablation Studies

We conduct ablation studies on the EORSSD dataset to evaluate the contribution of each component of our TSCNet. Concretely, we remove all TSCMs in our TSCNet, and directly connect $\boldsymbol{f}_{\text{b}}^i$ to the corresponding SP block, providing the baseline model. As shown in Tab. II, the performance of the baseline model drops significantly, with an average decrease of 5.62%, which means our TSCMs are very effective. The TSCM consists of three units of PAU and parallel TRU and RIU. In the following, we gradually add these units to the baseline model. First, we add PAU into the baseline model, resulting in an average performance improvement of 2.14%. Then, we continue to add TRU to the second variant, thereby increasing average performance by 0.96% compared to the No.2 variant. Meanwhile, we also add RIU to the second

variant, increasing average performance by 2.26% compared to the No.2 variant. Finally, we add all three units to the baseline to achieve our complete TSCNet. Thanks to the perfect collaboration between these three units, our complete TSCNet achieves satisfactory performance.

IV. Conclusion

In this brief, we make an attempt to investigate the collaboration of texture cues and semantic cues for ORSI-SOD, and propose a concise yet effective TSCNet. We implement our TSCNet on the effective encoder-decoder structure, and integrate a useful TSCM into this structure. In the TSCM, we first adopt semantic features to anchor the position of salient regions through joint channel attention and spatial attention. Then, we use texture features to assist the super-resolution of current features through an efficient variant of the self-attention mechanism. Meanwhile, we perform multi-scale perception and region-level interaction establishment to reconstruct the texture of features from a self-learning perspective. The close collaboration of all components enables our TSCNet to not only accurately locate salient objects, but also sharpen their details. Performance comparison and ablation studies demonstrate that our idea is effective, and our TSCNet exhibits competitive performance.

References

[1] C. Gong *et al.*, "Saliency propagation from simple to difficult," in *Proc. IEEE CVPR*, Jun. 2015, pp. 2531–2539.
[2] X. Zhou, Z. Liu, C. Gong, and W. Liu, "Improving video saliency detection via localized estimation and spatiotemporal refinement," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 2993–3007, Nov. 2018.
[3] K. Fu, C. Gong, I. Y.-H. Gu, and J. Yang, "Normalized cut-based saliency detection by adaptive multi-level region merging," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5671–5683, Dec. 2015.
[4] C. Li *et al.*, "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.
[5] Q. Zhang *et al.*, "Dense attention fluid network for salient object detection in optical remote sensing images," *IEEE Trans. Image Process.*, vol. 30, pp. 1305–1317, 2021.
[6] R. Cong *et al.*, "RRNet: Relational reasoning network with parallel multiscale attention for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–11, 2022.
[7] 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.
[8] J. Li, Z. Pan, Q. Liu, and Z. Wang, "Stacked U-shape network with channel-wise attention for salient object detection," *IEEE Trans. Multimedia*, vol. 23, pp. 1397–1409, 2021.
[9] B. Xu, H. Liang, R. Liang, and P. Chen, "Locate globally, segment locally: A progressive architecture with knowledge review network for salient object detection," in *Proc. AAAI*, Feb. 2021, pp. 3004–3012.
[10] N. Liu, N. Zhang, K. Wan, L. Shao, and J. Han, "Visual saliency transformer," in *Proc. IEEE ICCV*, Oct. 2021, pp. 4702–4712.
[11] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *Proc. IEEE ICCV*, Oct. 2017, pp. 202–211.
[12] Y. Zeng, P. Zhang, Z. Lin, J. Zhang, and H. Lu, "Towards high-resolution salient object detection," in *Proc. IEEE ICCV*, Oct. 2019, pp. 7233–7242.
[13] P. Zhang, W. Liu, Y. Zeng, Y. Lei, and H. Lu, "Looking for the detail and context devils: High-resolution salient object detection," *IEEE Trans. Image Process.*, vol. 30, pp. 3204–3216, 2021.
[14] X. Deng, P. Zhang, W. Liu, and H. Lu, "Recurrent multi-scale transformer for high-resolution salient object detection," in *Proc. ACM MM*, Oct. 2023, pp. 7413–7423.
[15] 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.
[16] X. Zhou *et al.*, "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.
[17] 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.
[18] 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.
[19] G. Li *et al.*, "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.
[20] X. Zhou *et al.*, "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.
[21] 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.
[22] Y. Liu, S. Zhang, Z. Wang, B. Zhao, and L. Zou, "Global perception network for salient object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–12, 2022.
[23] G. Li *et al.*, "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.
[24] O. Ali *et al.*, "Implementation of a modified U-Net for medical image segmentation on edge devices," *IEEE Trans. Circuits Syst. II-Express Briefs*, vol. 69, no. 11, pp. 4593–4597, Nov. 2022.
[25] A. Dosovitskiy *et al.*, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. ICLR*, 2021.
[26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, May 2015, pp. 1–14.
[27] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. ECCV*, Sept. 2018, pp. 3–19.
[28] A. Vaswani *et al.*, "Attention is all you need," in *Proc. NeurIPS*, Dec. 2017, pp. 6000–6010.
[29] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. ICLR*, May 2016, pp. 1–13.
[30] D.-P. Fan *et al.*, "Structure-measure: A new way to evaluate foreground maps," in *Proc. IEEE ICCV*, Oct. 2017, pp. 4548–4557.
[31] R. Achanta *et al.*, "Frequency-tuned salient region detection," in *Proc. IEEE CVPR*, Jun. 2009, pp. 1597–1604.
[32] D.-P. Fan *et al.*, "Enhanced-alignment measure for binary foreground map evaluation," in *Proc. IJCAI*, Jul. 2018, pp. 698–704.