ISPRS Journal of Photogrammetry and Remote Sensing 198 (2023) 184–196

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

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

Global–local–global context-aware network for salient object detection in optical remote sensing images

Zhen Bai [a,b], Gongyang Li [a,b,*], Zhi Liu [a,b]

[a] Shanghai Institute for Advanced Communication and Data Science, Shanghai University, Shanghai 200444, China
[b] School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China

ARTICLE INFO

Keywords:
Salient object detection
Optical remote sensing images
Transformer
Dynamic filter
Attention mechanism

ABSTRACT

For the salient object detection in optical remote sensing images (ORSI-SOD), many existing methods are trapped in a local–global mode, *i.e.*, CNN-based encoder binds with a specific global context-aware module, struggling to deal with the challenging ORSIs with complex background and scale-variant objects. To solve this issue, we explore the synergy of the global-context-aware and local-context-aware modeling and construct a preferable global–local–global context-aware network (GLGCNet). In the GLGCNet, a transformer-based encoder is adopted to extract global representations, combining with local-context-aware features gathered from three saliency-up modules for comprehensive saliency modeling, and an edge assignment module is additionally employed to refine the preliminary detection. Specifically, the saliency-up module involves two components, one for global–local context-aware transfer towards pixel-wise dynamic convolution parameters prediction, the other for dynamically local-context aware modeling. The corresponding position-sensitive filter is aware of its previous global-wise focus, thus enhancing the spatial compactness of salient objects and encouraging the feature upsampling achievement for multi-scale feature combinations. The edge assignment module enhances the robustness of preliminary saliency prediction and assigns the semantic attributes of preliminary saliency cues to the shallow-level edge feature to obtain final complete salient objects in a spatially and semantically global manner. Extensive experiments demonstrate that the proposed GLGCNet surpasses 23 state-of-the-art methods on three popular datasets.

1. Introduction

Simulating the human visual attention mechanism of humans during scene-free viewing is termed as visual saliency detection (Wang et al., 2021b; Wang and Shen, 2018). Compared with visual saliency detection, salient object detection (SOD) makes a further step to detect the salient objects (Wang et al., 2020; Li et al., 2021b; Wang et al., 2019a). SOD is taken as a visual saliency analysis process to facilitate a variety of real-world applications, such as video segmentation (Wang et al., 2018c,b), photo cropping (Wang et al., 2019b), semantic segmentation (Wang et al., 2022d, 2021d), and image resize (Wang et al., 2017), *etc*. Recently, the application scene of SOD has expanded from natural scene images (NSIs) to optical remote sensing images (ORSIs) (Yang et al., 2019b; Xu et al., 2021a; Han et al., 2014). ORSI-SOD is a challenging task as the acquisition condition of ORSIs with minimal human intervention is more easily affected by some uncontrollable factors (*e.g.*, terrain shade, illumination intensity, weather, and shooting time). This situation demands global-context-aware observation for the whole scene, and the attribution of the pixel-wise prediction made the local-context-aware modeling is also indispensable for ORSI-SOD. Till now, many popular methods directly draw together the convolution neural networks (CNNs) (LeCun et al., 1989) based encoder with different global-context-aware modules.

Absorbing the advantages from the NSI-SOD (Wang et al., 2022a) and according to the specific characteristics of ORSIs, existing ORSI-SOD methods (Li et al., 2019; Zhang et al., 2021; Zhou et al., 2022a; Cong et al., 2021; Tu et al., 2022; Huang et al., 2021; Li et al., 2022b,a; Huang et al., 2022) achieved impressive detection accuracy with the promotion of CNNs. The popular methods typically build upon a CNN-based encoder, densely connecting multi-scale CNN-based features or designing self-attention mechanism-based modules to capture global context information. However, these methods are confined to the local–global scheme, where the postponed global information capture process may be suboptimal for some challenging scenes, such as the motorway that crosses the entire image in Fig. 1, the motorway cannot be identified owing the methods with local observation at the beginning are easy to be attracted by local regions with strong appearance contrast. It is difficult to be remedied by subsequent global context-aware operations. Different from the above methods, GPNet (Liu et al.,

* Corresponding author at: Shanghai Institute for Advanced Communication and Data Science, Shanghai University, Shanghai 200444, China.
E-mail addresses: bz536476@163.com (Z. Bai), ligongyang@shu.edu.cn (G. Li), liuzhisjtu@163.com (Z. Liu).

https://doi.org/10.1016/j.isprsjprs.2023.03.013
Received 30 July 2022; Received in revised form 25 February 2023; Accepted 10 March 2023

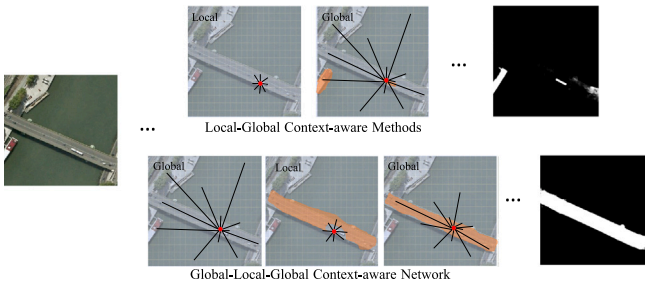

Fig. 1. Existing local–global context-aware methods are difficult to identify the salient object ignored in the local context-aware stage. In contrast, our GLGCNet is fully aware of both local and global contexts, enabling a holistic understanding of the whole ORSI and segmenting complete salient objects.

2022) embeds global-context-aware operation and multi-scale connection into a CNN-based encoder for representative feature extraction. Though available, it lacks the anti-interference capability. How to stimulate the synergy of global-context-aware and local-context-aware modeling to tackle challenging scenes, is under-explored in existing ORSI-SOD methods. To deal with the challenging scenes, we design a Global–Local–Global Context-aware Network (GLGCNet) for ORSI-SOD, building the interdependence between global-context-aware and local-context-aware modeling as shown in Fig. 1. Compared with the CNN-based encoder, the transformer-based encoder with global long-range dependencies (Liu et al., 2021d; Mao et al., 2021; Dosovitskiy et al., 2021), is good at understanding the whole scene content. With the deepening of the network, the resolution of high-level features decreases dramatically, which is not conducive to the accurate detection of some small or irregularly shaped objects. Instead of directly carrying out static local interaction, *i.e.*, static convolution layer, we believe the content-adaptive local interaction among adjacent pixels is beneficial to highlight the attention of these tough objects. Therefore, the global basic features are dynamically enhanced in the local view with attentional focus, which corresponds to the global–local transfer, creating inseparable synergy among the global-context-aware and local-context-aware modeling. Shallow-level features exhibit detailed boundary information, which is indispensable for complete SOD, but also carries some background noises. In GLGCNet, we further pursue complete SOD by selectively enhancing the edge cues of salient objects and propagating the semantic features of detected salient objects to them with a spatially and semantically global propagation. The global–local transfer and the preliminary saliency cues guided propagation build strong interdependence among global-context-aware and local-context-aware modeling, making a fully attentional model, with great adaptability for diverse ORSIs.

In particular, we first extract four-level global basic features from a transformer-based encoder. The saliency-up module adopts the dynamic convolution layer to adaptively achieve local interaction, preserving and refining the structures of salient objects, and also rescales the enhanced features spatially to promote the combination of features of adjacent levels. We deploy saliency-up modules on global features of three levels to progressively restore the resolution of saliency cues for preliminary saliency detection. The edge assignment module is applied to the feature of the first level to establish global interaction between the enhanced low-level feature and the saliency cues for final saliency detection.

Following the above global–local–global scheme, our GLGCNet can accurately and completely capture the salient objects without disturbances in some challenging scenes, which are difficult for the local–global scheme based methods, *i.e.*, SARNet (Huang et al., 2021) and MCCNet (Li et al., 2022b) in Fig. 7. Experimental results shown in Tables 1 and 2 demonstrate that GLGCNet surpasses 23 state-of-the-art across three benchmark datasets, which is consistent with the visual comparison in Fig. 7. Moreover, even compared with the lightweight method, *i.e.*, CorrNet (Li et al., 2022a), our method still has the lowest computational complexity.

Our main contributions are summarized as follows:

- We propose a transformer-based method for SOD in ORSIs, named *Global–Local–Global Context-aware Network* (GLGCNet). Unlike previous methods that follow the local–global scheme, we develop a novel global–local–global scheme in our GLGCNet to stimulate the synergy of global-context-aware and local-context-aware modeling.
- We propose a *Saliency-up Module* to build the global–local context-aware transfer, that is, handling global context-aware features in a local manner using content-adaptive dynamic convolution layers.
- We propose an *Edge Assignment Module* to detect more complete salient objects with sharp boundaries by injecting enhanced shallow-level features into high-level saliency cues in a spatially and semantically global manner.

2. Related work

In this section, we first summarize the works of ORSI-SOD, covering traditional and CNN-based methods, and then briefly describe the development of the vision transformer and the dynamic filter.

2.1. Salient object detection in optical remote sensing images

With different imaging conditions and interference from environmental factors, the ORSI-SOD task is more challenging than the NSI-SOD task (Wang et al., 2019d). Similar to the development of most computer version tasks, the early ORSI-SOD methods were mainly based on some superpixel-wise handcrafted features (*i.e.*, color, texture, intensity, histogram, and orientation) and employed classical machine learning algorithms (*e.g.*, Sparse coding (Zhao et al., 2015), Low-rank matrix decomposition (Liu et al., 2019b), Quaternion fourier transform (Zhang and Yang, 2014), Wavelet transform (Zhang and Zhang, 2017), *etc*) to deal with challenge optical remote sensing scenes.

The rise of CNN promotes the significant performance improvement of the ORSI-SOD task. Due to the lacking of ORSIs data in the early stage, Zhang and Ma (2021) introduced weakly supervised learning into ORSI-SOD. As the deep learning supervision scheme (Liang and Hu, 2015; Wang et al., 2019c; Zhou et al., 2019) works well, the publicly available ORSI-SOD datasets, *i.e.*, ORSSD (Li et al., 2019), EORSSD (Zhang et al., 2021), and ORSI-4199 (Tu et al., 2022) were built. To detect complete salient objects with scale variation, most existing methods followed the multi-scale architecture accompanied by effective strategies for context modeling (Zhou et al., 2022a; Li et al., 2020, 2022b, 2023, 2022c, 2019), for example, constructing a dense and nested structure with multi-resolution inputs (Li et al., 2019), adopting multiple convolution layers with different sizes of kernels and dilation rates (Zhou et al., 2022a; Li et al., 2020, 2022b, 2023, 2022c, 2019) and attention mechanism-based operations to explore foreground prior, edge clue, background cue, and global cues (Zhang et al., 2021; Li et al., 2022b,c,a, 2023), introducing global awareness module into CNN-based encoder (Cong et al., 2021; Liu et al., 2022; Wang et al., 2022c) to extract representative feature, and employing additional edge labels to promote the boundary-aware capability of models (Li et al., 2019; Zhou et al., 2022a).

To sum up, most above methods follow the local–global context-aware scheme which cannot well deal with challenging scenes. Different from these methods, our GLGCNet employs a transformer-based backbone to extract multi-level basic global features and adopts a dynamic filter to adaptively preserve and enhance important local details for preliminary prediction. This progress is reversed to most existing methods, following the global–local scheme. Besides these, we additionally sharpen the boundary of salient objects by globally propagating

the preliminary saliency cues to purified shallow-level features, instead of depending on edge supervision like DAFNet (Zhang et al., 2021), EMFINet (Zhou et al., 2022a), and MJRBM (Tu et al., 2022). Overall, our complete framework follows a distinctive global–local–global scheme to pursue superior performance.

2.2. Vision transformer

The transformer was first presented in the natural language processing (NLP) field, which shows great performance by capturing long-range dependencies among sequence elements. Dosovitskiy et al. (2021) introduced the ability of global interaction into computer vision and first applied a transformer to image classification. Subsequently, lots of improved transformer-based models were proposed in many aspects, *i.e.*, improving the training efficiency (Chen et al., 2021b; Touvron et al., 2021), reducing computation complexity (Liu et al., 2021b; Wang et al., 2022e), and modifying architecture to adapt to various computer vision tasks (Yuan et al., 2021; Zheng et al., 2021; Wang et al., 2021c, 2022e; Zhu et al., 2021). Some pioneering methods were improved to adapt to dense prediction tasks (Wang et al., 2022b; Chen et al., 2021a), for example, replacing the static window with shifted window strategy to interact with cross-window information (Liu et al., 2021b), embedding CNN into multi-head attention block (Wang et al., 2021c, 2022e) to explicitly model local and global context, and stacking CNN and transformer blocks to form a new encoder structure (Chen et al., 2021a). The transformer was also explored in the field of SOD with fully supervised and weakly supervised manners (Liu et al., 2021d; Mao et al., 2021).

Inspired by these excellent works, we adopt PVT-v2 as the backbone of our GLGCNet, anticipating global context information to guide the subsequent process of local details enhancement and global propagation.

2.3. Dynamic filter

Static convolution filters are content-agnostic and are shared across images and pixels, leading to sub-optimal feature learning. Dynamic filter, as opposed to static convolution, is content adaptive and can adjust its parameters according to the input features. Deformable convolution (Dai et al., 2017) reshaped the convolution kernels to adapt to the effective reception field. The early proposed dynamic filters (Chen et al., 2020a; Yang et al., 2019a) adjusted the kernels based on their inputs during inference by linear combining multiple static convolutions (Yang et al., 2019a). Another convolution parameter adaptation is achieved by directly generating the kernel weights from the inputs, embedding with an independent parameter prediction branch (Brun et al., 2022). By stacking this type of dynamic filter units, DyNet (Neubig et al., 2017), SENet (Hu et al., 2020a), and DynamicConv (Chen et al., 2020b) were constructed, but filter with generated parameters was still applied in a convolution manner (spatially shared) (Hou et al., 2021). CARAFE (Wang et al., 2021a) proposed a dynamic layer with a branch to predict a 2D filter for each pixel, with fewer parameters and computation complexity. While the channel-wise shared 2D filters cannot encode channel-specific information.

Our saliency-up module decouples the convolution filter into the channel and spatial domains to perform a pixel-specific filter on the ORSI-SOD. This module is spatial sensitivity, which reassembles dynamic filtered features to realize feature upsampling, adaptively preserving and highlighting the effective local information of salient objects. Notably, this module looks complex but simple and lightweight.

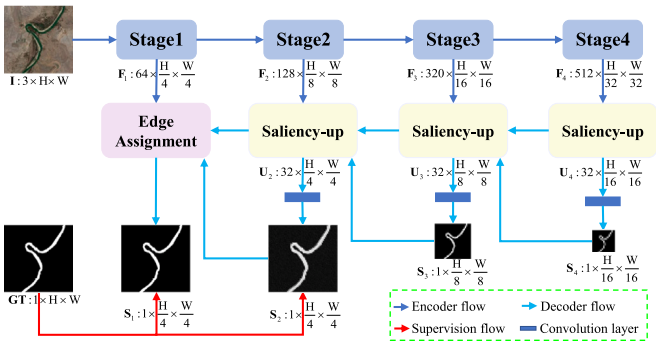

Fig. 2. Architecture overview of the proposed GLGCNet. For a given image, we first utilize PVT-V2-B2 (Wang et al., 2022e) to obtain four-scale features. Then, three relatively high-level features are fed into three saliency-up modules, respectively, generating discriminative features $\mathbf{U}_i$ with resolution amplification. With a convolution, $\mathbf{U}_i$ are compressed to the saliency map $\mathbf{S}_i$. Subsequently, the outputs $\mathbf{U}_2$ and $\mathbf{S}_2$ of first saliency-up module flow into the edge assignment module to propagate the saliency cues to $\mathbf{F}_1$ for final detection, generating the final saliency map $\mathbf{S}_1$. In the training phase, only $\mathbf{S}_1$ and $\mathbf{S}_2$ are supervised by **GT**.

3. Proposed method

3.1. Network overview

As depicted in Fig. 2, our GLGCNet is a U-shape (Ronneberger et al., 2015) structure, is comprised of three key parts: the encoder network, three saliency-up modules, and an edge assignment module.

For the encoder network, we adopt a pyramid vision transformer (PVT) (Wang et al., 2022e) as an encoder to extract long-range dependencies (*i.e.*, global) features as basic features. Then, the saliency-up module is employed to enhance the local representations and achieve feature up-sampling. Three saliency-up modules are connected in series to progressively realize the preliminary saliency prediction. The edge assignment module is in charge of the final salient object detection. And it removes noise and enhances the finer details of detected images through the propagation of preliminary saliency prediction.

Given an image $\mathbf{I} \in \mathbb{R}^{3\times H\times W}$, we first use PVT-V2-B2 (Wang et al., 2022e) to extract four features with four scales, *i.e.*, $\mathbf{F}_i \in \mathbb{R}^{C_i\times\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}}$, where i is the stage index and belongs to $\{1,2,3,4\}$, C_i corresponds to each stage is $\{64,128,320,512\}$, and the head number of the self-attention in ith stage is $\{1,2,5,8\}$. Then, the basic features $\mathbf{F}_i$ of three relatively high-level stages ($i \in \{2,3,4\}$) will be fed into the corresponding saliency-up module to produce up-sampling features $\mathbf{U}_i \in \mathbb{R}^{32\times\frac{H}{2^i}\times\frac{W}{2^i}}$. For each $\mathbf{U}_i$, a convolution layer is used to generate saliency map $\mathbf{S}_i \in \mathbb{R}^{1\times\frac{H}{2^i}\times\frac{W}{2^i}}$. The generated $\mathbf{U}_i$ and $\mathbf{S}_i$ of the above three levels are combined in a top-down pathway. Finally, the proposed edge assignment module transmits saliency cues of $\mathbf{U}_2$ and $\mathbf{S}_2$ to shallow-level feature $\mathbf{F}_1$ and infers complete salient objects. For the network training, we introduce the classic binary cross-entropy (BCE) loss and intersection-over-union (IoU) loss to supervise two outputs, *i.e.*, $\mathbf{S}_1$ and $\mathbf{S}_2$.

3.2. Saliency-up module

With the deepening of the network, the resolution of high-level features decreases dramatically, which is not conducive to the accurate detection of some small or irregularly shaped objects. The extracted global context feature representation $\mathbf{F}_i$ can roughly locate the location of salient objects, but the softmax normalization operation to the quadratic input sequence length, making the pixels belonging to salient objects with individual properties may be obscured. Accordingly, local interaction is needed to aggregate structural information of salient objects. In contrast to directly stacking several layers of static convolution,

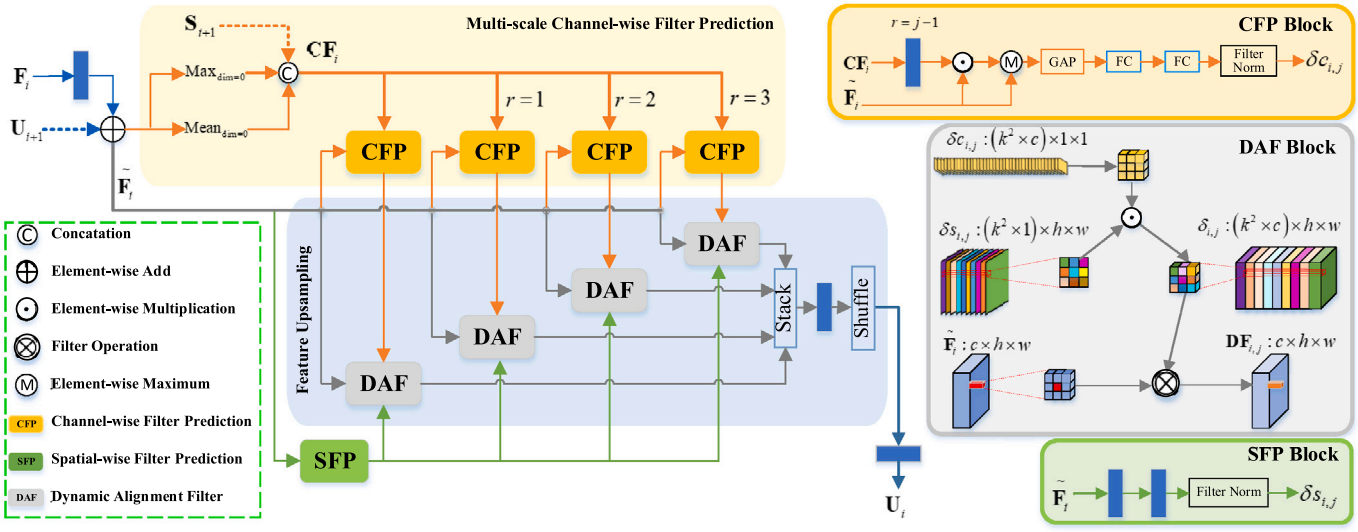

Fig. 3. Illustration of the saliency-up module.

Algorithm 1 Pseudo code of key blocks in the saliency-up module.

```
# CF: spatial attention feature, F: feature to be filtered
# b: batch size, h: height, w: width, r: dilated ratio
# ch: channel number, k: dynamic kernel size
# o: reduction ratio, p: padding
################### initialization ###################
--------------------- CFP --------------------------
conv = nn.Conv2d(3, 1, 3, p, r)
CFP = nn.Sequential(
    nn.AdaptiveAvgPool2d((1,1)),
    nn.Linear(ch, ch*o),
    nn.Relu(True),
    nn.Linear(ch*o, ch*k*k),
    FilterNorm(K, 'channel'))
--------------------- SFP --------------------------
SFP = nn.Sequential(
    nn.Conv2d(ch, ch, 1),
    nn.Conv2d(ch, k*k, 1),
    FilterNorm(K, 'spatial'))
--------------------- DAF --------------------------
unfold = nn.Unfold(k, 1, 1, 1)
#################### forward pass ####################
--------------------- CFP --------------------------
cs = conv(CF)
F = F.max(torch.mul(cs, F))
cp = CFP(F) #b, ch*k*k
--------------------- SFP --------------------------
sp = SFP(F) #b, k*k, h, w
--------------------- DAF --------------------------
cp = cp.view(b, ch, k*k, 1, 1)
sp = sp.unsqueeze(1) #b, 1, k*k, h, w
filter = torch.mul(cp, sp) #b, ch, k*k, h, w
F_unfold = unfold(F) #b, ch*k*k, h, w
FF = F_unfold.view(b, ch, k*k, h, w)
out = torch.mul(FF, filter).sum(dim=2)
return out
```

we consider the content-adaptive and spatial position-sensitive factors to preserve and enhance saliency cues in the local view. This is achieved using a dynamic convolution, which inserts a more diverse and effective attention mechanism into the convolution kernel space, and decouples the convolution filter into the channel and spatial domains to perform a pixel-specific filter on the given feature. We propose the saliency-up module that introduces saliency cues from the higher level to conduct adjacent-level contextual complementary, then serves as the global–local hub, which transfers the given global contextual information into the dynamic kernels required by the local connection operation, *i.e.*, dynamic convolution, to filter itself, adaptively achieving local interaction to enhance its representation capability. This module further realizes feature up-sampling by reassembling the dynamics-filtered features.

As shown in Fig. 3, our saliency-up module is implemented with the global–local context-aware transfer, and feature upsampling. Specifically, for the saliency-up module corresponding to $\mathbf{F}_4$, its input is $\tilde{\mathbf{F}}_4$, which is generated from $\mathbf{F}_4$ by channel reduction. While for the saliency-up module corresponding to $\mathbf{F}_{i\in\{2,3\}}$, its inputs are $\tilde{\mathbf{F}}_i$ and $\mathbf{S}_{i+1}$, respectively, $\tilde{\mathbf{F}}_i$ is the sum of the channel reduced $\mathbf{F}_i$ and $\mathbf{U}_{i+1}$. We present our saliency-up module in a Pytorch-like style on Algorithm 1. In the following, we elaborate on this module from the above two parts.

3.2.1. Global–local context-aware transfer

The global–local context-aware transfer refers to the process of generating dynamic convolution kernels according to the given global basic features, which is comprised of multi-scale channel-wise filter prediction (the convolution filter in the channel domain) and spatial-wise filter prediction (the convolution filter in the spatial domain) blocks.

Multi-scale channel-wise filter prediction. We employ improved spatial-wise attention and channel-wise attention on the channel-wise filter prediction. The generated processes of the spatial-wise attention feature $\mathbf{CF}_i \in \mathbb{R}^{3\times h\times w}$ is generated as follows:

$$\mathbf{CF}_i = \begin{cases} concat\left(P_{max}(\tilde{\mathbf{F}}_i), P_{avg}(\tilde{\mathbf{F}}_i), \mathbf{S}_{i+1}\right), & i = 2,3 \\ concat\left(P_{max}(\tilde{\mathbf{F}}_i), P_{avg}(\tilde{\mathbf{F}}_i)\right), & i = 4, \end{cases} \tag{1}$$

where $concat(\cdot)$, $P_{max}(\cdot)$, and $P_{avg}(\cdot)$ are feature concatenation, channel-wise global max pooling, and channel-wise global average pooling, respectively. Then $\mathbf{CF}_i$ flows into four channel-wise filter prediction (CFP) blocks with different dilation rates. In each CFP block, a dilated convolution layer, an element-wise maximum, a squeeze-and-excitation (SE) operation (Hu et al., 2020a),[1] and a filter normalization operation (Zhou et al., 2021) are applied on $\mathbf{CF}_i$ to generate channel-wise filter weights $\delta c_{i,j} \in \mathbb{R}^{(k^2\times 32)\times 1\times 1}$, where j is the index of the CFP block, $\delta c^n_{i,j} \in \mathbb{R}^{(k^2\times 1)\times 1\times 1}$ is the convolution weight for the nth channel, and k denotes the kernel size of dynamic convolution layer and is set to 3

[1] SE is implemented by a spatial global average pooling (GAP) and two fully connected layers.

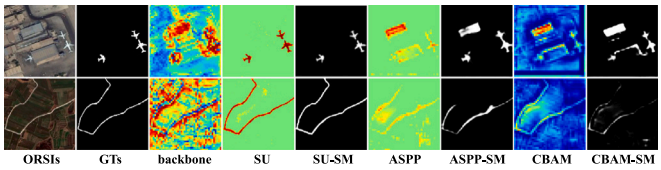

Fig. 4. Local context-aware saliency-up module. F2 is the feature generated by the second stage of PVT-V2-B2. We apply saliency-up, ASPP, and CBAM modules on F2, respectively. SU is the feature generated by the Saliency-up module, ASPP is the feature generated by ASPP, and CBAM is the feature generated by CBAM.

in this paper. This filter is channel-specific but spatial-agnostic. These operations are implemented as follows:

$$\delta c_{i,j} = FN\Big(SE\big(Max(dconv_{3\times3}^{j-1}(\mathbf{CF}_i)\odot\tilde{\mathbf{F}}_i),\tilde{\mathbf{F}}_i\big)\Big), \tag{2}$$

where $Max(\cdot,\cdot)$, $\odot$, $dconv_{3\times3}^{j-1}$, $SE(\cdot)$, and $FN(\cdot)$ respectively denote element-wise maximum, element-wise multiplication, a 3×3 dilated convolution layer with a dilation rate of $j-1$, SE operation, and the filter normalization operation. The filter normalization is implemented by zero-mean normalization measured in terms of standard deviations from the mean.

With multiple dilated convolution layers and the element-wise maximum operation, the saliency-up module can learn desired diverse representations to highlight the obscure pixels of salient objects with variable sizes. And SE operation further pays more attention to the feature maps with distinctive attributes. The filter normalization is adopted to limit the range of the generated filter values, avoiding the gradient vanishing/exploding during training.

Spatial-wise filter prediction. The spatial sensitivity supports the detail-oriented feature enhancement. However, the predicted filter of the CFP block is spatial-agnostic, which is not conducive to subsequent upsampling. To this end, the spatial-wise filter prediction (SFP) block is designed, which only contains two 1×1 convolution layers and a filter normalization operator to generate spatial-wise filter weights $\delta s_{i,j}\in\mathbb{R}^{(k^2\times1)\times h\times w}$ for each pixel of $\tilde{\mathbf{F}}_i$. $\delta s_{i,j}$ adjusts the filter predicted by CFP, *i.e.*, $\delta c_{i,j}$, to adapt to any position:

$$\delta s_{i,j} = FN\Big(\big(conv_{1\times1}(conv_{1\times1}(\tilde{\mathbf{F}}_i))\big)\Big), \tag{3}$$

where $conv_{1\times1}(\cdot)$ is the 1×1 convolution layer. The SFP block endows our saliency-up module with the spatial-specific property in an intuitive way, which enlarges the flexibility of feature recovery and promotes the subsequent feature upsampling.

Since convolution mainly acts on two channels, the number of parameters of linear projection in the above two filter prediction blocks accounts for a large proportion. In total, the number of parameters for a saliency-up module can be even lower than a static convolution layer.

3.2.2. Feature upsampling

This part consists of four dynamic alignment filter (DAF) blocks and a Stack&Group convolution&Shuffle operation (Shi et al., 2016).

Dynamic alignment filter. The DAF block is a combination of unfolding operations and multiplication operations, as presented in Algorithm 1. And it can filter the input feature by unfolding the inputs to fit the filters predicted by CFP and SFP blocks.

Following four DAF blocks, we stack four outputs of these DAF blocks, connecting four values corresponding to each pixel through group convolution with c groups, and shuffling these to achieve feature upsampling, generating $\mathbf{U}_i$ for the subsequent combination between features of adjacent levels.

The upsampling operation undertakes unifying feature scales for multi-level feature combinations, especially for dense prediction tasks. Compared with the representative upsampling operation, *i.e.*, bilinear interpolation, and deconvolution, our upsampling operation is content-adaptive and with less information loss, which is more appropriate for our explored dense prediction task. Notably, the number of dynamic convolutions can be adjusted according to the upsampling rate.

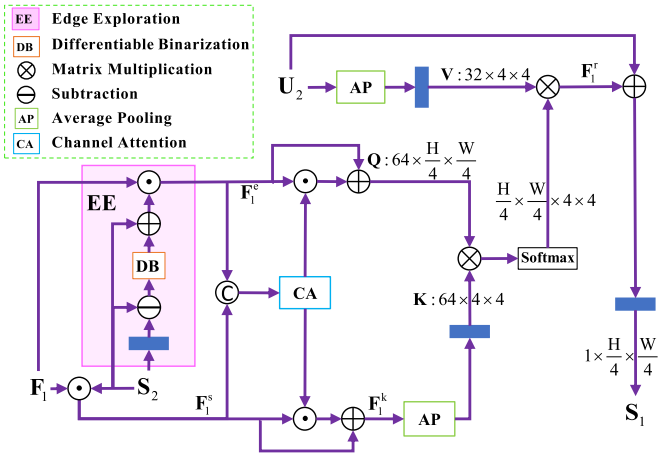

Fig. 5. Illustration of the edge assignment module.

As shown in Fig. 4, we replace the saliency-up module with classic atrous spatial pyramid pooling (ASPP) (Chen et al., 2017) and convolutional block attention module (CBAM) (Woo et al., 2018) respectively. The ASPP based on static convolution with fixed weights only empirically models local connections without the context-aware ability. In contrast, the content-adaptive local connections among adjacent pixels implemented by our saliency-up module are beneficial to selectively recovering the attention of confused pixels in salient objects. This module creates inseparable and effective synergy among the backbone and local-context-aware modeling, which is more suitable for complex ORSIs. From the first image of Fig. 4, the dynamic convolution with adaptive local smoother makes better use of the global-context-aware saliency cues and minimizes adverse effects of non-salient region, is better than directly adopting attention mechanism and static convolution (Han et al., 2022), *i.e.*, CBAM. The reason is that the saliency-up module enhances the feature in a spatial-sensitive local window, not the pixel-wise enhancement adopted by CBAM. As the property of the pixels in salient objects tends to be consistent, the generated saliency maps become sharper.

3.3. Edge assignment module

In the studies of dense prediction tasks, shallow-level features containing rich detail information, such as texture and edges, are widely adopted to complete the segmented objects. Nevertheless, by directly concatenating the preliminary saliency cues with shallow-level features, the details of non-salient objects may confuse the detection, especially in complicated ORSIs. When we obtain an accurate preliminary saliency map, directly multiplying the shallow-level feature with the preliminary saliency map just increases the saliency value of the salient object pixels with details. On the contrary, this strategy results in abnormal segmentation results. The effect of refining the preliminary saliency map is weak. The preliminary saliency cues only require the supplement of the valuable details of salient objects. Accordingly, we construct an edge assignment module to enhance the edge cues of shallow-level features of salient regions with the assistance of preliminary prediction, and model the interaction between the enhanced shallow-level feature and semantic features of the detected salient objects in a global view, as depicted in Fig. 5 to promote the complete saliency detection.

This module works in the mode of non-local connection (Wang et al., 2018a) and resets the components of query, key, and value to

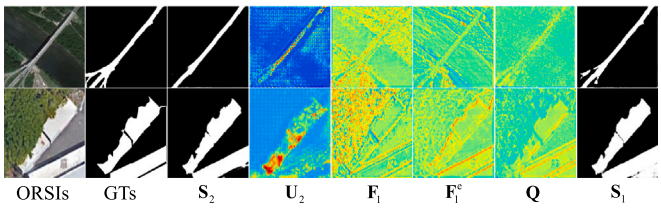

Fig. 6. Visualization of edge assignment module.

implement edge injection on preliminary acquired semantic saliency cues according to the requirements of ORSI-SOD.

Given the saliency-up feature $\mathbf{U}_2$, preliminary saliency map $\mathbf{S}_2$ and the shallow-level feature $\mathbf{F}_1$, we construct the saliency feature with the enhanced edge, shallow-level saliency feature, and semantic saliency feature as query, key, and value respectively. First, we take $\mathbf{S}_2$ as the mask of $\mathbf{F}_1$ to generate the detailed feature $\mathbf{F}_1^s$ of salient objects as follow:

$$\mathbf{F}_1^s = \mathbf{F}_1 \odot \mathbf{S}_2. \tag{4}$$

Simultaneously, we employ a specifically designed edge exploration (EE) block to improve the effectiveness of the preliminary saliency map $\mathbf{S}_2$ and generate edge feature $\mathbf{F}_e$. In this block, we employ a Gaussian convolution and a subtraction operation to amplify the preliminary saliency regions of $\mathbf{S}_2$, and then adopt an improved Sigmoid activation function (Liao et al., 2022) (*i.e.*, differentiable binarization) to activate the pixels covered by amplifying saliency region. The above are formulated in detail as follows:

$$\xi_b(\mathbf{Z}_p) = \frac{1}{1 + e^{-m(\mathbf{Z}_p - \bar{\mathbf{Z}})}}, \tag{5}$$

where $\xi_b(\cdot)$ is the differentiable binarization operation, m indicates the amplifying factor set to 10 empirically, p is the position of pixel in $\mathbf{Z}$, $\bar{\mathbf{Z}}$ is the mean of all pixels in $\mathbf{Z}$. This differentiable binarization position operation helps the edge exploration block to enhance edge pixels from the shallow-level feature, which greatly enhances the robustness and stability of $\mathbf{F}_1^e$.

$$\mathbf{F}_1^e = \mathbf{F}_1 \odot \Big(\xi_b\big(\mathbf{S}_2 \ominus gconv(\mathbf{S}_2)\big) \oplus \mathbf{S}_2\Big), \tag{6}$$

where $gconv(\cdot)$ is a convolution operation with a Gaussian kernel and zero bias to blur the edge of salient objects, the differences produced by the subtraction operation tend to emphasize the edges of salient objects. From Fig. 6, Then, we try to enhance $\mathbf{F}_1^e$ and $\mathbf{F}_1^s$ using the concatenation-channel attention operation, obtaining features $\mathbf{Q}$ and $\mathbf{F}_1^k$:

$$\mathbf{Q} = \mathbf{F}_1^e \oplus \mathbf{F}_1^e \odot CA(concat(\mathbf{F}_1^e, \mathbf{F}_1^s)), \tag{7}$$

$$\mathbf{F}_1^k = \mathbf{F}_1^s \oplus \mathbf{F}_1^e \odot CA(concat(\mathbf{F}_1^e, \mathbf{F}_1^s)), \tag{8}$$

where $CA(\cdot)$ is a channel attention operation, $\mathbf{Q}$ is defined as query. To decrease the computation complexity, we apply average pooling and convolution operations on $\mathbf{F}_1^k$ and $\mathbf{U}_2$ to obtain $\mathbf{K}$ and value $\mathbf{V}$, respectively:

$$\mathbf{K} = conv_{1\times1}\big(AP(\mathbf{F}_1^e)\big), \tag{9}$$

$$\mathbf{V} = conv_{1\times1}\big(AP(\mathbf{U}_2)\big), \tag{10}$$

where $\mathbf{K}$, $\mathbf{V}$, and $AP(\cdot)$ are key, value, and average pooling, respectively.

The key can be regarded as the detail feature of salient objects, and value is the semantic feature of salient objects. After matrix multiplication of query and key, the relation of edge pixels and salient regions are connected, which is defined as $\mathbf{M}$:

$$\mathbf{M} = \zeta(\mathbf{Q}^T \otimes \mathbf{K}), \tag{11}$$

$$\mathbf{F}_1^r = \mathbf{M}^T \otimes \mathbf{V}, \tag{12}$$

where ζ and $\otimes$ denote softmax layer and matrix multiplication, respectively. Then we project the acquired $\mathbf{M}$ onto the semantic level, *i.e.*, aggregating pixels with similar semantic to one vertex to generate $\mathbf{F}_1^r$ for the final saliency map is $\mathbf{S}_1$:

$$\mathbf{S}_1 = conv_{1\times1}(\mathbf{F}_1^r \oplus \mathbf{U}_2). \tag{13}$$

As shown $\mathbf{S}_2$ in Fig. 6, considering the inaccurate of the preliminary saliency map, *i.e.*, the detected salient objects are incomplete, the road forks in the first image are not detected. The EE block expands the saliency region to extract valuable shallow-level edge features, *e.g.*, $\mathbf{F}_1^e$ in Fig. 6. The following channel-wise enhancement operation further minimizes the detractors from the background, *i.e.*, $\mathbf{Q}$. By connecting detail-semantic in a spatially global view, EA accurately assigns the semantic attributes of salient objects to finer detailed features to improve the completeness of salient objects. As shown in Fig. 5, the Flops of this module is 0.0096G, which is lower than the classic self-attention operation. The Flops of the series operation of static convolution layer and concatenation is 0.1368G. Our proposed module makes a good trade-off between computational complexity and effectiveness.

3.4. Comprehensive loss function

As shown in Fig. 2, similar to some existing SOD methods (Qin et al., 2019; Li et al., 2021a; Zhou et al., 2022a), we combine the classic pixel-level BCE loss with map-level IoU loss as our loss function. A comprehensive loss function $\mathbb{L}_i$ is constructed to supervise the predicted saliency map. The resolutions of the generated saliency maps and ground truth are all resized to 352 × 352. This can be formulated as:

$$\mathbb{L}_i = \ell_{bce}(\mathbf{S}_i, \mathbf{G}) + \ell_{iou}(\mathbf{S}_i, \mathbf{G}), \tag{14}$$

where $\mathbf{G} \in \{0,1\}^{352\times352}$ is the ground truth, and $\mathbf{S}_i \in [0,1]^{352\times352} (i = 1, 2)$ is the predicted saliency map, $\ell_{bce}(\cdot)$ is BCE loss, and $\ell_{iou}(\cdot)$ is IoU loss. The total loss $\mathbb{L}_{\text{total}}$ of our GLGCNet in the training phase is the sum of $\mathbb{L}_1$ and $\mathbb{L}_2$. This loss function is enough to urge our GLGCNet to adapt to a variety of complicated ORSIs.

4. Experiments

4.1. Experimental protocol

4.1.1. Datasets

We conduct our experiments on three ORSI datasets (see Table 1), including EORSSD, ORSSD, and ORSI-4199. The ORSSD dataset (Li et al., 2019) contains 800 ORSIs, where 600 images are for training and 200 images for testing. The EORSSD dataset (Zhang et al., 2021) is an extended version of ORSSD and contains 1400 training images and 600 testing images. The ORSI-4199 dataset (Tu et al., 2022) is the biggest and most challenging ORSI-SOD dataset, where 2000 images for training and 2199 images for testing. These three datasets cover various salient objects, *i.e.*, airplane, ship, car, river, pond, bridge, stadium, beach, etc, and the scenes with cluttered backgrounds and scatter distribution are more challenging than NSIs.

4.1.2. Implementation details

The adopted backbone is PVT-V2-B2 pretrained by ImageNet-1k (Russakovsky et al., 2015). The experiments are implemented on the platform of Pytorch (Paszke et al., 2019) by adapting an NVIDIA GTX 2080Ti GPU (11G memory). During training, except for the parameters of the backbone, the additional parameters in the proposed GLGCNet are initialized with the random normal distribution, Adam (Kingma and Ba, 2015) is taken as the optimizer to train GLGCNet, and respectively set the learning rate and weight decay to 10^{-5} and 10^{-4}. All imported ORSIs are resized into 352 × 352. For each training iteration of the

training stage, we set the number of a batch to 10. For the EORSSD dataset (Zhang et al., 2021), we train our GLGCNet with 11,200 augmented pairs of ORSI and GT for 38 epochs. For the ORSSD dataset (Li et al., 2019), the proposed GLGCNet is trained with 4800 augmented pairs for 45 epochs. While for the ORSI-4199 dataset (Tu et al., 2022), the proposed GLGCNet is trained with 16,000 augmented pairs for 38 epochs.

4.1.3. Evaluation metrics

We use S-measure (Fan et al., 2017) ($S_\alpha, \alpha = 0.5$), maximum F-measure (Achanta et al., 2009) ($F_\beta^{\max}, \beta^2 = 0.3$), mean F-measure ($F_\beta^{\text{mean}}$), adaptive F-measure ($F_\beta^{\text{adp}}$), maximum E-measure (Fan et al., 2018) ($E_\xi^{\max}$), mean E-measure (E_ξ^{mean}), adaptive E-measure (E_ξ^{adp}), and mean absolute error (MAE, $\mathcal{M}$) to evaluate the performance of our proposed method and all compared methods.

Additionally, we binarize the saliency maps with thresholds ranging from 0 to 255 and form Precision-Recall curves for the relative state-of-the-art methods by connecting the generated pairs of precision and recall scores.

4.2. Comparison with state-of-the-art methods

To evaluate the effectiveness of the proposed model, 23 state-of-the-art models are adopted for comparison, including DSS (Hou et al., 2017), RADF (Hu et al., 2018), R3Net (Deng et al., 2018), PoolNet (Liu et al., 2019a), EGNet (Zhao et al., 2019), GCPA (Chen et al., 2020c), MINet (Pang et al., 2020), ITSD (Zhou et al., 2020), GateNet (Zhao et al., 2020), SUCA (Li et al., 2021c), PA-KRN (Xu et al., 2021b), VST (Liu et al., 2021d), LVNet (Li et al., 2019), DAFNet (Zhang et al., 2021), MJRBM (Tu et al., 2022), SARNet (Huang et al., 2021), EMFINet (Zhou et al., 2022a), ERPNet (Zhou et al., 2022b), ACCoNet (Li et al., 2022c), MCCNet (Li et al., 2022b), CorrNet (Li et al., 2022a), GPNet (Liu et al., 2022), and HFANet (Wang et al., 2022c). For the methods with the released source code, Parameters, FLOPs and Speed are provided. For a fair comparison, we use either the implementations with recommended parameter settings or saliency maps provided by the authors.

4.2.1. Quantitative comparison

In Table 1, we report the quantitative comparison results of our method and all compared methods on three ORSI-SOD datasets with respect to S_α, $F_\beta^{\max}$, F_β^{mean}, F_β^{adp}, $E_\xi^{\max}$, E_ξ^{mean}, E_ξ^{adp}, $\mathcal{M}$, and measure 12 competitive SOD methods and our GLGCNet in terms of the PR curves, shown in Fig. 8.

Our method performs excellently on the tested two datasets as reported in Table 1, GLGCNet surpasses all compared methods on almost all evaluated metrics. On the EORSSD dataset, the $E_\xi^{\max}$ of our method lags behind that of DAFNet, but other evaluated scores are much higher than the indicators of DAFNet (*e.g.*, S_α: 0.9375 (Ours) *vs.* 0.9166 (DAFNet), F_β^{adp}: 0.8499s (Ours) *vs.* 0.6427 (DAFNet), E_β^{adp}: 0.9701 (Ours) *vs.* 0.8446 (DAFNet), $\mathcal{M}$: 0.0055 (Ours) *vs.* 0.0060 (DAFNet)), and on ORSSD dataset, only F_β^{mean} of GLGCNet is slightly lower than that of MCCNet. Besides these, in Fig. 8, our method is represented as the outermost red line, which is superior to the other compared methods. This is consistent with the remarkable quantitative results of our method on three datasets in Tables 1 and 2.

Even though the CNN-based NSI-SOD methods are retrained on ORSIs, most CNN-based NSI-SOD methods are generally inferior to the specialized ORSI-SOD methods, which illustrates the necessity of proposing specialized solutions for the ORSI-SOD.

We measure the computational complexity in terms of inference speed (without I/O time), network parameters, and FLOPs, which are captured from the available public ORSI-SOD benchmarks (Li et al., 2019; Zhang et al., 2021) and our retraining, and report them in Table 1. Compared with state-of-art methods (including NSI-SOD and ORSI-SOD methods), the inference speed of our method is at the midstream level (20~30 fps), and network parameters and smaller FLOPs of our method rank at the leading level. The low computational complexity of our GLGCNet benefits from the fact that our key modules are applied to compressed basic features with only 32 channels. The parameters and FLOPs of GPNet (Liu et al., 2022) rank second, but the performance of this method is far inferior to our GLGCNet. The poor performance results from the ignorance of the gap between the global-context-aware and local-context-aware. In our GLGCNet, the gap is considered in the saliency-up module and filled by the global–local transfer. Compared with the lightweight method CorrNet which focuses on simplifying the underlying implementation to decrease computation complexity, the inference speed of our model is slower than that of the CorrNet model due to the unfold operation of the saliency-up module, but the detection performance exceeds that of this model. From Tables 1 and 2, the transformer-based models, *i.e.*, VST (Liu et al., 2021d) and HAFNet (Wang et al., 2022c) require more parameters and computation, and they also achieve satisfactory performance. VST performs better on the ORSI-4199 dataset with more training data, which also exposes the performance limitations of VST on small datasets. Our method with great adaptability is friendly to small datasets. From the above quantitative comparison and computational complexity comparison, our model makes a good trade-off between performance and computational complexity, which is effective and efficient.

4.2.2. Visual comparison

As shown in Fig. 7, we post some representative scenes of ORSIs, including tiny salient objects (airplane and car), narrow salient objects with complex geometry (river, pool, and dyke), big salient objects (building and motorway), the complex scene with confusing background (shadow), and complex scene with interference (fog). The three CNN-based ORSI-SOD methods perform certain competitiveness in several sporadic scenes, but cannot deal with challenging ORSIs.

The transformer-based methods, *i.e.*, VST and HFANet, lack the capabilities of details capture and anti-interference. These are remedied by our designed edge assignment and saliency-up modules. Concretely, for the first and third scenes, the two traditional methods perform worst, the tiny salient objects can be located by most of the compared methods. Compared with these methods, the salient objects detected by our method have finer boundaries without noise, which is profited from the adopted edge assignment module. Moreover, for the detection of the salient objects with complex topology in the seventh and eighth lines, our GLGCNet with global–local context-aware transfer of saliency-up modules performs excellently in competition.

4.3. Ablation studies

To gain insight into our key components, we do extensive ablation experiments on ORSSD and EORSSD datasets to investigate their effectiveness of these, including the effectiveness of the global–local–global scheme, the design rationality of the saliency-up module, and the effectiveness of the edge assignment module. For fair experiments, we rigorously retrain each variant with the same settings as the original GLGCNet.

4.3.1. The effectiveness of key components

To evaluate the effectiveness of the global–local–global scheme and our key modules, we quantitatively measure the performance of GLGCNet with flexible backbones, the performance improvements brought by the proposed modules on flexible backbones, and the indispensability of the saliency-up and edge assignment modules.

The effectiveness of global–local–global scheme. The backbone of the proposed GLGCNet is flexible and replaceable. To demonstrate that, we replace the originally adopted PVT-V2-B2 with four common used backbones, *i.e.*, VGG16 (Simonyan and Zisserman, 2014), ResNet50 (Simonyan and Zisserman, 2014), tiny Swin transformer (Liu et al., 2021b), and PVT-V2-B3. Notably, on account of the VGG16

Table 1
Benchmarking results of 23 leading SOD methods on EORSSD and ORSSD datasets. The comparison methods are 11 CNN-based NSI-SOD methods (C.N.), 1 transformer-based NSI-SOD method (T.N.), 9 CNN-based ORSI-SOD methods (C.R.), 1 hybrid encoder-based ORSI-SOD method (H.O.), and 1 transformer-based ORSI-SOD method (T.O.). ↑ & ↓ denote that larger and smaller are better, respectively. The top three results are marked in **red**, **blue** and **green**, respectively.

Methods	Type	Speed	#Param	FLOPs	EORSSD (Zhang et al., 2021)								ORSSD (Li et al., 2019)							
		(fps) ↑	(M) ↓	(G) ↓	S_α ↑	$F_\beta^{\max}$ ↑	F_β^{mean} ↑	F_β^{adp} ↑	$E_\xi^{\max}$ ↑	E_ξ^{mean} ↑	E_ξ^{adp} ↑	$\mathcal{M}$ ↓	S_α ↑	$F_\beta^{\max}$ ↑	F_β^{mean} ↑	F_β^{adp} ↑	$E_\xi^{\max}$ ↑	E_ξ^{mean} ↑	E_ξ^{adp} ↑	$\mathcal{M}$ ↓
DSS_{17} (Hou et al., 2017)	C.N.	8	62.23	114.6	.7868	.6849	.5801	.4597	.9186	.7631	.6933	.0186	.8262	.7467	.6962	.6206	.8860	.8362	.8085	.0363
$RADF_{18}$ (Hu et al., 2018)	C.N.	7	62.54	214.2	.8179	.7446	.6582	.4933	.9140	.8567	.7162	.0168	.8259	.7619	.6856	.5730	.9130	.8298	.7678	.0382
$R3Net_{18}$ (Deng et al., 2018)	C.N.	2	56.16	47.5	.8184	.7498	.6302	.4165	.9483	.8294	.6462	.0171	.8141	.7456	.7383	.7379	.8913	.8681	.8887	.0399
$PoolNet_{19}$ (Liu et al., 2019a)	C.N.	25	53.63	123.4	.8207	.7545	.6406	.4611	.9292	.8193	.6836	.0210	.8403	.7706	.6999	.6166	.9343	.8650	.8124	.0358
$EGNet_{19}$ (Zhao et al., 2019)	C.N.	9	108.07	291.9	.8601	.7880	.6967	.5379	.9570	.8775	.7566	.0110	.8721	.8332	.7500	.6452	.9731	.9013	.8226	.0216
$GCPA_{20}$ (Chen et al., 2020c)	C.N.	23	67.06	54.3	.8869	.8347	.7905	.6723	.9524	.9167	.8647	.0102	.9026	.8687	.8433	.7861	.9509	.9341	.9205	.0168
$MINet_{20}$ (Pang et al., 2020)	C.N.	12	47.56	146.3	.9040	.8344	.8174	.7705	.9442	.9346	.9243	.0093	.9040	.8761	.8574	.8251	.9545	.9454	.9423	.0144
$ITSD_{20}$ (Zhou et al., 2020)	C.N.	16	**17.08**	54.5	.9050	.8523	.8221	.7421	.9556	.9407	.9103	.0106	.9050	.8735	.8502	.8068	.9601	.9482	.9335	.0165
$GateNet_{20}$ (Zhao et al., 2020)	C.N.	25	100.02	108.3	.9114	.8566	.8228	.7109	.9610	.9385	.8909	.0095	.9186	.8871	.8679	.8229	.9664	.9538	.9428	.0137
$SUCA_{21}$ (Li et al., 2021c)	C.N.	24	117.71	56.4	.8988	.8229	.7949	.7260	.9520	.9277	.9082	.0097	.8989	.8484	.8237	.7748	.9584	.9400	.9194	.0145
$PA\text{-}KRN_{21}$ (Xu et al., 2021b)	C.N.	16	141.06	617.7	.9192	.8639	.8358	.7993	.9616	.9536	.9416	.0104	.9239	.8890	.8727	.8548	.9680	.9620	.9579	.0139
VST_{21} (Liu et al., 2021d)	T.N.	23	44.03	23.2	.9208	.8716	.8263	.7089	.9743	.9442	.8941	.0067	.9365	.9095	.8817	.8262	**.9810**	.9621	.9466	.0094
$LVNet_{19}$ (Li et al., 2019)	C.O.	1.4	-	-	.8630	.7794	.7328	.6284	.9254	.8801	.8445	.0146	.8815	.8263	.7995	.7506	.9456	.9259	.9195	.0207
$DAFNet_{21}$ (Zhang et al., 2021)	C.O.	26	29.35	68.5	.9166	.8614	.7845	.6427	**.9861**	.9291	.8446	**.0060**	.9191	.8928	.8511	.7876	.9771	.9539	.9360	.0113
$MJRBM_{21}$ (Tu et al., 2022)	C.O.	32	43.54	95.7	.9197	.8656	.8239	.7066	.9646	.9350	.8897	.0099	.9204	.8842	.8566	.8022	.9623	.9415	.9328	.0163
$SARNet_{21}$ (Huang et al., 2021)	C.O.	47	25.91	129.7	.9240	.8719	.8541	.8304	.9620	.9555	.9536	.0099	.9134	.8850	.8619	.8512	.9557	.9477	.9464	.0187
$EMFINet_{21}$ (Zhou et al., 2022a)	C.O.	25	107.26	480.9	.9290	.8720	.8486	.7984	.9711	.9604	.9501	.0084	.9366	.9002	.8856	.8617	.9737	.9671	.9663	.0109
$ERPNet_{22}$ (Zhou et al., 2022b)	C.O.	50	56.48	87.2	.9210	.8632	.8304	.7554	.9603	.9401	.9228	.0089	.9254	.8974	.8745	.8356	.9710	.9566	.9520	.0135
$ACCoNet_{22}$ (Li et al., 2022c)	C.O.	**81**	102.55	180.0	.9290	.8837	.8552	.7969	.9727	.9653	.9450	.0074	**.9437**	**.9149**	.8971	.8806	.9796	**.9754**	.9721	**.0088**
$MCCNet_{22}$ (Li et al., 2022b)	C.O.	**95**	67.65	112.8	**.9327**	**.8904**	.8604	.8137	**.9755**	**.9685**	.9538	**.0066**	**.9437**	**.9155**	**.9054**	**.8957**	**.9800**	**.9758**	**.9735**	**.0087**
$CorrNet_{22}$ (Li et al., 2022a)	C.O.	**100**	**4.09**	**21.1**	.9289	.8778	**.8620**	**.8311**	.9696	.9646	**.9593**	.0083	.9380	.9129	**.9002**	**.8875**	.9790	.9746	.9721	.0098
$GPNet_{22}$ (Liu et al., 2022)	H.O.	47	25.95	**13.3**	.9233	.8687	.8447	.8132	.9672	.9617	.8132	.0085	.9185	.8829	.8683	.8545	.9638	.9590	.9581	.0125
$HFANet_{22}$ (Wang et al., 2022c)	T.O.	26	60.53	68.3	**.9380**	**.8876**	**.8681**	**.8365**	.9740	**.9679**	**.9644**	.0070	.9399	.9112	.8981	.8819	.9770	.9712	**.9722**	.0092
Ours	T.O.	21	**25.15**	**9.8**	**.9375**	**.8924**	**.8714**	**.8499**	**.9803**	**.9757**	**.9701**	**.0055**	**.9488**	**.9236**	**.9054**	**.8931**	**.9864**	**.9820**	**.9800**	**.0071**

Table 2
Quantitative comparisons with state-of-the-art NSI-SOD and ORSI-SOD methods on the ORSI-4199 dataset. We mark the top three results in **red**, **blue**, and **green** respectively.

Methods	Type	ORSI-4199 (Tu et al., 2022)							
		S_α ↑	$F_\beta^{\max}$ ↑	F_β^{mean} ↑	F_β^{adp} ↑	$E_\xi^{\max}$ ↑	E_ξ^{mean} ↑	E_ξ^{adp} ↑	$\mathcal{M}$ ↓
$R3Net_{18}$ (Deng et al., 2018)	C.N.	.8142	.7847	.7790	.7776	.8880	.8722	.8645	.0401
$PoolNet_{19}$ (Liu et al., 2019a)	C.N.	.8271	.8010	.7779	.7382	.8964	.8676	.8531	.0541
$EGNet_{19}$ (Zhao et al., 2019)	C.N.	.8464	.8267	.8041	.7650	.9161	.8947	.8620	.0440
$BASNet_{19}$ (Qin et al., 2019)	C.N.	.8341	.8157	.8042	.7810	.9069	.8881	.8882	.0454
$MINet_{20}$ (Pang et al., 2020)	C.N.	.8665	.8531	.8457	.8364	.9297	.9231	.9077	.0344
$GateNet_{20}$ (Zhao et al., 2020)	C.N.	.8680	.8626	.8414	.7946	.9369	.9199	.8816	.0357
$SAMNet_{21}$ (Liu et al., 2021c)	C.N.	.8409	.8249	.8029	.7744	.9186	.8938	.8781	.0432
$HVPNet_{21}$ (Liu et al., 2021a)	C.N.	.8471	.8295	.8041	.7652	.9201	.8956	.8687	.0419
$ENFNet_{21}$ (Tu et al., 2021)	C.N.	.7766	.7285	.7177	.7271	.8370	.8107	.8235	.0608
$SUCA_{21}$ (Li et al., 2021c)	C.N.	**.8794**	.8692	.8590	.8415	**.9438**	**.9356**	**.9186**	**.0304**
$PA\text{-}KRN_{21}$ (Xu et al., 2021b)	C.N.	.8491	.8415	.8324	.8200	.9280	.9168	.9063	.0382
VST_{21} (Liu et al., 2021d)	T.N.	**.8790**	**.8717**	.8524	.7947	**.9481**	**.9348**	.8997	**.0281**
$DAFNet_{21}$ (Zhang et al., 2021)	C.O.	.8552	.8458	.8261	.7819	.9220	.9007	.8905	.0396
$MJRBM_{22}$ (Tu et al., 2022)	C.O.	.8593	.8493	.8309	.7995	.9311	.9102	.8891	.0374
$EMFINet_{22}$ (Zhou et al., 2022a)	C.O.	.8675	.8584	.8479	.8186	.9340	.9257	.9136	.0330
$ERPNet_{22}$ (Zhou et al., 2022b)	C.O.	.8670	.8553	.8374	.8024	.9290	.9149	.9024	.0357
$ACCoNet_{22}$ (Li et al., 2022c)	C.O.	.8775	.8686	.8620	**.8581**	.9412	.9342	.9167	.0314
$CorrNet_{22}$ (Li et al., 2022a)	C.O.	.8623	.8560	.8513	.8534	.9330	.9206	.9142	.0366
$MCCNet_{22}$ (Li et al., 2022b)	C.O.	.8746	.8690	**.8630**	**.8592**	.9413	**.9348**	.9182	.0316
$GPNet_{22}$ (Liu et al., 2022)	H.O.	.8573	.8450	.8396	.8371	.9263	.9184	.9084	.0384
$HAFNet_{22}$ (Wang et al., 2022c)	T.O.	.8767	**.8700**	**.8624**	.8323	.9431	.9336	**.9191**	.0314
Ours	T.O.	**.8839**	**.8808**	**.8712**	**.8672**	**.9508**	**.9469**	**.9245**	**.0274**

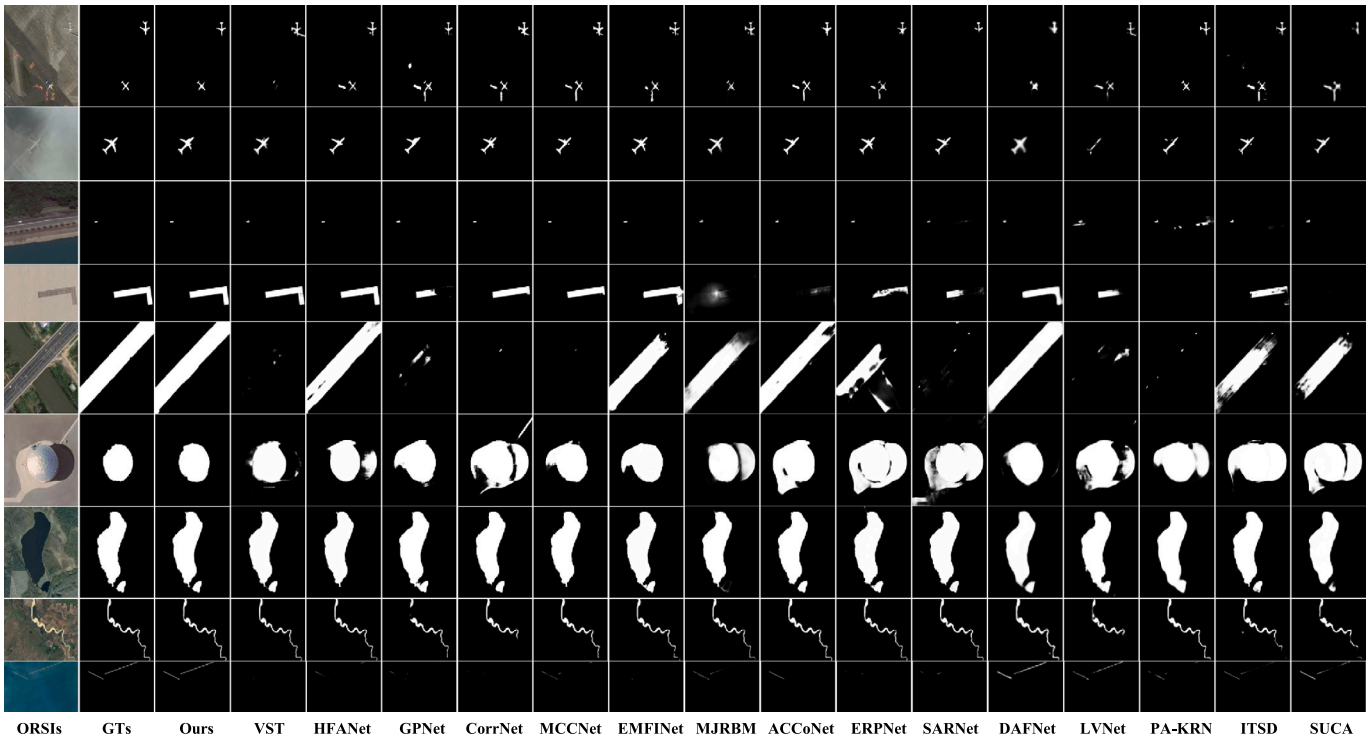

Fig. 7. Visual comparisons with 15 representative state-of-the-art methods, including 1 transformer-based NSI-SOD method (VST Liu et al., 2021d), 1 transformer-based ORSI-SOD method (HFANet Wang et al., 2022c), 1 hybrid encoder-based ORSI-SOD methods (GPNet Liu et al., 2022), 10 CNN-based ORSI-SOD methods (CorrNet Li et al., 2022a, GPNet Liu et al., 2022, MCCNet Li et al., 2022b, EMFINet Zhou et al., 2022a, MJRBM Tu et al., 2022, ACCoNet Li et al., 2022c, ERPNet Zhou et al., 2022b, SARNet Huang et al., 2021, DAFNet Zhang et al., 2021, and LVNet Li et al., 2019), 3 CNN-based NSI-SOD methods (PA-KRN Xu et al., 2021b, ITSD Zhou et al., 2020, and SUCA Li et al., 2021c) on various scenes.

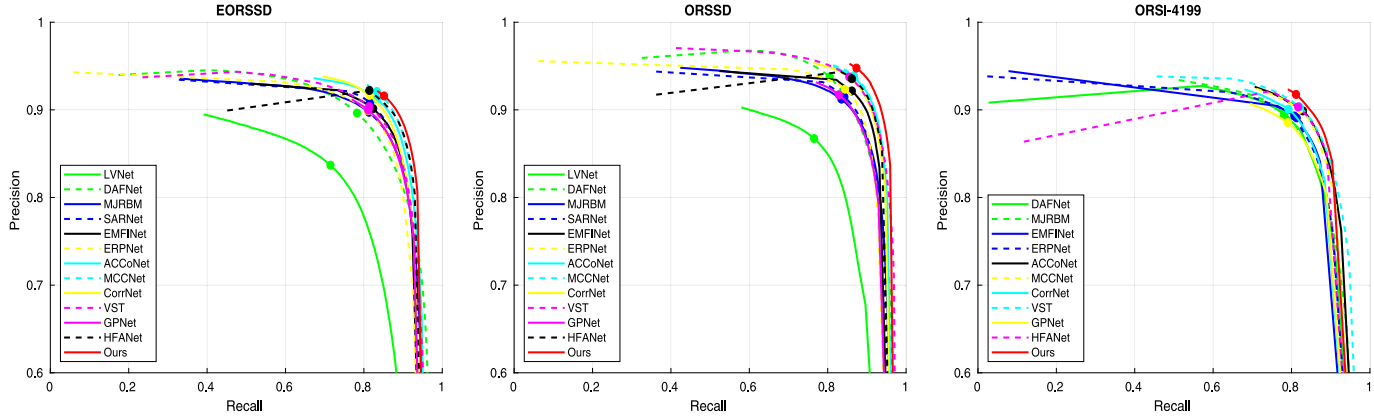

Fig. 8. Quantitative comparison in terms of PR curve on three datasets for ORSI-SOD.

having five feature extraction blocks, the variant is embedded with four saliency-up modules. We retrain these four variants and provide the quantitative results in Table 3. The VGG16-based variant and ResNet50-based variants with CNN-based backbone belong to local–global scheme-based methods, which perform inferior to other transformer-based variants, and with four saliency-up modules, the VGG16-based variant cost more consumption with the highest Flops. With the improvement of basic feature extraction ability, the performance of the variant with PVT-V2-B3 has indeed slightly improved to a certain extent. But this variant has more FLOPs consumption and parameters than the original GLGCNet. When replacing the backbone with the tiny Swin-Transformer, the performance is competitive with the original network. These above all indicate that the extraction of global basic features does promote the optical ORSI-SOD task. And compared with the normal local–global scheme, the adopted global–local–global scheme is more adaptable to this task.

The performance improvement brought by the proposed modules. Additionally, we construct four types of backbone-based UNet by removing our designed modules. The performance improvement brought by our proposed modules can be observed by comparing the pure UNets and corresponding variants (*e.g.*, VGG16-UNet→VGG16-Ours: 4.27%/5.27%/5.39% on EORSSD/ORSSD/ORSI-4199 in S_α). Besides the performance improvements, the lower computational complexity and fewer network parameters also demonstrate the superiority of the saliency-up and edge assignment modules. From Table 3, the performance improvement of CNN-based variants is more obvious than that of transformer-based variants, the reason is that the closer to the upper-performance limit, the harder it is to improve. For the PVTv2-B3 backbone with the strongest feature representative ability,

Table 3
Performance of GLGCNet with flexible backbones.

Backbones	#Param	FLOPs	EORSSD			ORSSD			ORSI-4199		
	(M) ↓	(G) ↓	S_α ↑	F_β^{adp} ↑	$\mathcal{M}$ ↓	S_α ↑	F_β^{adp} ↑	$\mathcal{M}$ ↓	S_α ↑	F_β^{adp} ↑	$\mathcal{M}$ ↓
VGG16-UNet	15.21	23.0	.8819	.8075	.0126	.8765	.8194	.0228	.8248	.7953	.0500
VGG16-Ours	**15.20**	22.8	.9246	.8279	.0089	.9292	.8551	.0103	.8707	.8521	.0309
ResNet50-UNet	24.70	10.9	.8864	.7466	.0114	.9001	.8277	.0146	.8307	.8311	.0403
ResNet50-Ours	24.68	10.7	.9286	.8321	.0079	.9312	.8525	.0097	.8737	.8508	.0303
SwinT-UNet	27.95	11.4	.9060	.7663	.0103	.9185	.8403	.0128	.8259	.8020	.0387
SwinT-Ours	27.94	11.2	.9315	.8474	.0067	.9407	.8952	.0091	.8784	.8559	.0298
PVTv2-B3-UNet	45.03	17.1	.9303	.8315	.0074	.9319	.8578	.0104	.8705	.8433	.0323
PVTv2-B3-Ours	45.02	16.9	**.9405**	.8477	**.0053**	.9470	**.8999**	.0073	**.8875**	.8657	**.0259**
PVTv2-B2-UNet	25.17	10.1	.9223	.8289	.0089	.9309	.8385	.0118	.8650	.8393	.0336
Ours	25.15	**9.8**	.9375	**.8499**	.0055	**.9488**	.8931	**.0071**	.8839	**.8672**	.0274

Table 4
Performance of GLGCNet with two proposed modules. The baseline is a U-Net with PVT-V2-B2 as the backbone, EA represents the edge assignment module, and SU is the Saliency-up module.

Model	EORSSD			ORSSD			ORSI-4199		
	S_α ↑	F_β^{adp} ↑	$\mathcal{M}$ ↓	S_α ↑	F_β^{adp} ↑	$\mathcal{M}$ ↓	S_α ↑	F_β^{adp} ↑	$\mathcal{M}$ ↓
Baseline	.9223	.8289	.0089	.9309	.8385	.0118	.8650	.8493	.0316
w/o SU	.9293	.8324	.0065	.9329	.8789	.0092	.8698	.8421	.0312
w/ 4SU	.9319	.8304	.0081	.9375	.8652	.0133	.8699	.8532	.0302
w/o EA	.9317	.8388	.0063	.9387	**.8938**	.0089	.8746	.8563	.0290
Ours	**.9375**	**.8499**	.0055	**.9488**	.8931	**.0071**	**.8839**	**.8672**	**.0274**

Table 5
Performance of different variants to our saliency-up module. CFP corresponds to the channel-wise filter prediction block, MC is the series of dilated convolution and the element-wise maximum comparison operation in the CFP block, SFP indicates the spatial filter prediction block, SGS is the stack&group convolution&shuffle operation.

No.	CFP	MC	SFP	SGS	ASPP	CBAM	EORSSD			ORSSD			ORSI-4199		
							S_α ↑	F_β^{adp} ↑	$\mathcal{M}$ ↓	S_α ↑	F_β^{adp} ↑	$\mathcal{M}$ ↓	S_α ↑	F_β^{adp} ↑	$\mathcal{M}$ ↓
0	✓	✓	✓	✓			**.9375**	.8499	**.0055**	**.9488**	**.8931**	**.0071**	**.8839**	**.8672**	**.0274**
1	✓	✓		✓			.9339	.8496	.0059	.9479	.8922	.0080	.8779	.8655	.0295
2	✓			✓			.9359	**.8504**	.0061	.9437	.8874	.0079	.8773	.8649	.0310
3			✓	✓			.9327	.8364	.0063	.9387	.8856	.0086	.8750	.8522	.0296
4	✓	✓	✓				.9324	.8435	.0057	.9441	.8828	.0092	.8801	.8657	.0290
5					✓		.9362	.8358	.0068	.9389	.8707	.0102	.8825	.8429	.0288
6						✓	.9254	.7730	.0061	.9371	.8722	.0109	.8768	.8585	.0291

embedding our designed two modules into it still brings performance improvement that cannot be ignored (*i.e.*, 1.62%/4.21%/2.24% on EORSSD/ORSSD/ORSI-4199 in F_β^{adp}).

The indispensability of the proposed modules. To prove the individual contribution of saliency-up and edge assignment modules for our GLGCNet, we take a PVTv2-B2-UNet as a baseline and provide three variants by deleting one module in turn. From the quantitative comparison in Table 4, we observe that each module of GLGCNet contributes to the ultimate excellent performance. Although the variant without edge assignment module performs the highest F_β^{adp} on the ORSSD dataset, the other indexes are still far lower than that of the complete GLGCNet. The w/ 4SU indicates that each stage of the basic feature is followed by a saliency-up module. Although the saliency-up module can adaptively enhance the structure information of salient objects, it cannot capture finer details, proven by the performance degradation. These can confirm that the saliency-up module corresponding to global–local context-aware modeling and the edge assignment module for global edge refinement are indispensable to our GLGCNet, and further proves the superiority of the global–local–global scheme.

4.3.2. The design rationality of saliency-up module

We now discuss the various design choices in our saliency-up module that affect the detection performance by constructing four variants, *i.e.*, model '1': w/o SFP, model '2': w/o SFP and MC, model '3': w/o CFP, model '4': replacing the operation of stack&group-conv&shuffle with typical bilinear interpolation for feature upsampling, which means that only one dynamic filter is adopted, model '5': replacing the saliency-up module with ASPP module, and model '6': replacing the saliency-up module with CBAM module. The quantitative results of trained variants are reported in Table 5. The models '5' and '6' are built upon the concept of contrasting examples in Fig. 4. The visual features in this figure and the results of the comparative experiments all confirm the context-aware capability of the saliency-up module. Overall, it can be observed that models '2' and '3' occur different degrees of performance degradation. Although model '2' gets one higher metric score, the other five metrics are lower than the original GLGCNet. The performance decline of model '4' illustrates that stacking and shuffling four dynamic filtered results to achieve upsampling is more effective than typical upsampling for the ORSI-SOD task which is the type of dense prediction. These demonstrate that the design of the saliency-up module is rationality and fits the ORSI-SOD task for better performance.

Table 6
Effectiveness of edge assignment module. EE is the edge exploration block, CA is the concatenation-channel attention operation, and GP denotes the global-wise propagation.

Model	EORSSD			ORSSD			ORSI-4199		
	$S_\alpha \uparrow$	$F_\beta^{adp} \uparrow$	$\mathcal{M} \downarrow$	$S_\alpha \uparrow$	$F_\beta^{adp} \uparrow$	$\mathcal{M} \downarrow$	$S_\alpha \uparrow$	$F_\beta^{adp} \uparrow$	$\mathcal{M} \downarrow$
w/ EA (Ours)	**.9375**	**.8499**	.0055	**.9488**	.8931	**.0071**	**.8839**	**.8672**	**.0274**
wo/ EE	.9347	.8436	.0065	.9429	.8904	.0084	.8799	.8603	.0286
wo/ CA	.9360	.8471	**.0052**	.9437	**.8933**	.0075	.8830	.8635	.0280
wo/ GP	.9344	.8446	.0052	**.9487**	.8947	.0082	.8787	.8591	.0283

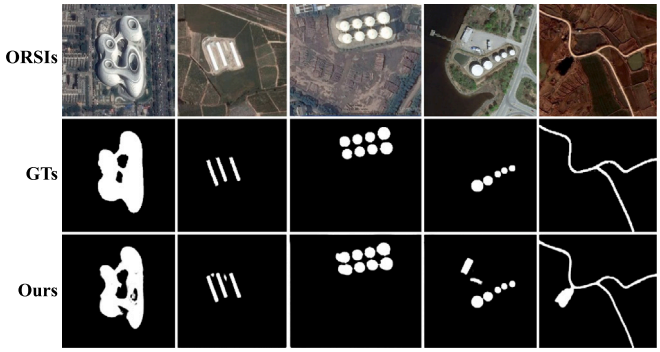

Fig. 9. Some representative failure cases on the challenging scenes.

4.3.3. The effectiveness of edge assignment module to GLGCNet

We construct three variants: deleting the edge exploration block for edge feature construction (*i.e.*, wo/ EE), removing the concatenation-channel attention operation for the enhancement of shallow-level saliency features (*i.e.*, wo/ CA), and deleting the global-wise propagation operation (*i.e.*, wo/GP). The results are demonstrated in Table 6, and the performance of these variants all less inferior to the complete GLGCNet. In general, the above analysis clearly verifies the effectiveness of our edge assignment module. The key operations undertake different functions, promoting the edge assignment module to focus on valuable edge features, which eliminates some negative effects caused by inaccurate preliminary saliency prediction as shown in Fig. 6, to detect more complete salient objects with finer boundaries.

4.4. Failure cases

Although the proposed GLGCNet performs better than existing methods, it still struggles with some limitations in a few challenging scenes. We show some failure cases of our GLGCNet in Fig. 9.

As shown in the first three columns of Fig. 9, it is still challenging to completely distinguish the salient objects with multi-semantic attributes and irregular geometry, such as the irregularly shaped stadium and the internal venues with weak semantic relevance, landmarks and rural buildings with similar appearance and semantics, and the side and top of storage oil tanks. The main reasons are that understanding the semantic correlation and differentiation of multiple attributes is still weak for our GLGCNet and salient objects with multiple relatively independent semantic attributes are rare in training data. As the last two columns of Fig. 9, it is still challenging to completely eliminate the interference of some non-salient objects with training data bias, such as rural buildings near storage oil tanks and lakes beside the road. The high proportion of the two categories of rural building and lake in the training data leads to the fact that in these scenes, even if they are relatively insignificant, these two categories can still be misidentified as salient objects. To deal with these challenging scenes, in the future, two attempts should be made. The first one is developing better learning strategies from the training data with category bias and a few challenging scenes. The second is modeling more comprehensive and adaptive semantic attribute relationships to understand abstract semantic information in the task of ORSI-SOD.

5. Conclusion

In this paper, we explore the synergy of the global-context-aware and local-context-aware modeling and propose an effective GLGCNet to deal with complicated scenes in a global–local–global scheme with a transformer-based backbone. The dynamic convolution-based saliency-up module adaptively highlights pixels that belong to salient objects but have independent attributes by transferring the global basic feature into dynamic kernels, and further realizes the feature upsampling with less information loss. The edge assignment module is executed in a global view, globally exploring the connectivity between preliminary saliency cues and shallow-level features to enhance the edge pixels of salient objects to deal with narrow salient objects with complex geometry. We conduct extensive experiments on three public ORSI-SOD datasets. The experimental results demonstrate the superiority of our proposed GLGCNet.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62171269, and in part by the China Postdoctoral Science Foundation under Grant 2022M722037.

References

Achanta, R., Hemami, S., Estrada, F., Susstrunk, S., 2009. Frequency-tuned salient region detection. In: Proc. IEEE CVPR. pp. 1597–1604. http://dx.doi.org/10.1109/CVPR.2009.5206596.

Brun, A., Cucci, D.A., Skaloud, J., 2022. Lidar point–to–point correspondences for rigorous registration of kinematic scanning in dynamic networks. ISPRS J. Photogramm. Remote Sens. 189, 185–200. http://dx.doi.org/10.1016/j.isprsjprs.2022.04.027.

Chen, Y., Dai, X., Liu, M., Chen, D., Yuan, L., Liu, Z., 2020a. Dynamic convolution: Attention over convolution kernels. In: Proc. IEEE CVPR. pp. 11027–11036. http://dx.doi.org/10.1109/CVPR42600.2020.01104.

Chen, Y., Dai, X., Liu, M., Chen, D., Yuan, L., Liu, Z., 2020b. Dynamic convolution: Attention over convolution kernels. In: Proc. IEEE CVPR. pp. 11027–11036. http://dx.doi.org/10.1109/CVPR42600.2020.01104.

Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A.L., Zhou, Y., 2021a. TransUNet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306.

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

Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., Ma, S., Xu, C., Xu, C., Gao, W., 2021b. Pre-trained image processing transformer. In: Proc. IEEE CVPR. pp. 12294–12305. http://dx.doi.org/10.1109/CVPR46437.2021.01212.

Chen, Z., Xu, Q., Cong, R., Huang, Q., 2020c. Global context-aware progressive aggregation network for salient object detection. In: Proc. AAAI. pp. 10599–10606. http://dx.doi.org/10.1609/aaai.v34i07.6633.

Cong, R., Zhang, Y., Fang, L., Li, J., Zhang, C., Zhao, Y., Kwong, S., 2021. RRNet: Relational reasoning network with parallel multi-scale attention for salient object detection in optical remote sensing images. IEEE Trans. Geosci. Remote Sens. http://dx.doi.org/10.1109/TGRS.2021.3123984.

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y., 2017. Deformable convolutional networks. In: Proc. IEEE ICCV. pp. 764–773. http://dx.doi.org/10.1109/ICCV.2017.89.

Deng, Z., Hu, X., Zhu, L., Xu, X., Qin, J., Han, G., Heng, P.-A., 2018. R^{3}Net: Recurrent residual refinement network for saliency detection. In: Proc. IJCAI. pp. 684–690. http://dx.doi.org/10.24963/ijcai.2018/95.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In: Proc. ICLR. pp. 1–5.

Fan, D.-P., Cheng, M.-M., Liu, Y., Li, T., Borji, A., 2017. Structure-measure: A new way to evaluate foreground maps. In: Proc. IEEE ICCV. pp. 4548–4557. http://dx.doi.org/10.1109/ICCV.2017.487.

Fan, D.-P., Gong, C., Cao, Y., Ren, B., Cheng, M.-M., Borji, A., 2018. Enhanced-alignment measure for binary foreground map evaluation. In: Proc. IJCAI. pp. 698–704. http://dx.doi.org/10.5555/3304415.3304515.

Han, Q., Fan, Z., Dai, Q., Sun, L., Cheng, M.-M., Liu, J., Wang, J., 2022. On the connection between local attention and dynamic depth-wise convolution. In: ICLR.

Han, J., Zhou, P., Zhang, D., Cheng, G., Guo, L., Liu, Z., Bu, S., Wu, J., 2014. Efficient, simultaneous detection of multi-class geospatial targets based on visual saliency modeling and discriminative learning of sparse coding. ISPRS J. Photogramm. Remote Sens. 89, 37–48. http://dx.doi.org/10.1016/j.isprsjprs.2013.12.011.

Hou, X., Bai, Y., Li, Y., Shang, C., Shen, Q., 2021. High-resolution triplet network with dynamic multiscale feature for change detection on satellite images. ISPRS J. Photogramm. Remote Sens. 177, 103–115. http://dx.doi.org/10.1016/j.isprsjprs.2021.05.001.

Hou, Q., Cheng, M.-M., Hu, X., Borji, A., Tu, Z., Torr, P., 2017. Deeply supervised salient object detection with short connections. In: Proc. IEEE CVPR. pp. 5300–5309. http://dx.doi.org/10.1109/TPAMI.2018.2815688.

Hu, J., Shen, L., Albanie, S., Sun, G., Wu, E., 2020a. Squeeze-and-excitation networks. IEEE Trans. Pattern Anal. Mach. Intell. 42 (8), 2011–2023. http://dx.doi.org/10.1109/TPAMI.2019.2913372.

Hu, X., Zhu, L., Qin, J., Fu, C.-W., Heng, P.-A., 2018. Recurrently aggregating deep features for salient object detection. In: Proc. AAAI. pp. 6943–6950. http://dx.doi.org/10.5555/3504035.3504885.

Huang, Z., Chen, H., Liu, B., Wang, Z., 2021. Semantic-guided attention refinement network for salient object detection in optical remote sensing images. Remote Sens. 13 (11), 2163. http://dx.doi.org/10.3390/rs13112163.

Huang, Z., Xiang, T.-Z., Chen, H.-X., Dai, H., 2022. Scribble-based boundary-aware network for weakly supervised salient object detection in remote sensing images. ISPRS J. Photogramm. Remote Sens. 191, 290–301. http://dx.doi.org/10.1016/j.isprsjprs.2022.07.014.

Kingma, D.P., Ba, J.L., 2015. Adam: A method for stochastic optimization. In: ICLR. pp. 1–15.

LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D., 1989. Backpropagation applied to handwritten zip code recognition. Neural Comput. 1 (4), 541–551. http://dx.doi.org/10.1162/neco.1989.1.4.541.

Li, C., Cong, R., Guo, C., Li, H., Zhang, C., Zheng, F., Zhao, Y., 2020. A parallel down-up fusion network for salient object detection in optical remote sensing images. Neurocomputing 415, http://dx.doi.org/10.1016/j.neucom.2020.05.108, 411–420.

Li, C., Cong, R., Hou, J., Zhang, S., Qian, Y., Kwong, S., 2019. Nested network with two-stream pyramid for salient object detection in optical remote sensing images. IEEE Trans. Geosci. Remote Sens. 57 (11), 9156–9166. http://dx.doi.org/10.1109/TGRS.2019.2925070.

Li, G., Liu, Z., Bai, Z., Lin, W., Ling, H., 2022a. Lightweight salient object detection in optical remote sensing images via feature correlation. IEEE Trans. Geosci. Remote Sens. 60, 1–12. http://dx.doi.org/10.1109/TGRS.2022.3145483.

Li, G., Liu, Z., Chen, M., Bai, Z., Lin, W., Ling, H., 2021a. Hierarchical alternate interaction network for RGB-D salient object detection. IEEE Trans. Image Process. 30, 3528–3542. http://dx.doi.org/10.1109/TIP.2021.3062689.

Li, G., Liu, Z., Lin, W., Ling, H., 2022b. Multi-content complementation network for salient object detection in optical remote sensing images. IEEE Trans. Geosci. Remote Sens. 60, 1–13. http://dx.doi.org/10.1109/TGRS.2021.3131221.

Li, G., Liu, Z., Shi, R., Hu, Z., Wei, W., Wu, Y., Huang, M., Ling, H., 2021b. Personal fixations-based object segmentation with object localization and boundary preservation. IEEE Trans. Image Process. 30, 1461–1475. http://dx.doi.org/10.1109/TIP.2020.3044440.

Li, G., Liu, Z., Zeng, D., Lin, W., Ling, H., 2022c. Adjacent context coordination network for salient object detection in optical remote sensing images. IEEE Trans. Cybern. 53 (1), 526–538. http://dx.doi.org/10.1109/TCYB.2022.3162945.

Li, G., Liu, Z., Zhang, X., Lin, W., 2023. Lightweight salient object detection in optical remote-sensing images via semantic matching and edge alignment. IEEE Trans. Geosci. Remote Sens. 61, 1–11. http://dx.doi.org/10.1109/TGRS.2023.3235717.

Li, J., Pan, Z., Liu, Q., Wang, Z., 2021c. Stacked U-shape network with channel-wise attention for salient object detection. IEEE Trans. Multimed. 23, 1397–1409. http://dx.doi.org/10.1109/TMM.2020.2997192.

Liang, M., Hu, X., 2015. Feature selection in supervised saliency prediction. IEEE Trans. Cybern. 45 (5), 914–926. http://dx.doi.org/10.1109/TCYB.2014.2338893.

Liao, M., Zou, Z., Wan, Z., Yao, C., Bai, X., 2022. Real-time scene text detection with differentiable binarization and adaptive scale fusion. TPAMI http://dx.doi.org/10.1109/TPAMI.2022.3155612.

Liu, Y., Gu, Y.-C., Zhang, X.-Y., Wang, W., Cheng, M.-M., 2021a. Lightweight salient object detection via hierarchical visual perception learning. IEEE Trans. Cybern. 51 (9), 4439–4449. http://dx.doi.org/10.1109/TCYB.2020.3035613.

Liu, J.-J., Hou, Q., Cheng, M.-M., Feng, J., Jiang, J., 2019a. A simple pooling-based design for real-time salient object detection. In: Proc. IEEE CVPR. pp. 3912–3921. http://dx.doi.org/10.1109/CVPR.2019.00404.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021b. Swin transformer: Hierarchical vision transformer using shifted windows. In: Proc. IEEE ICCV. pp. 9992–10002. http://dx.doi.org/10.1109/ICCV48922.2021.00986.

Liu, Y., Zhang, X.-Y., Bian, J.-W., Zhang, L., Cheng, M.-M., 2021c. SAMNet: Stereoscopically attentive multi-scale network for lightweight salient object detection. IEEE Trans. Image Process. 30, 3804–3814. http://dx.doi.org/10.1109/TIP.2021.3065239.

Liu, N., Zhang, N., Wan, K., Shao, L., Han, J., 2021d. Visual saliency transformer. In: Proc. IEEE ICCV. pp. 4702–4712. http://dx.doi.org/10.1109/ICCV48922.2021.00468.

Liu, Y., Zhang, S., Wang, Z., Zhao, B., Zou, L., 2022. Global perception network for salient object detection in remote sensing images. IEEE Trans. Geosci. Remote Sens. 60, 1–12. http://dx.doi.org/10.1109/TGRS.2022.3141953.

Liu, Z., Zhao, D., Shi, Z., Jiang, Z., 2019b. Unsupervised saliency model with color Markov chain for oil tank detection. Remote Sens. 11 (9), 1–18. http://dx.doi.org/10.3390/rs11091089.

Mao, Y., Zhang, J., Wan, Z., Dai, Y., Li, A., Lv, Y., Tian, X., Fan, D.-P., Barnes, N., 2021. Transformer transforms salient object detection and camouflaged object detection. arXiv preprint arXiv:2104.10127.

Neubig, G., Dyer, C., Goldberg, Y., Matthews, A., Ammar, W., Anastasopoulos, A., Ballesteros, M., Chiang, D., Clothiaux, D., Cohn, T., Duh, K., Faruqui, M., Gan, C., Garrette, D., Ji, Y., Kong, L., Kuncoro, A., Kumar, G., Malaviya, C., Michel, P., Oda, Y., Richardson, M., Saphra, N., Swayamdipta, S., Yin, P., 2017. Dynet: The dynamic neural network toolkit. arXiv preprint arXiv:1701.03980.

Pang, Y., Zhao, X., Zhang, L., Lu, H., 2020. Multi-scale interactive network for salient object detection. In: Proc. IEEE CVPR. pp. 9410–9419. http://dx.doi.org/10.1109/CVPR42600.2020.00943.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. PyTorch: An imperative style, high-performance deep learning library. In: Proc. NeurIPS. pp. 8024–8035.

Qin, X., Zhang, Z., Huang, C., Gao, C., Dehghan, M., Jagersand, M., 2019. BASNet: Boundary-aware salient object detection. In: Proc. IEEE CVPR. pp. 7479–7489. http://dx.doi.org/10.1109/CVPR.2019.00766.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional networks for biomedical image segmentation. In: Proc. MICCAI. pp. 234–241. http://dx.doi.org/10.1007/978-3-319-24574-4_28.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Zhiheng Huang, A.K., Khosla, A., Bernstein, M., 2015. ImageNet large scale visual recognition challenge. In: Int. J. Comput. Vis.. pp. 211–252. http://dx.doi.org/10.1007/s11263-015-0816-y.

Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z., 2016. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proc. IEEE CVPR. pp. 1874–1883. http://dx.doi.org/10.1109/TPAMI.2022.3155612.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jegou, H., 2021. Training data-efficient image transformers & distillation through attention. In: Proc. ICML, Vol. 139. pp. 10347–10357.

Tu, Z., Ma, Y., Li, C., Tang, J., Luo, B., 2021. Edge-guided non-local fully convolutional network for salient object detection. IEEE Trans. Circuits Syst. Video Technol. 31 (2), 582–593. http://dx.doi.org/10.1109/TCSVT.2020.2980853.

Tu, Z., Wang, C., Li, C., Fan, M., Zhao, H., Luo, B., 2022. ORSI salient object detection via Multiscale Joint Region and boundary model. IEEE Trans. Geosci. Remote Sens. 60, 1–13. http://dx.doi.org/10.1109/TGRS.2021.3101359.

Wang, J., Chen, K., Xu, R., Liu, Z., Loy, C.C., Lin, D., 2021a. CARAFE++: Unified content-aware ReAssembly of features. IEEE Trans. Pattern Anal. Mach. Intell. http://dx.doi.org/10.1109/TPAMI.2021.3074370.

Wang, X., Girshick, R., Gupta, A., He, K., 2018a. Non-local neural networks. In: Proc. IEEE CVPR. pp. 7794–7803. http://dx.doi.org/10.1109/CVPR.2018.00813.

Wang, W., Lai, Q., Fu, H., Shen, J., Ling, H., Yang, R., 2022a. Salient object detection in the deep learning era: An in-depth survey. IEEE Trans. Pattern Anal. Mach. Intell. 44 (6), 3239–3259. http://dx.doi.org/10.1109/TPAMI.2021.3051099.

Wang, L., Li, R., Zhang, C., Fang, S., Duan, C., Meng, X., Atkinson, P.M., 2022b. UNetFormer: A UNet-like transformer for efficient semantic segmentation of remote sensing urban scene imagery. ISPRS J. Photogramm. Remote Sens. 190, 196–214. http://dx.doi.org/10.1016/j.isprsjprs.2022.06.008.

Wang, Z., Liu, Z., Li, G., Wang, Y., Zhang, T., Xu, L., Wang, J., 2021b. Spatio-temporal self-attention network for video saliency prediction. IEEE Trans. Multimed. http://dx.doi.org/10.1109/TMM.2021.3139743.

Wang, Q., Liu, Y., Xiong, Z., Yuan, Y., 2022c. Hybrid feature aligned network for salient object detection in optical remote sensing imagery. IEEE Trans. Geosci. Remote Sens. 60, 1–15. http://dx.doi.org/10.1109/TGRS.2022.3181062.

Wang, W., Shen, J., 2018. Deep visual attention prediction. IEEE Trans. Image Process. 27 (5), 2368–2378. http://dx.doi.org/10.1109/TIP.2017.2787612.

Wang, W., Shen, J., Cheng, M.-M., Shao, L., 2019a. An iterative and cooperative top-down and bottom-up inference network for salient object detection. In: Proc. IEEE CVPR. pp. 5961–5970. http://dx.doi.org/10.1109/CVPR.2019.00612.

Wang, W., Shen, J., Dong, X., Borji, A., Yang, R., 2020. Inferring salient objects from human fixations. IEEE Trans. Pattern Anal. Mach. Intell. 42 (8), 1913–1927. http://dx.doi.org/10.1109/TPAMI.2019.2905607.

Wang, W., Shen, J., Ling, H., 2019b. A deep network solution for attention and aesthetics aware photo cropping. IEEE Trans. Pattern Anal. Mach. Intell. 41 (7), 1531–1544. http://dx.doi.org/10.1109/TPAMI.2018.2840724.

Wang, W., Shen, J., Sun, H., Shao, L., 2018b. Video co-saliency guided co-segmentation. IEEE Trans. Circuits Syst. Video Technol. 28 (8), 1727–1736. http://dx.doi.org/10.1109/TCSVT.2017.2701279.

Wang, W., Shen, J., Yang, R., Porikli, F., 2018c. Saliency-aware video object segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 40 (1), 20–33. http://dx.doi.org/10.1109/TPAMI.2017.2662005.

Wang, W., Shen, J., Yu, Y., Ma, K.-L., 2017. Stereoscopic thumbnail creation via efficient stereo saliency detection. IEEE Trans. Vis. Comput. Graphics 23 (8), 2014–2027. http://dx.doi.org/10.1109/TVCG.2016.2600594.

Wang, W., Sun, G., Gool, L.V., 2022d. Looking beyond single images for weakly supervised semantic segmentation learning. IEEE Trans. Pattern Anal. Mach. Intell. http://dx.doi.org/10.1109/TPAMI.2022.3168530.

Wang, W., Xie, E., Li, X., Fan, D.-P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L., 2021c. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In: Proc. IEEE ICCV. pp. 548–558. http://dx.doi.org/10.1109/ICCV48922.2021.00061.

Wang, W., Xie, E., Li, X., Fan, D.-P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L., 2022e. PVT v2: Improved baselines with pyramid vision transformer. Comput. Vis. Media http://dx.doi.org/10.1007/s41095-022-0274-8.

Wang, S., Yang, S., Wang, M., Jiao, L., 2019c. New contour cue-based hybrid sparse learning for salient object detection. IEEE Trans. Cybern. http://dx.doi.org/10.1109/TCYB.2018.2881482.

Wang, W., Zhao, S., Shen, J., Hoi, S.C.H., Borji, A., 2019d. Salient object detection with pyramid attention and salient edges. In: Proc. IEEE CVPR. pp. 1448–1457. http://dx.doi.org/10.1109/CVPR.2019.00154.

Wang, W., Zhou, T., Yu, F., Dai, J., Konukoglu, E., Gool, L.V., 2021d. Exploring cross-image pixel contrast for semantic segmentation. In: Proc. IEEE ICCV. pp. 7283–7293. http://dx.doi.org/10.1109/ICCV48922.2021.00721.

Woo, S., Park, J., Lee, J.-Y., Kweon, I.S., 2018. CBAM: Convolutional block attention module. In: ECCV. pp. 3–19. http://dx.doi.org/10.1007/978-3-030-01234-2_1.

Xu, B., Hu, H., Zhu, Q., Ge, X., Jin, Y., Yu, H., Zhong, R., 2021a. Efficient interactions for reconstructing complex buildings via joint photometric and geometric saliency segmentation. ISPRS J. Photogramm. Remote Sens. 175, 416–430. http://dx.doi.org/10.1016/j.isprsjprs.2021.03.006.

Xu, B., Liang, H., Liang, R., Chen, P., 2021b. Locate globally, segment locally: A progressive architecture with knowledge review network for salient object detection. In: Proc. AAAI. pp. 3004–3012. http://dx.doi.org/10.1609/aaai.v35i4.16408.

Yang, B., Bender, G., Le, Q.V., Ngiam, J., 2019a. CondConv: Conditionally parameterized convolutions for efficient inference. In: Proc. NeurIPS. pp. 1307–1318. http://dx.doi.org/10.5555/3454287.3454404.

Yang, H., Cao, Z., Cui, Z., Pi, Y., 2019b. Saliency detection of targets in polarimetric SAR images based on globally weighted perturbation filters. ISPRS J. Photogramm. Remote Sens. 147, 65–79. http://dx.doi.org/10.1016/j.isprsjprs.2018.10.017.

Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z., Tay, F.E.H., Feng, J., Yan, S., 2021. Tokens-to-token vit: Training vision transformers from scratch on ImageNet. In: Proc. IEEE ICCV. pp. 538–547. http://dx.doi.org/10.1109/ICCV48922.2021.00060.

Zhang, Q., Cong, R., Li, C., Cheng, M.-M., Fang, Y., Cao, X., Zhao, Y., Kwong, S., 2021. Dense attention fluid network for salient object detection in optical remote sensing images. IEEE Trans. Image Process. 30, 1305–1317. http://dx.doi.org/10.1109/TIP.2020.3042084.

Zhang, L., Ma, J., 2021. Salient object detection based on progressively supervised learning for remote sensing images. IEEE Trans. Geosci. Remote Sens. http://dx.doi.org/10.1109/TGRS.2020.3045708.

Zhang, L., Yang, K., 2014. Region-of-interest extraction based on frequency domain analysis and salient region detection for remote sensing image. IEEE Geosci. Remote Sens. Lett. 11 (5), 916–920. http://dx.doi.org/10.1109/LGRS.2013.2281827.

Zhang, L., Zhang, J., 2017. A new saliency-driven fusion method based on complex wavelet transform for remote sensing images. IEEE Geosci. Remote Sens. Lett. 14 (12), 2433–2437. http://dx.doi.org/10.1109/LGRS.2017.2768070.

Zhao, J., Liu, J.-J., Fan, D.-P., Cao, Y., Yang, J., Cheng, M.-M., 2019. EGNet: Edge guidance network for salient object detection. In: Proc. IEEE ICCV. pp. 8779–8788. http://dx.doi.org/10.1109/ICCV.2019.00887.

Zhao, X., Pang, Y., Zhang, L., Lu, H., Zhang, L., 2020. Suppress and balance: A simple gated network for salient object detection. In: Proc. ECCV. pp. 35–51. http://dx.doi.org/10.1007/978-3-030-58536-5_3.

Zhao, D., Wang, J., Shi, J., Jiang, Z., 2015. Sparsity-guided saliency detection for remote sensing images. J. Appl. Remote Sens. 9 (1), 1–14. http://dx.doi.org/10.1117/1.JRS.9.095055.

Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., Zhang, L., 2021. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: Proc. IEEE CVPR. pp. 6877–6886. http://dx.doi.org/10.1109/CVPR46437.2021.00681.

Zhou, Y., Huo, S., Xiang, W., Hou, C., Kung, S.-Y., 2019. Semi-supervised salient object detection using a linear feedback control system model. IEEE Trans. Cybern. 49 (4), 1173–1185. http://dx.doi.org/10.1109/TCYB.2018.2793278.

Zhou, J., Jampani, V., Pi, Z., Liu, Q., Yang, M.-H., 2021. Decoupled dynamic filter networks. In: Proc. IEEE CVPR. pp. 6643–6652. http://dx.doi.org/10.1109/CVPR46437.2021.00658.

Zhou, X., Shen, K., Liu, Z., Gong, C., Zhang, J., Yan, C., 2022a. Edge-aware multiscale feature integration network for salient object detection in optical remote sensing images. IEEE Trans. Geosci. Remote Sens. 60, 1–15. http://dx.doi.org/10.1109/TGRS.2021.3091312.

Zhou, X., Shen, K., Weng, L., Cong, R., Zheng, B., Zhang, J., Yan, C., 2022b. Edge-guided recurrent positioning network for salient object detection in optical remote sensing images. IEEE Trans. Cybern. http://dx.doi.org/10.1109/TCYB.2022.3163152.

Zhou, H., Xie, X., Lai, J.-H., Chen, Z., Yang, L., 2020. Interactive two-stream decoder for accurate and fast saliency detection. In: Proc. IEEE CVPR. pp. 9138–9147. http://dx.doi.org/10.1109/CVPR42600.2020.00916.

Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J., 2021. Deformable DETR: Deformable transformers for end-to-end object detection. In: Proc. ICLR.