Go Closer to See Better: Camouflaged Object Detection via Object Area Amplification and Figure-Ground Conversion

Haozhe Xing, Shuyong Gao, Yan Wang, Xujun Wei, Hao Tang, and Wenqiang Zhang

***Abstract*— Camouflaged Object Detection (COD) aims to detect objects well hidden in the environment. The main challenges of COD come from the high degree of texture and color overlapping between the objects and their surroundings. Inspired by that humans tend to go closer to the object and magnify it to recognize ambiguous objects more clearly, we propose a novel three-stage architecture called Search-Amplify-Recognize and design a network SARNet to address the challenges. Specifically, In the Search part, we utilize an attention-based backbone to locate the object. In the Amplify part, to obtain rich searched features and fine segmentation, we design Object Area Amplification modules (OAA) to perform cross-level and adjacent-level feature fusion and amplifying operations on feature maps. Besides, the OAA can be regarded as a simple and effective plug-in module to integrate and amplify the feature maps. The main components of the Recognize part are the Figure-Ground Conversion modules (FGC). The FGC modules alternately pay attention to the foreground and background to precisely separate the highly similar foreground and background. Extensive experiments on benchmark datasets show that our model outperforms other SOTA methods not only on COD tasks but also in COD downstream tasks, such as polyp segmentation and video camouflaged object detection. Source codes will be available at https://github.com/Haozhe-Xing/SARNet.**

***Index Terms*— Camouflaged object detection, search-amplify-recognize architecture, figure-ground conversion.**

I. Introduction

CAMOUFLAGE is a survival skill that animals acquire with constant evolution. In order to avoid predators or prey on other animals [1], animals employ several camouflage skills to make themselves difficult to be identified. Common camouflage methods mainly include background matching and disruptive coloration, both of which allow the camouflaged objects to hide better in the environment. Camouflage Object Detection (COD) [2] is an emerging and challenging task. The challenges of COD come from the high similarity of foreground and background. In general, the color, texture, and brightness of the camouflaged object are consistent with the surrounding environment. COD has a rich history in biology, art, and the military [3]. It also has great application value in agriculture, medical care [4], military, marine animal segmentation [5], and art [6].

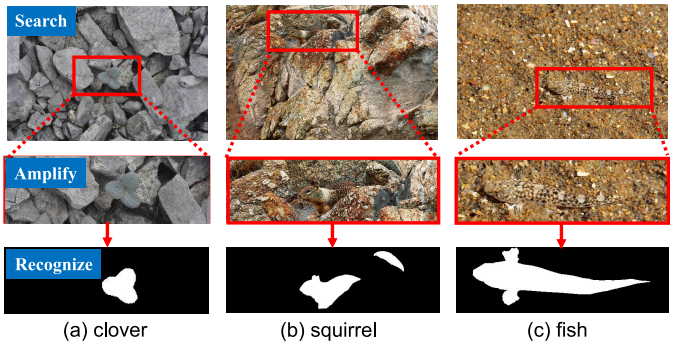


Fig. 1. Illustration of our Search-Amplify-Recognize (SAR) architecture for the COD task. If we examine the first line (a) ∼ (c) directly, it is difficult to distinguish or even locate the camouflaged objects. However, if we increase the size of the object area, it becomes simpler to identify them precisely.

In recent years, Salient Object Detection (SOD) [7], [8], [9] has achieved a booming development with a large number of sub-tasks, such as weakly-supervised SOD [10], co-salient object detection [11], RGB-T salient object detection [12], few-cost salient object detection [13], etc. In contrast, COD is slower in development. One aspect is that large-scale open-source dataset is not available until 2020 [14], the other reasons are as follows: (1) Feature extraction module has difficulty in obtaining discriminative feature embeddings for camouflaged objects. (2) The extreme similarity between foreground and background makes it difficult to segment the camouflaged objects. In recent years, there are a number of approaches in similar types of tasks such as HTC [15], PiCANet [16], BASNet [17], PFANet [18], EGNet [19], F3Net [20], GCPANet [21], CIR-Net [22], RRNet [23] VST [24], and these models achieve decent performance in camouflaged object detection. SINet [14] and PFNet [25] divide

Manuscript received 9 February 2023; accepted 27 February 2023. Date of publication 10 March 2023; date of current version 4 October 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62072112, in part by the Fudan University-CIOMP Joint Fund under Grant FC2019-005, and in part by the Double First-Class Construction Fund under Grant XM03211178. This article was recommended by Associate Editor H. Meng. *(Corresponding authors: Shuyong Gao; Wenqiang Zhang.)*

Haozhe Xing, Yan Wang, and Xujun Wei are with the Academy for Engineering and Technology, Fudan University, Shanghai 200433, China (e-mail: hzxing21@fudan.edu.cn; yanwang19@fudan.edu.cn; xjwei21@m.fudan.edu.cn).

Shuyong Gao is with the Shanghai Key Laboratory of Intelligent Information Processing, School of Computer Science, Fudan University, Shanghai 200433, China (e-mail: sygao18@fudan.edu.cn).

Hao Tang is with the Department of Information Technology and Electrical Engineering, ETH Zurich, 8092 Zurich, Switzerland (e-mail: hao.tang@vision.ee.ethz.ch).

Wenqiang Zhang is with the Shanghai Key Laboratory of Intelligent Information Processing, School of Computer Science, Academy for Engineering and Technology, Fudan University, Shanghai 200433, China, and also with the Yiwu Research Institute of Fudan University, Yiwu, Zhejiang 322000, China (e-mail: wqzhang@fudan.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2023.3255304.

Digital Object Identifier 10.1109/TCSVT.2023.3255304

the process of identifying camouflaged objects into two steps: search and identification. Specifically, the prospective objects are initially roughly located from a global perspective in the detection phase. Then the rough location clues are used to refine the segmentation of camouflaged objects gradually. Motivated by the process that humans tend to go closer to magnify the object to see the target more clearly when observing an ambiguous object, we develop a new promising Search-Amplify-Recognize (SAR) architecture as shown in Fig. 1, which is different from Search-Identify [14], [25] architecture. Fig. 1 visually illustrates that the camouflaged object becomes easily segmented after enlarging the area where the camouflaged object is located. Considering that directly increasing the resolution of the input image consumes a large number of computational resources, we decide to zoom in only on the area where the object is located and carefully design the Amplify part.

As shown in Fig. 2, the camouflage objects are easily grouped into the background, and they are likely to be missed if the model ignores mining background features. Researchers attempted to use texture-aware features [26], [27], cross-level features fusion [28], [29], and edge features [30], [31] to solve these issues. Due to the great similarity between the camouflage objects and their surroundings, these approaches may misidentify the background and foreground of camouflaged objects, resulting in incomplete object recognition. Inspired by the process that foreground and background can be transformed into each other when people's attention shifts [32], [33], [34], we develop the Recognize part. Figure-Ground Conversion (FGC) modules in the Recognize part alternately mine camouflaged features from foreground and background and allow confused misclassified features to be retrieved. More complete camouflage objects can be obtained by applying the figure-ground conversion architecture. Furthermore, the FGC module pays more attention to the edge area of the background without taking the ground-truth edge as supervision, which avoids introducing a lot of redundant background information.

In summary, our contributions are as follows:

- We propose a three-stage architecture called Search-Amplify-Recognize (SAR) and design the SARNet model. The search and amplify parts aim to roughly locate and magnify the object, and the Recognize part fine segments the object.
- We propose a Figure-Ground Conversion (FGC) module which can selectively focus on the foreground or the edge area of the background. Besides, we develop a simple but effective plug-in module, named the Object Area Amplification module (OAA), which can also promote other COD models.
- Extensive experiments on all the benchmark datasets demonstrate that our method outperforms previous state-of-the-art methods by large margins. In addition, our method achieves state-of-the-art performance on polyp segmentation and Video Camouflaged Object Detection (VCOD), which illustrates that our method has great application value for COD downstream tasks. Besides, this work discusses the effect of the resolution of the input image and feature maps on the COD task.

(a) cat (b) transparent person (c) seahorse

Fig. 2. Examples in which foreground and background are very similar. When focusing on the head of the cat (a), we initially find a cat. However, if paying more attention to the white cushion first, we find a cushion.

II. Related Work

A. Camouflaged Object Detection

Fan et al. [2], [14] proposed COD10K, the largest dataset in the field of camouflaged object detection, and also proposes a camouflaged object detection network based on search and recognition patterns. The appearance of this work has also made deep learning-based camouflage object detection booming. Positioning and Focus Network (PFNet) [25] introduced the concept of distraction to the COD problem and develops a distraction mining strategy to make the prediction more accurate. Zhai et al. [31] put forward a graph-based, mutual learning approach for COD and camouflaged object-aware edge extraction (COEE) tasks. Camouflaged object ranking (COR) and camouflaged object localization (COL) tasks and corresponding training and testing datasets are proposed by [35]. Li et al. [36] introduced a joint salient object detection and camouflaged object detection network within an adversarial learning framework. Sun et al. [28] integrated a context-aware module to get rich global context information for improving COD accuracy. [30] provided a module based on the uncertainty of pseudo-edge labels to improve the performance of SINet-V2 [2]. Pang et al. [37] introduced a mixed-scale triplet network using triple inputs (0.5× scale, 1× scale, 1.5× scale) that achieved a good performance but has big FLOPs (101.8G) as shown on Table II.

There are five COD-related works published on the TCSVT. Zhang et al. [38] take advantage Bayesian framework to detect camouflaged foreground pixels in videos. Li et al. [5] proposed a marine animal segmentation dataset named MAS3K, which is a COD downstream application. Bi et al. [39] reviewed the methods and datasets for camouflaged object detection and provided some applications and research directions. C^2FNet [29] proposed a context-aware module, and an effective cross-level fusion module, which achieves a good performance. TANet [26] designed a novel texture-aware refinement model and a boundary-consistency loss to improve the model performance.

Different from these methods using context-aware [29] or texture-aware [26], we consider that amplifying the objects' area and alternately focusing on foreground and background information are promising directions to break objects' camouflage. In this paper, we propose a Search-Amplify-Recognize (SAR) architecture and carefully design the SARNet. In addition, due to the high similarity of the foreground and background of the camouflaged object, existing methods often

confuse foreground and background. To address this issue, we design a novel figure-ground conversion module to more accurately distinguish the foreground and background.

To validate the generalization capability of our model, we conduct experiments on polyp segmentation and video camouflage object detection tasks. The model achieves optimal performance on both tasks, which shows great potential and application value.

B. Feature Amplification

Common feature amplification methods include upsampling operation and feature super-resolution. Upsampling operation is also a key step of feature super-resolution or a simple feature super-resolution method. Linear interpolation-based including nearest interpolation, bilinear interpolation, bicubic interpolation, and deep learning based methods, such as PixelShuffle [40], DUpsampling [41], Meta-SR [42] and CARAFECR [43]. With the development of deep learning, super-resolution on a single image or feature-level super-resolution is applied to many computer vision fields. For example, [44], [45], [46], [47], and [48] applied super-resolution on a single image to improve the performance of object detection. Some works [49], [50] proposed feature-level super-resolution methods. Li et al. [50] leveraged pixel-level attention to model long-range dependency and global information for better reconstruction. Noh et al. [49] proposed a feature-level super-resolution approach for proposal-based detectors.

However, whether super-resolution is useful for camouflage object detection has not been explored. To the best of our knowledge, we are the first to propose and verify that the COD task is extremely sensitive to the resolution of the feature maps and the images. Based on this finding, we design a simple but effective object area amplification module that utilizes cross-scale features to complement detail features and an upsampling operation to magnify the object area. Extensive experiments demonstrate that the model with this simple module can achieve state-of-the-art performance with less computational resource consumption.

C. Figure-Ground Organization

Foreground and background can be swapped as attention shifts, which is pointed out by the figure-ground organization in Gestalt psychology. Inspired by it, we propose the Figure-Ground Conversion module (FGC) to enable the model to alternately focus on the foreground and the background.

One similar approach to the figure-ground organization is disruption minimization [25], [51], which is used in many tasks, such as salient object detection [52], [53], visual tracking [54], and semantic segmentation [55]. However, [52], [53], [54], and [55] only focused on disruption in foreground or background. Some works [25], [51] developed complex structures to focus disruption in both foreground and background in one module. Different from them, our FGC module is a switch structure, which can choose to focus on the foreground or the background. In addition, different from other methods that use all the background information, our Edge-FGC modules only focus on the edge area in the background, which can avoid introducing too much useless information and has better performance on COD tasks. In addition, we employ the inflation algorithm (a traditional morphological operation) to get receptive fields around the target, which makes our module more effective and simple.

TABLE I
QUANTITATIVE COMPARISON THAT RETRAIN AND TEST MODELS ON COD AND SOD TASKS WITH DIFFERENT INPUT RESOLUTIONS. OTHERS REPRESENT THE AVERAGE OF OTHER EVALUATION METRICS EXCEPT FOR MAE

Model	SINet [14]				PFNet [25]			
Input-size	$224^2 \rightarrow 416^2$		$224^2 \rightarrow 672^2$		$224^2 \rightarrow 416^2$		$224^2 \rightarrow 672^2$	
\	SOD	COD	SOD	COD	SOD	COD	SOD	COD
$MAE(\%) \downarrow$	-0.6	**-1.3**	-0.5	**-1.7**	-0.3	**-1.4**	-0.5	**-1.8**
$Others(\%) \uparrow$	+2.30	**+7.60**	+2.55	**+11.0**	+2.30	**+8.30**	+3.82	**+11.55**

III. METHODOLOGY

A. Motivation

When people attempt to recognize a confusing image, they subconsciously zoom it in to make it clear and easier to view. This process is quite similar to increasing the resolution of input images in COD tasks. Therefore, it is likely that increasing the resolution of input images could be an effective approach for identifying camouflaged objects. To validate this conjecture, we feed PFNet [25] and SINet [14] models with different input resolutions, then we retrain and test them on COD and SOD datasets, respectively. For a fair comparison, all the experiment details are the same except for datasets, and all the results are evaluated with the same code. In order to compare the sensitivity of SOD and COD tasks to the input resolution more intuitively, we performed statistics on Table I.

For PFNet, as can be seen from Table I, in COD tasks, MAE decreases by 1.4 % when the input size raised from 224×224 to 416×416 , while MAE only decreases by 0.3 % in SOD tasks. For SINet, when the input size raised from 224×224 to 672×672 , the growth of $Others$ in COD (11.0 %) is 4.3 times that of SOD (2.55 %). $Others$ represents the average of E_ϕ^{ad}, S_α, F_{max}, F_ϕ, F_ϕ^{ad} and F_β^w. Although increasing the resolution of input images or super-resolution is common in high-level computer vision tasks, it doesn't mean that they are very efficient methods in all computer vision tasks. In addition, either adding super-resolution to the method or taking the high-resolution images as inputs introduces a significant amount of computational expense, so it is worth discussing whether the large cost of introducing super-resolution methods is valuable in relation to the performance gains it brings. As shown in Table I, introducing super-resolution methods only leads to tiny performance gains in SOD tasks, while bringing huge performance gains in COD tasks, which illustrates COD tasks are particularly sensitive to resolution and confirms our conjecture.

Considering increasing the input resolution of the model will often bring a large cost of computing resources, we decide to only amplify the object area and propose a new promising Search-Amplify-Recognize architecture. As shown in Fig. 1,

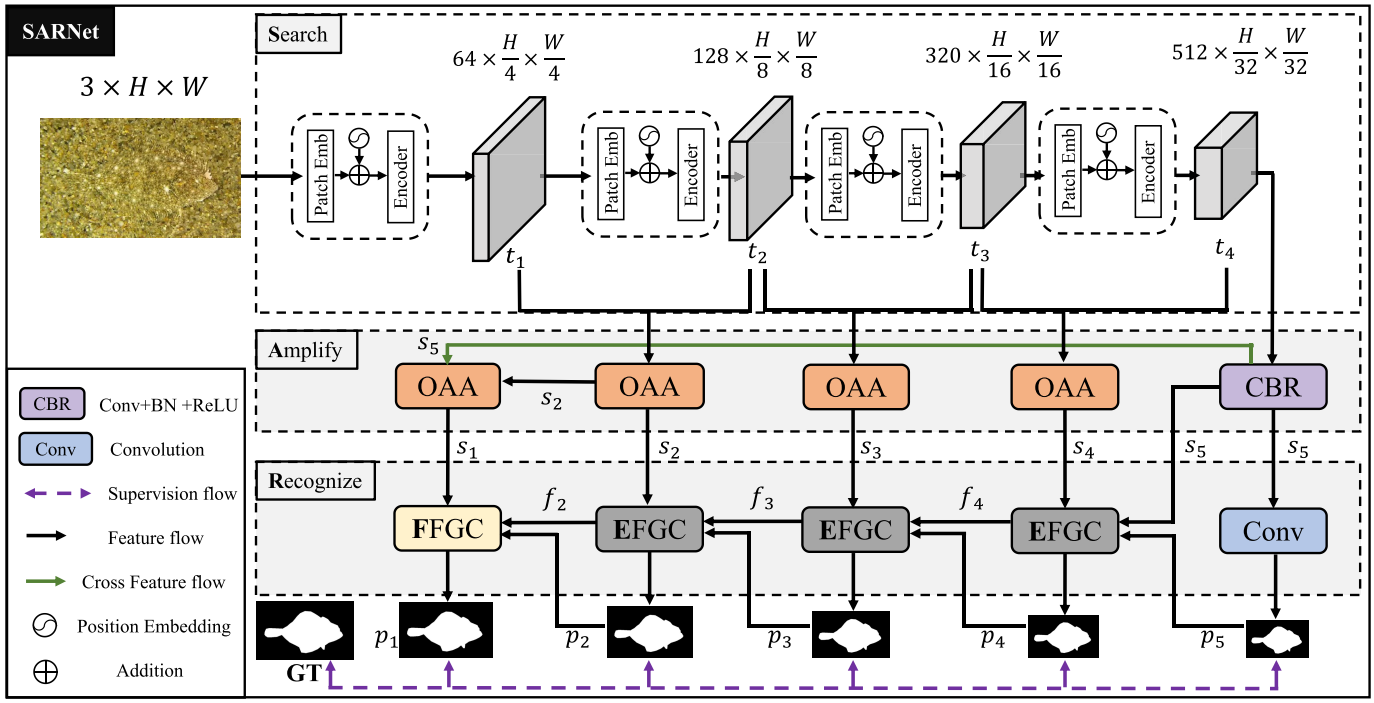


Fig. 3. The architecture of the SARNet. It consists of three main components: Pyramid Vision Transformer as the backbone to **search** coarse features; Object Area Amplification (OAA) module to rich and **amplify** object features; Figure-Ground Conversion modules (FFGC, EFGC) are utilized for **recognition**.

the images with small resolution make it more challenging to perfectly segment the camouflaged objects. However, if we amplify the area where the camouflaged object is located, it becomes relatively easy to finely segment the camouflaged object.

B. Network Overview

Fig. 3 shows the overall architecture of our proposed model. The model is divided into three parts: The Search part, the Amplify part, and the Recognize part. Pyramid Vision Transformer [56] is employed as the backbone network, which served as the Search part. The Amplify part consists of four OAA modules and a CBR block. As for the Recognize part, it includes four FGC modules and a convolution block.

C. The Search Part

We choose Pyramid Vision Transformer (PVT) [56] as our backbone (PVT-V2-B3) to extract multi-level features, which serves as an attention-based feature search to produce high-resolution feature maps and has low memory costs. Pyramid Vision Transformer consists of four stages. Patches $\frac{HW}{4^2}$ are generated by feeding the input image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ firstly. Then, we feed the flattened patches to a linear projection to get $\frac{HW}{4^2} \times C_1$ embedded patches. After that, the embedded patches, along with position embedding, pass through a Transformer encoder with an L1 layer, and the output is reshaped to a $\frac{H}{4} \times \frac{W}{4} \times C_1$ feature map t_1. In the same way, using the feature map from the prior stage as input, we obtain the following feature maps t_2, t_3, and t_4. Our proposed object area amplification module and figure-ground conversion module are based on an attention-based feature search.

D. The Magnify Part

We actively or passively zoom the image to take advantage of local low-resolution detailed cues to compose similar objects when we cannot see an image clearly. Motivated by this progress, we design the Amplify part, which consists of four OAA modules and a CBR block. The Amplify part performs cross-level and adjacent-level feature fusion to alleviate the problem that the searched features are not rich enough to be recognized. In addition, it utilizes amplifying operations on feature maps, which is beneficial to the fine segmentation of the Recognize part. It builds a good connecting link between the Search part and the Recognize part. There are four inputs $\{t_i | i = 1, 2, 3, 4\}$ and five output $\{s_i | i = 1, 2, 3, 4, 5\}$ in the Amplify part. Specifically, $\{s_i | i = 1, 2, 3, 4\}$ are obtained from the OAA modules and s_5 is gotten by feeding t_4 into a sequential operation that consists of 3×3 Convolution layer, Batch normalization, and ReLU activation.

1) Object Area Amplification Module (OAA): As shown in Fig. 5, OAA takes $t_i \in \mathbb{R}^{C_1 \times H_1 \times W_1}$ and $t_{i+1} \in \mathbb{R}^{C_2 \times H_2 \times W_2}$ of two adjacent scales as input to two branches B_1 and B_2. In branch B_1, t_i is fed into one 3×3 convolution layer followed by batch normalization and a GeLU [57] activation to obtain squeezed feature which is then magnified by one bilinear interpolate operation. In branch B_2, aiming at exploiting the adjacent deeper feature while suppressing noise from the shallow layer, take as input t_{i+1} and performs similar operations as branch B_1. The features of these two branches are fused by a nonlinear transformation as the OAA output $s_i \in \mathbb{R}^{C_{out} \times H_{out} \times W_{out}}$. The process for obtaining $\{s_i | i = 2, 3, 4\}$:

$$s_i = F_{CBG}(Cat(F_{CBG}(U(t_i)),\ F_{CBG}(U(t_{i+1})))), \quad (1)$$

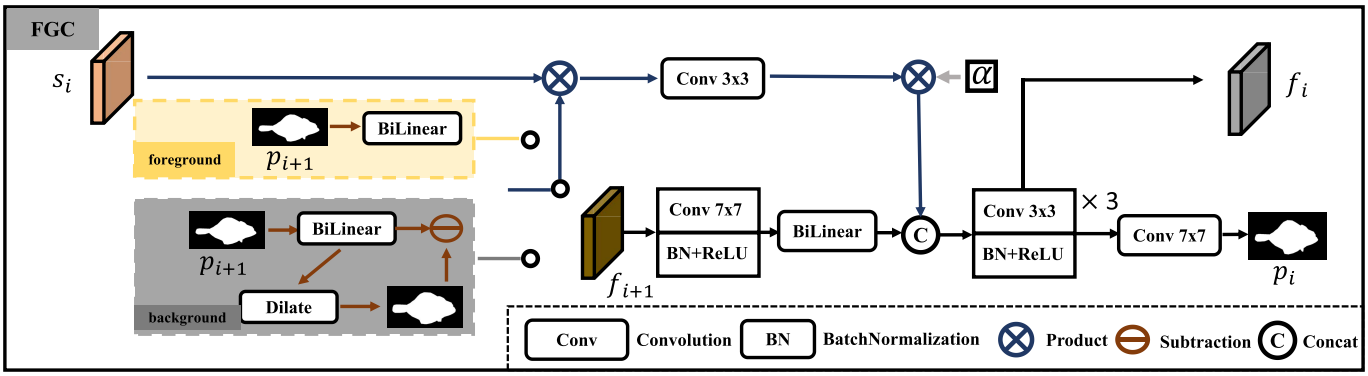


Fig. 4. The architecture of the FGC module. When FGC selects the foreground, we call it the FFGC module, otherwise, we call it the EFGC module. As for FFGC and EFGC, there are three inputs: current-level features s_i, deep-level features f_{i+1}, and deep-level predictions p_{i+1}.

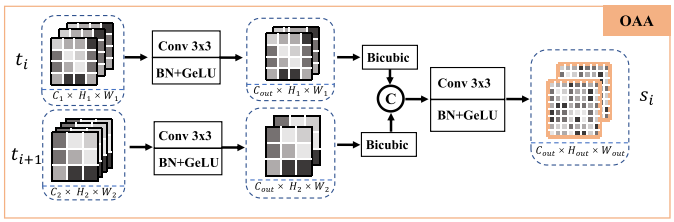


Fig. 5. The architecture of the OAA module. There are two inputs of this module: current-level t_i and deeper-level t_{i+1} feature maps. Bicubic means upsampling operation using Bicubic interpolation.

where t_i and t_{i+1} denote the extracted features. The Bicubic interpolation Upsampling operation is represented by U. F_{CBG} is the sequential operation consists of 3×3 Convolution layer, Batch normalization, and GeLU activation; Cat denotes the concatenation operation. To make the model have a global receptive field, we feed the cross-scale features s_5 and s_2 into the OAA module to obtain s_1.

E. The Recognize Part

Since many camouflage objects are quite similar to their surroundings, the model can lose some of their structures in the background, such as the examples shown in Fig. 2. To address this problem, we propose the Recognize part. The Recognize part continuously refines the segmentation results by alternately focusing on the foreground or background. It helps the model to avoid the false foreground and background and identify the whole body of camouflaged objects accurately. The Recognize part mainly consists of FGC modules. The FGC module can choose to pay attention to the foreground (FFGC) or the edge area in the background (EFGC).

1) Figure-Ground Conversion Module (FGC): As shown in Fig. 4, the FGC module takes three features as input: current-level feature $s_i \in \mathbb{R}^{c_1 \times h \times w}$, deeper-level feature $f_{i+1} \in \mathbb{R}^{c_2 \times \frac{h}{2} \times \frac{w}{2}}$, and deeper-level prediction $p_{i+1} \in \mathbb{R}^{1 \times \frac{h}{2} \times \frac{w}{2}}$. The FGC module can be used as the FFGC module or the EFGC module. As shown in Fig. 3, EFGC modules with a grey color focus on the edge information, and FFGC modules with a yellow color focus on foreground information.

For EFGC, we take dilated upsampled prediction to minus the upsampled prediction to obtain the background attention $p'_{i+1} \in \mathbb{R}^{1 \times h \times w}$. Element-wise multiplying on current-level feature s_i and background attention p'_{i+1} makes the module focus only on features on the edge area. The results of the process are described as $f'_i \in \mathbb{R}^{c_1 \times h \times w}$, and visual results are shown on $edge(bkg)$ in Fig. 7. For FFGC, we use the prediction p_{i+1} to multiply current-level feature s_i in element-wise to pay more attention to the foreground features in the current level. The process is shown as follows:

$$f'_i = s_i * \begin{cases} Di(Up(p_{i+1})) - Up(p_{i+1}), & i = 2, 3, 4 \\ Up(p_{i+1}), & i = 1 \end{cases} \tag{2}$$

where p_{i+1} is the output of FGC_{i+1}; Up represents bilinear interpolation; Di represents dilated, it is an inflation algorithm in computer vision, which is a traditional morphological operation; If $i = 2, 3, 4$, f'_i is the medium result of the EFGC, while for $i = 1$, f'_i is the result of the FFGC.

Then we use a 3×3 convolution layer on f'_i and sequential operations which consist of a 7×7 convolution layer, batch normalization, ReLU activation function, and bilinear upsampling operation on f_{i+1} to align their shapes and features and "Concat" them to get fused features. Moreover, three sequential operations (each operation consists of a 3×3 convolution block, batch normalization, and GeLU activation function.) are utilized to refine the fused features and obtain the output $f_i \in \mathbb{R}^{c_1 \times h \times w}$, which is further fed to a 7×7 convolution layer to get $p_i \in \mathbb{R}^{1 \times h \times w}$. For $i = 1, 2, 3, 4$, the whole process can be formulated as follows:

$$\begin{cases} f'_i = F_{C_3}(f'_i), \quad f_{i+1} = Up(F_{C_7BR}(f_{i+1})), \\ f_i = F_{3C_3BR}(Cat(f_{i+1}, \ \alpha * f'_i))), \quad p_i = F_{C_7}(f_i), \end{cases} \tag{3}$$

where f_{i+1} is the output of FGC_{i+1}, and FGC_1 only has p_1 as the output; F_{C_3} is 3×3 Convolution layer, F_{C_7} is 7×7 convolution layer; Cat represents concatenation function with dim $=$ 1; F_{C_7BR} is sequential operation consists of 7×7 convolution layer, batch normalization, and ReLU activation function. F_{3C_3BR} is three sequential operation, which consists of 3×3 convolution layer, batch normalization, ReLU activation function. α is a learnable parameter. p_5 is obtained by applying one simple 3×3 convolution layer on s_5 : $p_5 = F_{C_3}(s_5)$.

F. Loss Function

Five output predictions $p = \{p_i | i = 1, 2, 3, 4, 5\}$ resized to the same resolution with input, to get the final output

$P = \{P_i | i = 1, 2, 3, 4, 5\}$ of our model. Especially, P_1 is the final prediction. We impose binary cross-entropy (BCE) loss l_{bce} and IoU loss l_{iou}. l_{bce} and l_{iou} are as follows:

$$l_{iou} = 1 - \frac{\sum_{j=1}^{L}[G_j \times P_j]}{\sum_{j=1}^{L}[G_j + P_j - P_j \times P_j]}, \quad (4)$$

$$l_{bce} = -\sum_{j=1}^{L}\sum_{c\in\{0,1\}}[\delta(G_j = c)log(P_j = c)], \quad (5)$$

where L represents the number of pixels, G refers to the ground truth of the camouflaged object, and δ is the indicator function. The loss functions used are $L_{overall}$ as follows:

$$l_{fm} = \alpha l_{bce} + \beta l_{iou}, \quad (6)$$

$$L_{overall} = l^5_{fm} + 2 * l^4_{fm} + 2 * l^3_{fm} + 3 * l^2_{fm} + 6 * l^1_{fm}, \quad (7)$$

where $\alpha = 2$, $\beta = 1$ in our loss function.

IV. Experiment

A. Datasets

To experimentally analyze the proposed approach, SARNet is tested on four benchmarks, CHAMELEON [62], CAMO [63], COD10K [14], and NC4K [35] respectively.

CAMO contains a total of 1,250 images, each of which has at least one camouflaged object. 1,000 images were used as a training set, and 250 images as a test set. CAMO involves a variety of challenging scenarios such as object appearance, background clutter, shape complexity, small object, object occlusion, multiple objects, and distraction.

CHAMELEON searched 76 images with camouflaged animals as keywords from Google search.

COD10K is the largest benchmark dataset at present, which contains 5,066 images of camouflaged objects, including 3,040 images as a training set and 2,026 images for the test. It includes five super-classes and 69 sub-classes.

NC4K is a testing dataset used in camouflage target segmentation, which has a total of 4,121 images. The image scales of these datasets are variable, and there are different levels of camouflage images. In addition, camouflage objects and salient objects coexist, and art images exist in these benchmarks.

B. Evaluation Metrics

The commonly used evaluation indexes in camouflage target detection are as follows: Structure-measure (S_α) [64], Adaptive E-measure (E_ϕ [65], E^{ad}_ϕ [66]), F-measure, Weighted F-measure (F^w_β) [67], Mean Absolute Error (M). Structure-measure aims to evaluate the structural information of the prediction. E-measure pay more attention to pixel-level matching and image-level statistics, while F-measure is more concerned with the comprehensive measure of both the precision and recall of the prediction map. Weighted F-measure (F^w_β) can provide more reliable evaluation results than the traditional F_β. The mean absolute error (M) calculates the element-wise difference. The evaluation code can be found at https://github.com/lartpang/PySODMetrics.

C. Experiment Details

The PyTorch framework is used to implement our model. A twelve-core PC with an Intel(R) Xeon(R) Silver 4310 2.10GHz CPU (total memory capacity is 128GB) and an NVIDIA GeForce RTX 3090 GPU (with 20GB memory) is used for both training and testing. For the training process, 4040 camouflage object images from CAMO and COD10K for training, we resize the input image into a resolution of 384×384 and augment the input image by randomly horizontal flipping and color jittering. The optimizer is Stochastic Gradient Descent (SGD) optimizer with a momentum of 0.9 and a weight decay of 5×10^{-4}. In addition, the training model's batch size is 16, the training epoch is 100, and the basic learning rate is 0.001. Besides, As for testing, input images are resized to 384×384 for model inference and then resized to the original size for evaluation. We don't use any post-processing for our final output of the model. The inference speed of our model on a device with an AMD Ryzen 7 5800H with Radeon Graphics CPU and an NVIDIA GeForce RTX 3060 laptop GPU is 19 FPS, which is the same as C^2FNet (TCSVT'22).

D. Compared With the State-of-the-Art Methods

In order to prove the effectiveness of our model, we compared our model with the existing state-of-the-art methods on four benchmarks using four evaluation metrics. Existing state-of-the-art methods include SINet [14], PFNet [25], MGL [31], ERRNet [61], PraNet [58], RankNet [35], C^2FNet [28], UJSC [36], SINet-V2 [2], UGTR [59], TINet [27], DGNet [60], UR-SINet-V2 [30], C^2FNet-V2 [29], ZoomNet [37]. To ensure a fair comparison, we download the results of these state-of-the-art models and evaluated them with the same evaluation tools.

1) Quantitative Evaluation: It can be seen in Table II that our model achieves the best performance and outperforms other methods by a large margin on four datasets. Compared with ZoomNet, which was published on CVPR 2022, F^w_β average increases by 5.2% with our methods on four benchmarks, which only cost 0.23 times FLOPs. On COD10K, which contains 2026 test images, E_ϕ, S_α and F^w_β of our method respectively increased by 4.4%, 5.3%, and 8.6% compared with C^2FNet-V2 (TCSVT 2022). Row (SARNet-H) means input images into the SARNet with 672 × 672 resolution. Compare with the performance of the SARNet-H and SARNet, F^w_β increases by 4.3%, which also proves COD tasks are sensitive to the input resolution of images.

2) Qualitative Evaluation: Visual results compared with other SOTA methods are shown in Fig. 6. From the results, our model can identify objects under various camouflages, such as background matching, disruptive coloration, small, occluded, and mimicry. Especially, from row 7 and row 9 in Fig. 6, different from other SOTA methods, our methods can avoid the salient object to identify the camouflaged object. For small targets (row 4 and row 5), our model can locate the small targets well and segment them precisely. For background matching types of camouflaged targets (row 1, row 3-4, and row 7-9), our model can identify the camouflaged targets

TABLE II

QUANTITATIVE COMPARISON BETWEEN OUR PROPOSED METHOD AND OTHER STATE-OF-THE-ART METHODS ON FOUR BENCHMARKS. * MEANS MULTI-TRAINING METHODS AND † REPRESENTS USING MORE DATASETS FOR TRAINING. THE TOP TWO RESULTS EXCEPT FOR "SARNET-H" ARE HIGHLIGHTED IN RED, AND BLUE

Method	Source	In-size	Param	FLOPs	CAMO(250 images)					CHAMELEON(76 images)					COD10K(2,026 images)					NC4K(4,121 images)				
					$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$
SINet [14]	CVPR'20	352^2	49.0M	23.3G	0.100	0.771	0.751	0.675	0.606	0.044	0.891	0.869	0.790	0.740	0.051	0.806	0.771	0.634	0.551	0.058	0.871	0.808	0.769	0.723
MGLR* [31]	CVPR'21	473^2	73.7M	378.6G	0.088	0.812	0.775	0.726	0.673	0.030	0.918	0.893	0.834	0.818	0.035	0.852	0.814	0.711	0.673	0.052	0.867	0.832	0.782	0.742
PraNet [58]	MICCAI'20	352^2	30.5M	13.2G	0.086	0.835	0.780	0.737	0.687	0.037	0.923	0.868	0.809	0.785	0.043	0.868	0.794	0.687	0.645	0.055	0.883	0.829	0.778	0.740
PFNet [25]	CVPR'21	416^2	44.3M	26.6G	0.085	0.841	0.782	0.746	0.695	0.033	0.931	0.882	0.828	0.810	0.040	0.877	0.800	0.701	0.660	0.053	0.887	0.829	0.784	0.745
RankNet [35]	CVPR'21	352^2	55.5M	36.9G	0.080	0.838	0.787	0.744	0.696	0.030	0.935	0.890	0.841	0.822	0.037	0.880	0.804	0.715	0.673	0.048	0.895	0.840	0.804	0.766
C^2FNet* [28]	IJCAI'21	352^2	26.3M	13.2G	0.080	0.854	0.796	0.762	0.719	0.032	0.935	0.888	0.844	0.828	0.036	0.890	0.813	0.723	0.686	0.049	0.897	0.838	0.795	0.762
UJSC† [36]	CVPR'21	352^2	55.9M	36.6G	0.073	0.859	0.800	0.772	0.728	0.030	0.945	0.891	0.847	0.833	0.035	0.884	0.809	0.721	0.684	0.047	0.898	0.842	0.806	0.771
TINet [27]	AAAI'21	352^2	-	-	0.087	0.836	0.781	0.728	0.678	-	-	-	-	-	0.042	0.861	0.793	0.679	0.635	0.055	0.879	0.829	0.773	0.734
TANet [26]	TCSVT'21	384^2	-	-	0.083	0.834	0.793	-	0.690	0.036	0.911	0.888	-	0.786	0.041	0.848	0.803	-	0.629	-	-	-	-	-
SINet-V2 [2]	TPAMI'21	352^2	26.6M	14.7G	0.070	0.882	0.820	0.782	0.743	0.030	0.942	0.888	0.835	0.816	0.037	0.887	0.815	0.718	0.680	0.048	0.903	0.847	0.805	0.770
UGTR* [59]	ICCV'21	473^2	37.2M	121.4G	0.086	0.821	0.783	0.735	0.683	0.031	0.910	0.888	0.820	0.800	0.036	0.852	0.817	0.712	0.673	0.052	0.874	0.839	0.787	0.749
UR-SINet-V2 [30]	ACMMM'21	352^2	-	-	0.067	0.891	0.814	-	0.758	0.023	0.960	0.901	-	0.862	0.033	0.903	0.816	-	0.708	0.045	0.910	0.844	-	0.787
DGNet [60]	MIR'22	352^2	21.0M	-	0.057	0.901	0.839	0.806	0.769	-	-	-	-	-	0.033	0.896	0.822	0.728	0.693	0.042	0.911	0.857	0.814	0.784
ERRNet* [61]	PR'22	352^2	67.7M	20.1G	0.088	0.817	0.761	0.714	0.660	0.036	0.927	0.877	0.825	0.805	0.044	0.867	0.780	0.674	0.629	-	-	-	-	-
C^2FNet-V2* [29]	TCSVT'22	352^2	25.2M	18.1G	0.077	0.859	0.799	0.770	0.730	0.028	0.946	0.893	0.857	0.845	0.036	0.887	0.811	0.725	0.691	-	-	-	-	-
ZoomNet [37]	CVPR'22	384^2	32.4M	101.8G	0.069	0.858	0.806	0.784	0.738	0.024	0.948	0.901	0.871	0.850	0.029	0.888	0.837	0.770	0.732	0.041	0.897	0.855	0.824	0.791
SARNet	Ours	384^2	47.2M	23.1G	**0.047**	**0.927**	**0.868**	**0.850**	**0.828**	**0.021**	**0.957**	**0.912**	**0.879**	**0.871**	**0.024**	**0.931**	**0.864**	**0.800**	**0.777**	**0.032**	**0.937**	**0.886**	**0.863**	**0.842**
SARNet-H	Ours	672^2	47.2M	70.7G	**0.046**	**0.929**	**0.874**	**0.866**	**0.844**	**0.017**	**0.972**	**0.933**	**0.915**	**0.909**	**0.021**	**0.941**	**0.885**	**0.839**	**0.820**	**0.032**	**0.934**	**0.889**	**0.872**	**0.851**

TABLE III

QUANTITATIVE COMPARISON BETWEEN OUR PROPOSED METHOD AND OTHER SOTA METHODS USING THE SAME BACKBONE WITH OURS ON FOUR BENCHMARKS. VST IS A SOD METHOD BASED ON THE TRANSFORMER BACKBONE

Method	Source	Param	FLOPs	CAMO (250 images)					CHAMELEON (76 images)					COD10K (2,026 images)					NC4K (4,121 images)				
				$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$
SINet [14]	CVPR'20	46.6M	22.5G	0.051	0.918	0.864	0.831	0.805	0.026	0.948	0.899	0.848	0.835	0.029	0.916	0.848	0.764	0.737	0.036	0.926	0.879	0.839	0.817
PraNet [58]	MICCAI'20	46.9M	17.7G	0.061	0.893	0.840	0.814	0.781	0.028	0.942	0.896	0.851	0.838	0.031	0.905	0.839	0.758	0.728	0.041	0.915	0.866	0.832	0.805
PFNet [25]	CVPR'21	29.2M	61.5G	0.052	0.911	0.857	0.827	0.798	0.033	0.933	0.880	0.828	0.806	0.028	0.916	0.850	0.766	0.742	0.036	0.927	0.879	0.841	0.819
C^2FNet [28]	IJCAI'21	46.5M	18.1G	0.056	0.903	0.846	0.819	0.791	0.026	0.943	0.900	0.854	0.846	0.029	0.911	0.842	0.762	0.734	0.040	0.918	0.869	0.834	0.810
VST [24]	ICCV'21	44.5M	21.6G	0.073	0.844	0.792	0.740	0.694	0.039	0.899	0.872	0.797	0.768	0.041	0.838	0.784	0.658	0.608	0.050	0.877	0.832	0.773	0.733
SINet-V2 [2]	TPAMI'21	45.5M	23.3G	0.058	0.904	0.854	0.810	0.782	0.031	0.933	0.885	0.820	0.803	0.031	0.905	0.842	0.745	0.719	0.039	0.917	0.873	0.822	0.800
C^2FNet-V2 [29]	TCSVT'22	45.7M	23.2G	0.062	0.882	0.825	0.801	0.766	0.024	0.953	0.904	0.867	0.857	0.030	0.905	0.835	0.759	0.729	0.042	0.914	0.860	0.828	0.802
ZoomNet [37]	CVPR'22	50.2M	83.0G	0.056	0.892	0.846	0.822	0.786	0.024	0.926	0.898	0.858	0.836	0.025	0.905	0.862	0.794	0.759	0.036	0.913	0.876	0.841	0.812
SARNet	Ours	47.2M	23.1G	**0.047**	**0.927**	**0.868**	**0.850**	**0.828**	**0.021**	**0.957**	**0.912**	**0.879**	**0.871**	**0.024**	**0.931**	**0.864**	**0.800**	**0.777**	**0.032**	**0.937**	**0.886**	**0.863**	**0.842**

completely. For obscured targets (row 2 and row 5), our model can accurately identify the camouflaged target.

1) Comparison with Other State-of-the-Art Methods with Pyramid Vision Transformer (PVT) Backbone: Due to our model being designed for Pyramid Vision Transformer (PVT), we replace other state-of-the-art methods backbone with Pyramid Vision Transformer (PVT) for a fair comparison. In addition, we did not change the original structure of these models, but replaced the backbone and changed the input channels of some modules to match the output channels of the PVT backbone. Then, training and testing them with their original sets and learning strategies, and then the results are evaluated with the same evaluation tools. From Table III, compared with PVT-ZoomNet, our method has fewer parameters and FLOPs but has better performance. Specifically, F_β^w in CAMO improved by 4.2% compared with PVT-ZoomNet. In COD10K, F_β^w of our method increases by 4.8% compared with PVT-C^2FNet-V2. In NC4K, M of our method decreases by 1.0 % compared with PVT-C^2FNet-V2. F_β^w in COD10K increases by 16.9% compared to our method and VST, which also use transformer backbone. Overall, our method still outperforms previous state-of-the-art methods by large margins with the same backbone.

E. Ablation Study

We conducted a series of ablation studies to validate the effectiveness of two key components tailored for camouflaged object segmentation in our model, i.e., the Object Area Amplification module (OAA) and the Figure-Ground Conversion module (FGC). The results of the ablation studies are shown in Table IV. "B" means PVT-V2-B3 backbone with simple concatenations and upsampling with convolutions.

1) Effectiveness of OAA Module: The OAA module can magnify feature maps and make the model employ the neighbor features to complement the detail features, which helps the model to find features that are easily lost and make pixel-wise class predictions more precisely. In Table IV, compared with the model without OAA (row (a)), F_β^w of the model with OAA (row (b)) in CHAMELEON increases by 5.0%. In addition, we also insert OAA module in other state-of-the-art models, such as SINet [14], PraNet [58], and SINet-V2 [2], to illustrate its generalization properties. As can be seen from Table VI, OAA can enable them to achieve large performance gains. F_β^w of SINet [14] achieves a performance improvement of 8.5% by assembling SFR on COD10K. The performance of PraNet [58]'s M is improved by an average of 0.6% on NC4K.

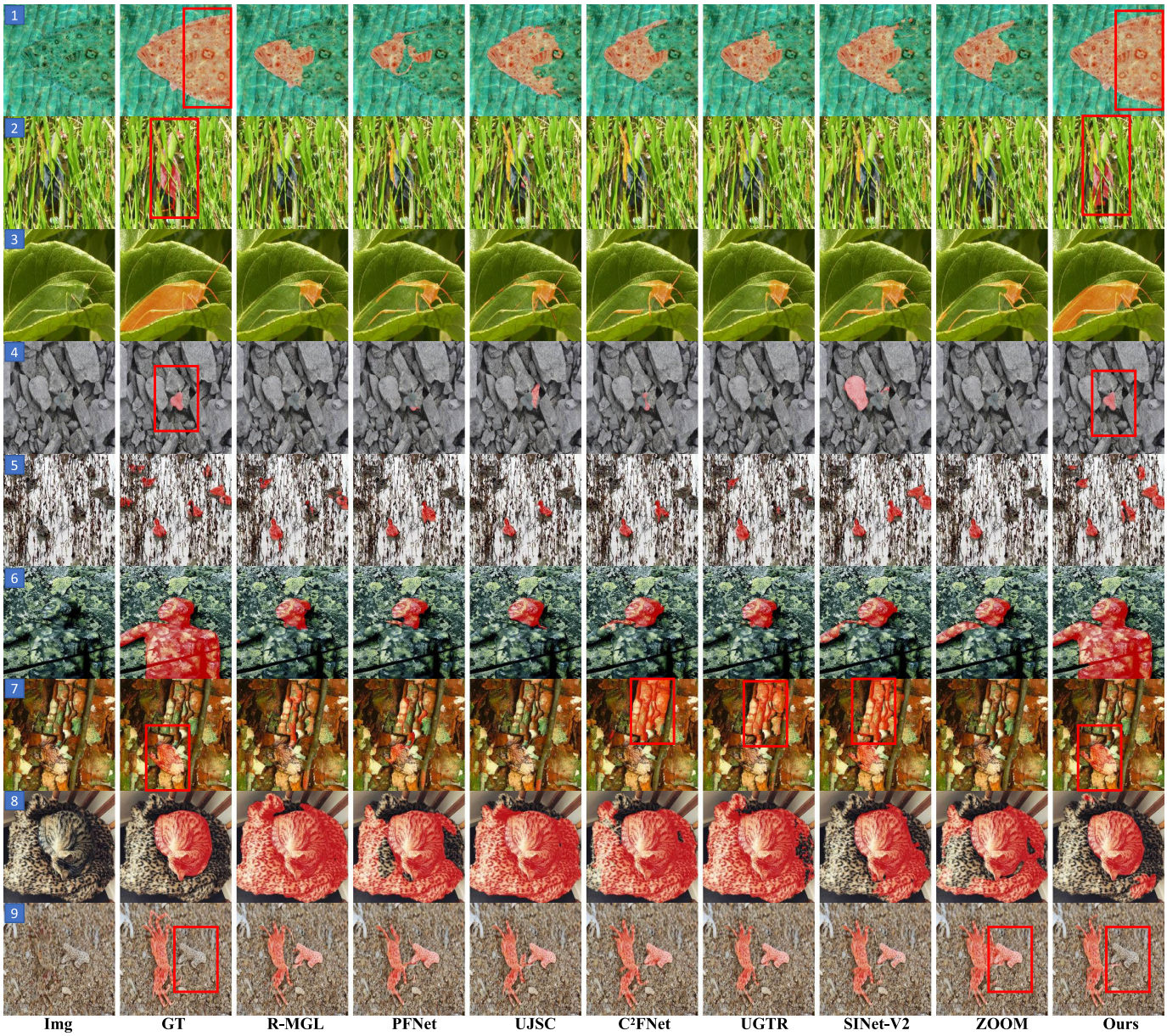


Fig. 6. Visualization comparison between our method and other state-of-the-art methods. In various disguises, such as background matching (row 1, row 3-4, and row 7-9), disruptive coloration (row 6 and row 7), a small object (row 4 and row 5), false objects (row 7 and row 9), occluded objects (row 2 and row 5), our method is much better than other methods. For better visualization, we highlight predictions on the images, and the red part of the images are predictions.

TABLE IV

ABLATION STUDY FOR THE SARNET. THE FIRST ROW "B" IN THIS TABLE SERVES AS A BASELINE. "OAA" AND "FGC" RESPECTIVELY MEANS THE OAA MODULE AND THE FGC MODULE

Method	CAMO					CHAMELEON					COD10K					NC4K				
	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$
(a) B	0.055	0.906	0.856	0.815	0.788	0.030	0.943	0.888	0.828	0.813	0.032	0.905	0.834	0.734	0.706	0.040	0.920	0.869	0.819	0.796
(b) B + OAA	0.052	0.911	0.857	0.835	0.806	0.023	0.956	0.910	0.872	0.861	0.027	0.920	0.855	0.781	0.755	0.036	0.926	0.878	0.847	0.823
(c) B + FGC	0.051	0.920	0.861	0.842	0.816	0.023	0.956	0.906	0.870	0.861	0.025	0.930	0.858	0.792	0.768	0.033	0.934	0.883	0.859	0.837
(d) B + OAA + FGC	**0.047**	**0.927**	**0.868**	**0.850**	**0.828**	**0.021**	**0.957**	**0.912**	**0.879**	**0.871**	**0.024**	**0.931**	**0.864**	**0.800**	**0.777**	**0.032**	**0.937**	**0.886**	**0.863**	**0.842**

From Fig. 6, row 2, row 4, and row 7 show that the OAA module can help the model to recognize targets (see Fig. 1).

2) Visualization Results of OAA Module: As shown in Fig. 8 column (c) and (d), the features of OAA output pay more attention to camouflage objects and less attention to distractions in the background compared with features of OAA input, which shows OAA module provides more precise attention at the feature level (The intensity of the feature

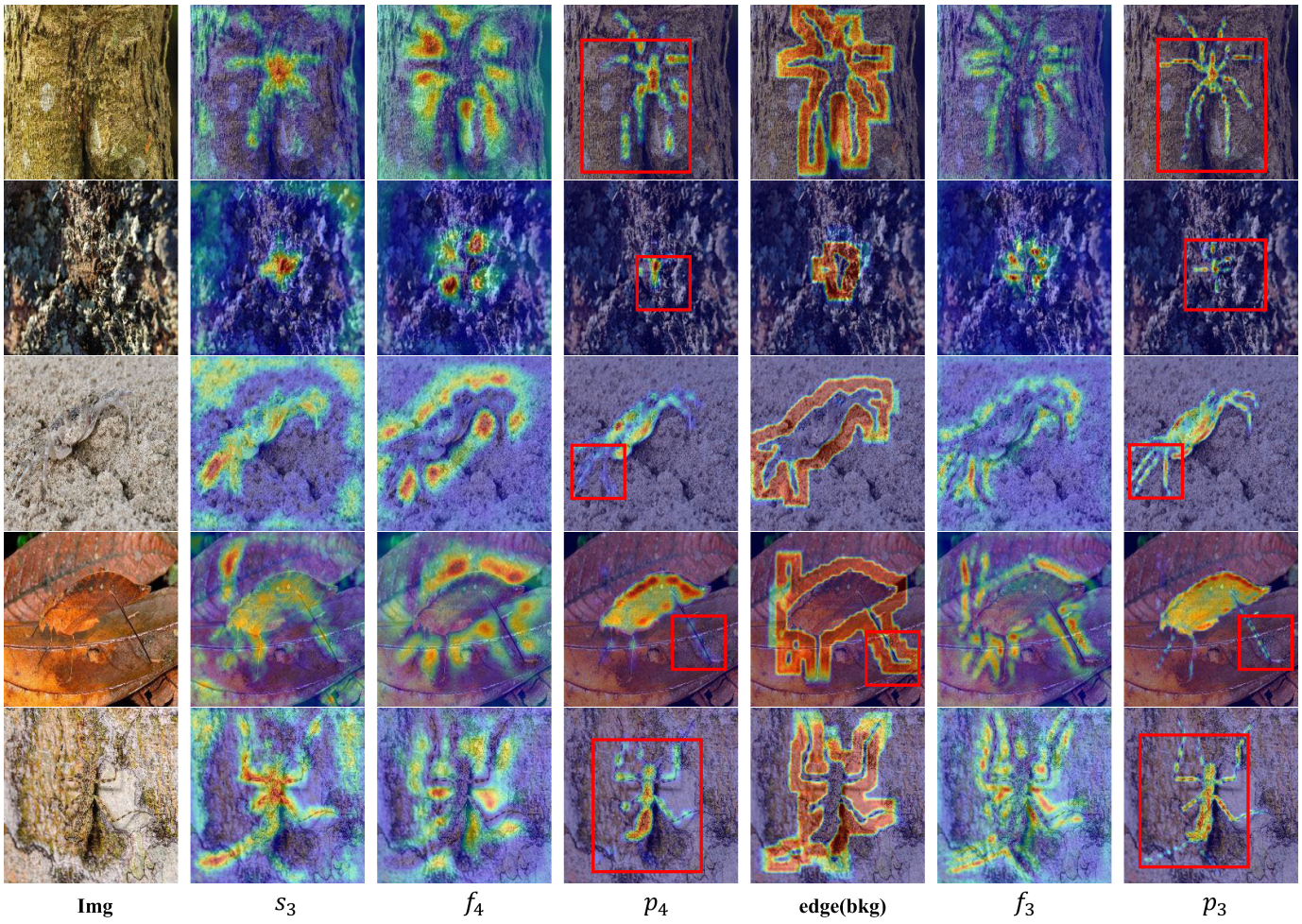


Fig. 7. Visualization of intermediate model results. Specifically, s_3 is the output of the OAA module. The input to the FGC module is s_3, f_4, p_4, and the output of the FGC module is f_3, p_3. $edge(bkg)$ is the output of the background in the FGC module.

TABLE V
ABLATION STUDY FOR THE OAA MODULE OF THE SARNET

Method	Param	FLOPs	COD10K				NC4K			
			$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\beta^w\uparrow$
(a) w\o UP	46.9M	21.0G	0.028	0.918	0.851	0.747	0.035	0.928	0.878	0.823
(b) Billinear	47.2M	23.1G	0.025	0.928	0.862	0.774	**0.032**	0.936	**0.886**	0.840
(c) Nearest	47.2M	23.1G	**0.024**	0.927	0.861	0.771	**0.032**	0.934	0.885	0.839
(d) CARAFECR [43]	51.2M	24.6G	**0.024**	0.932	**0.864**	0.776	**0.032**	0.935	0.885	0.839
(e) SAU [68]	47.4M	24.0G	0.025	0.929	0.861	0.772	**0.032**	0.937	**0.886**	0.840
(f) SAFA + SAU [68]	47.5M	24.2G	**0.024**	0.929	0.861	0.772	**0.032**	0.934	0.885	0.839
(g) OAA + 2Conv (3×3)	47.4M	23.9G	0.025	0.929	0.863	0.774	0.032	0.937	0.886	0.842
(h) OAA + CBAM	47.2M	23.1G	0.025	0.929	0.862	0.773	0.032	0.936	0.885	0.839
(i) OAA + 1Conv (3×3)	47.3M	23.4G	0.024	0.931	0.863	0.776	0.032	0.935	0.885	0.841
(j) OAA (Conv (7×7))	51.5M	28.8G	0.025	0.929	0.859	0.770	0.033	0.934	0.882	0.837
(k) OAA + ASPP	47.5M	24.1G	0.025	0.929	0.861	0.772	0.032	0.937	0.886	0.840
(l) **OAA (Bicubic)**	47.2M	23.1G	**0.024**	**0.931**	**0.864**	**0.777**	**0.032**	**0.937**	**0.886**	**0.842**

TABLE VI
QUANTIFICATION PERFORMANCE THAT OTHER SOTA METHODS WITH OR WITHOUT THE OAA MODULES

Methods	Source	COD10K				NC4K			
		$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\beta^w\uparrow$
SINet [14]	CVPR'20	0.051	0.771	0.806	0.551	0.058	0.808	0.871	0.723
OAA-SINet	Ours	**0.041**	**0.783**	**0.870**	**0.636**	**0.055**	**0.817**	**0.880**	**0.731**
PraNet [58]	MICCAI'20	0.043	0.794	0.868	0.645	0.055	0.829	0.883	0.740
OAA-PraNet	Ours	**0.036**	**0.807**	**0.872**	**0.681**	**0.049**	**0.837**	**0.887**	**0.764**
SINet-V2 [2]	TPAMI'21	0.037	0.815	0.887	0.680	0.048	0.847	0.903	0.770
OAA-SINet-V2	Ours	**0.033**	**0.822**	**0.888**	**0.699**	**0.045**	**0.850**	**0.900**	**0.780**

information can be referred to the Jet algorithm for color mapping.). In addition. as shown in Fig. 7, s_3 is the output of the OAA module, which proves that the OAA module can capture the target and its surroundings, including some detailed information, providing excellent conditions for further fine segmentation.

3) Ablation Studies on the Composition of OAA Module: Regarding the composition of the OAA module, we have also conducted extensive ablation experiments. As shown in Fig. 5, row (a) is the OAA module of the SARNet without upsampling operation (the position of "Bicubic" in 5). Row (b)-(f) and row (l) represent different methods of increasing feature resolution. Row (b), (c), and (l) are three common upsampling operations, and rows (d), (e), and (f) are three super-resolution methods based on deep learning. Compared with the results of row (a) and row (b)-(f), (l), it is obvious to find that upsampling operation or super-resolution methods can improve the performance of methods. Compared with the results of row (b)-(f) and row (l), we can find that different upsampling operations or super-resolution methods based on deep learning share a comparable performance, and taking "Bicubic" as the upsampling operation has better performance.

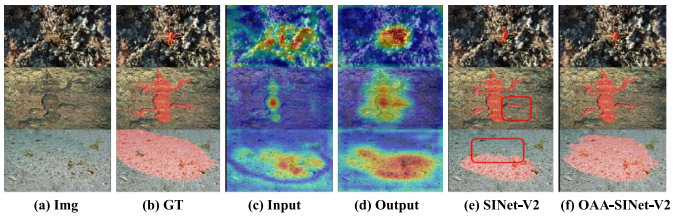


Fig. 8. The effectiveness of OAA on SINet-V2. Columns (c) and (d) are the attention map before and after the features pass through the OAA module. Columns (e) and (f) denotes the final prediction with and without the OAA module, respectively.

TABLE VII
ABLATION STUDY FOR THE FGC MODULE IN THE SARNET

Method	COD10K				NC4K			
	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\beta^w\uparrow$
(a) FGC (0000)	0.025	0.929	0.861	0.773	0.033	0.935	0.884	0.838
(b) FGC (1111)	0.025	0.928	0.862	0.775	0.032	0.935	0.885	0.840
(c) FGC (0101)	0.025	0.926	0.858	0.768	0.033	0.936	0.885	0.839
(d) FGC (1010)	0.025	0.929	0.862	0.772	0.033	0.935	0.885	0.838
(e) FGC (w\o α)	0.024	0.930	0.863	0.776	0.032	0.936	0.885	0.839
(f) FGC (s_1)	0.025	0.928	0.862	0.774	0.032	0.935	0.885	0.839
(g) FGC (3×3 Conv)	0.025	0.926	0.862	0.770	0.032	0.934	0.886	0.837
(h) **FGC(1000)**	**0.024**	**0.931**	**0.864**	**0.777**	**0.032**	**0.937**	**0.886**	**0.842**

Row (g)-(k) are ablation studies about different structures of the OAA module. Row (g) means replacing all one CBG block with two CBG blocks in the OAA module. Row (j) represents replacing all 3×3 convolution layers with 7×7 convolution layer in the OAA module. Row (h), row (i), and row (k) respectively add a $Conv$ layer, CBAM [69] layer, ASPP [70] layer operation to the final outputss_i. Compared with other complex structures (row(g)-(k)), our OAA module is simpler, more effective, and cost less.

4) Effectiveness of FGC Module: This module helps the model extract valid information from the foreground and background alternately, thus making the segmentation results more accurate. As shown in Table IV row (a) and (c), F_β^w increase by 4.8 % in CHAMELEON, F_β^w in COD10K, increases by 6.2 %. As shown in Fig. 6 (row 7 and row 8), this module allows the model to exclude better backgrounds that are very similar to the camouflage objects and remove the false positive area. From row 9 in Fig. 6, our method can recognize false negatives well, which is much better than other methods. The figure-ground conversion module allows the model to pay attention to both the foreground and the background of the images by focusing on the foreground and background, which can correct the false positive and false negative predictions.

5) Visualization Results of FGC Module: As shown in Fig. 7, s_3, f_4, p_4 are the inputs of FGC modules, f_3, p_3 are the outputs of FGC modules. Compared with p_4 and p_3, p_3 has more detail and a better performance. $edge(bkg)$ shows that the EFGC module pays more attention to the area around the target, f_4 and f_3 show the EFGC module can truly focus on the area around the target. All of these show the effectiveness of the FGC module.

6) Ablation Studies on the Composition of FGC Module: We conduct a number of experiments to find attention should be paid to the foreground or background at each level. As shown in Table VII, 0,1 separately represents the FFGC, EFGC module, i.e., row (h) FGC (1000) represents using FFGC to get f_1, EFGC to get f_2, f_3, f_4, which is our final choice. Compared with row (a) - (d), row (h) has better performance. The difference between rows (e) and (g) is that the FGC module in row (e) is without α. Row (f) means to get s_1 using the same way got s_5. In addition, row (g) is replacing the 7×7 convolution module (the convolution module calculates the final output) with the 3×3 convolution module. The results of rows (e)-(h) mean that our modules are finely designed and each part plays a role in performance improvement.

7) Effectiveness of OAA and FGC Module: Table IV row (d) has a better performance compared with row (a), row (b), and row (c), which proves that the proposed OAA and FGC modules can work well together. F_β^w of row (d) increase by 2.2% in COD10K compared with row (b). In addition, F_β^w of row (d) is 1.2% higher than that of row (c) in CAMO.

8) Effectiveness of Our Proposed New Architecture: Original state-of-the-art methods are mainly based on search-focus architecture, such as SINet [14], and SINet-V2 [2]. We proposed a new Search-Amplify-Recognize architecture, which is very promising proved by experiments. From Table VI, we can see that inserting the OAA module into other state-of-the-art methods can obviously improve the performance. Especially, S_α of SINet in COD10K increases by 6.4% if inserted the OAA module into it.

V. APPLICATION

A. Application on Polyp Segmentation

COD tasks and medical image segmentation are very similar. One of COD's downstream applications is polyp segmentation. Following PraNet, we train our model using the images from Kvasir [74] and CVC-ClinicDB [75], which consists of 1450 images (results shown on row SARNet (M) in Table VIII). Besides, we also mixed the COD training datasets in it (results shown on row SARNet (C, M) in Table VIII). Then, testing them on CVC-300 [76], ETIS [77], Kvasir [74], and CVC-ColonDB [78]. Other training settings and testing settings are the same as it in COD tasks. F_β^w of ETIS in row SARNet (C, M) increases by 1.1% compared with row SARNet (M), which illustrates that the COD dataset can improve the performance of medical image segmentation tasks to some extent.

1) Comparison with Other Polyp Segmentation Methods: In Table VIII, we compare our methods with SFA [73], UNet [71], UNet++ [72], PraNet [58], and C^2FNet-V2 [29]. The results of SFA, UNet, UNet++, and PraNet are downloaded from here. The results of C^2FNet-V2 are taken from the published paper. In addition, evaluation metrics of polyp segmentation are the same as COD tasks.

2) Performance on CVC-300: CVC-300 consists of 60 test images. Compared with PraNet, E_ϕ of our method increased by 2.3%.

3) Performance on Kvasir: Kvasir consists of 100 test images. Compared with PraNet, M of our method decreases by 1.0%.

TABLE VIII

QUANTITATIVE COMPARISON BETWEEN OUR METHOD AND OTHER SOTA METHODS FOR POLYP SEGMENTATION ON FOUR DATASETS. (M) MEANS ONLY TRAINING ON MEDICAL DATASETS, (C,M) REPRESENTS TRAINING ON COD AND MEDICAL DATASETS

Method	CVC-300					Kvasir					CVC-ColonDB					ETIS-LaribPolupDB				
	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$
UNet [71]	0.022	0.847	0.843	0.710	0.684	0.055	0.881	0.858	0.825	0.794	0.059	0.692	0.710	0.521	0.491	0.036	0.643	0.684	0.399	0.366
UNet++ [72]	0.018	0.834	0.839	0.722	0.687	0.048	0.886	0.862	0.849	0.808	0.061	0.680	0.692	0.507	0.467	0.035	0.629	0.683	0.424	0.390
SFA [73]	0.065	0.644	0.640	0.389	0.341	0.075	0.834	0.782	0.726	0.670	0.094	0.661	0.628	0.420	0.367	0.109	0.532	0.557	0.266	0.235
PraNet [58]	0.010	0.950	0.925	0.853	0.843	0.030	0.944	0.915	0.904	0.885	0.043	0.847	0.820	0.723	0.699	0.031	0.808	0.794	0.624	0.600
C^2FNet-V2 [29]	0.008	0.970	0.928	0.868	0.864	-	-	-	-	-	0.044	0.883	0.821	0.735	0.713	0.032	0.853	0.794	0.649	0.629
SARNet (M)	**0.006**	**0.973**	0.944	0.889	0.888	0.020	0.957	0.925	0.904	0.892	0.027	**0.924**	0.885	0.836	0.818	0.026	0.909	0.876	0.806	0.788
SARNet (C,M)	**0.006**	0.971	**0.947**	**0.896**	**0.893**	**0.016**	**0.960**	**0.932**	**0.907**	**0.897**	**0.025**	**0.924**	**0.894**	**0.846**	**0.830**	**0.024**	**0.912**	**0.884**	**0.815**	**0.799**

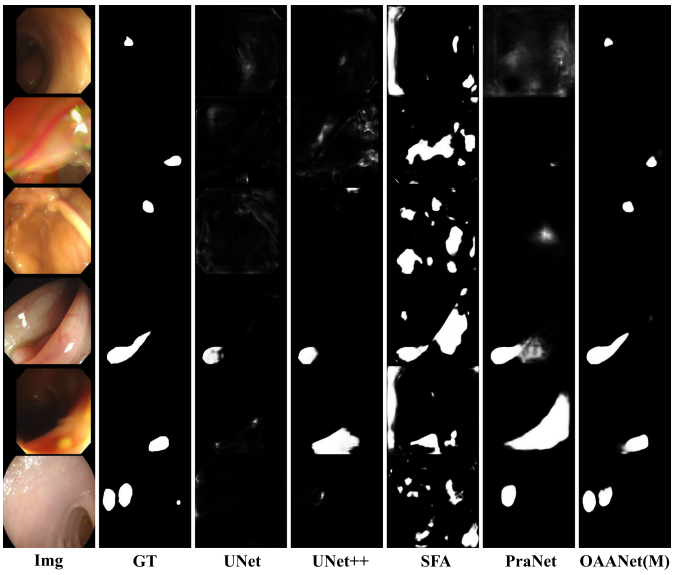


Fig. 9. Visual comparison between different methods on the polyp segmentation task.

4) Performance on CVC-ColonDB: CVC-ClinicDB consists of 380 test images. Compared with C^2FNet-V2, F_β^w of our method increases by **10.5%**.

5) Performance on ETIS: ETIS-laribPolupDB consists of 196 test images. Compared to C^2FNet-V2, F_β^w of our method increases by **15.9%**.

6) Qualitative Evaluation: As shown in Fig. 9, row (1)-(3) are small polyps. Our method can identify the targets precisely, while other methods fail. For similar medium objects (row 4 and row 5), our method identifies the object completely, while other methods lose some parts of the target. Row 6 is multi-objects, our method can identify the multi-objects completely, while UNet, UNet++, and SFA lose the objects, and PraNet only identifies one object. Overall, our methods outperform other methods on the polyp segmentation task.

B. Application on Video Camouflaged Object Detection

To further validate the generalizability and application value of our model, we evaluate our proposed SARNet on Video Camouflaged Object Detection (VCOD), which is the largest dataset of VCOD tasks now. We evaluate our method on MoCA-Mask [81] datasets. MoCA-Mask [81] relabeled the datasets MoCA [82], which consists of 37K frames from 141 YouTube Videos.

TABLE IX

QUANTITATIVE COMPARISON WITH OTHER SOTA METHODS WITHOUT PSEUDO ON VCOD. BOLD INDICATES OUR METHODS. * MEANS ONLY TRAINING ON COD DATASETS

Method	Source	MoCA-Mask				
		$M\downarrow$	$E_\phi\uparrow$	$S_\alpha\uparrow$	$F_\phi\uparrow$	$F_\beta^w\uparrow$
RCRNe [79]	ICCV'2019	0.026	0.536	0.558	0.158	0.138
MG [80]	ICCV'2021	0.078	0.501	0.502	0.130	0.112
SINet-V2 [2]	TPAMI'21	0.024	0.623	0.596	0.232	0.206
SLT-Net-long [81]	CVPR'22	0.019	0.741	0.624	0.313	0.292
SLT-Net-short [81]	CVPR'22	0.019	0.714	0.631	0.310	0.285
SARNet*	Ours	**0.018**	**0.704**	**0.640**	**0.338**	**0.318**
SARNet	Ours	**0.014**	**0.713**	**0.676**	**0.379**	**0.353**

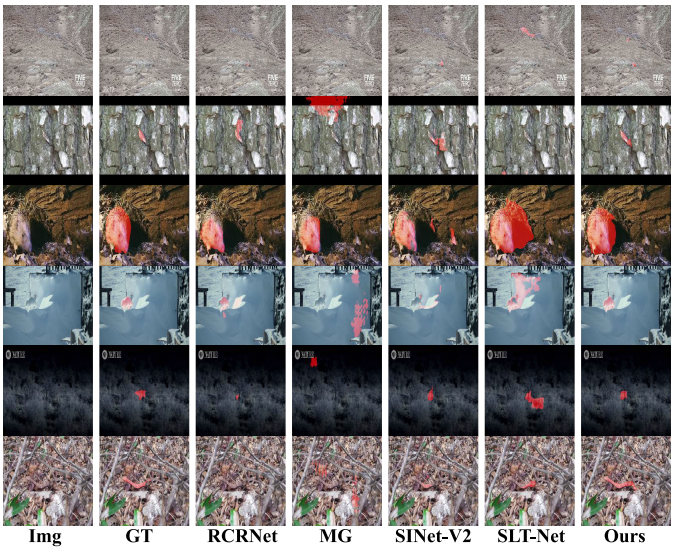


Fig. 10. Visual comparison of different methods on the video camouflaged object detection task.

1) Comparison with Other Video Camouflaged Object Detection Methods: We compared our proposed SARNet with other methods including RCRNet [79], MG [80], SINet-V2 [2], SLT-Net (SOTA methods published on CVPR 2022 and using PVT-V2-b5 as backbone) [81] as shown on Table IX. Our method (SARNet *) achieve state-of-the-art performance. One point worth noting is that our model (SARNet*) doesn't train on VCOD datasets while SLT-Net training on both COD and VCOD datasets. When training SARNet on both COD and VCOD datasets, the performance is better as row

(SARNet) shown on Table IX. Compared with SLT-Net-long, F_ϕ of our method (SARNet) increases by 6.6 %, thus further illustrating the validity of our method on the VCOD task. Visual comparison as shown in Fig. 10 also illustrates the superiority of our method.

VI. Conclusion

In this paper, we discuss the effect of the resolution of the input image on the COD task and find that the COD task is extremely sensitive to the resolution of the input image. To the best of our knowledge, we are the first to introduce that increasing the resolution (feature-level or image-level) is an important direction in camouflaged detection to help the model improve performance. Furthermore, we develop a promising Search-Amplify-Recognize architecture and design an object area amplification network for camouflaged object detection (SARNet). According to this paradigm, we carefully design the object area amplification module and the figure-ground conversion module serves as the Amplify part and Recognize part. Our model outperforms previous state-of-the-art methods by large margins on all benchmarks of camouflaged object detection. Moreover, the proposed model also outperforms previous state-of-the-art methods on poly segmentation and the VCOD task, which shows that our model has great potential for downstream tasks in camouflaged object detection.

References

[1] J. Skelhorn and C. Rowe, "Cognition and the evolution of camouflage," *Proc. Roy. Soc. B, Biol. Sci.*, vol. 283, no. 1825, Feb. 2016, Art. no. 20152890.

[2] D.-P. Fan, G.-P. Ji, M.-M. Cheng, and L. Shao, "Concealed object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 6024–6042, Oct. 2022.

[3] M. Stevens and S. Merilaita, "Animal camouflage: Current issues and new perspectives," *Phil. Trans. Roy. Soc. B, Biol. Sci.*, vol. 364, no. 1516, pp. 423–427, Feb. 2009.

[4] Y.-H. Wu et al., "JCS: An explainable COVID-19 diagnosis system by joint classification and segmentation," *IEEE Trans. Image Process.*, vol. 30, pp. 3113–3126, 2021.

[5] L. Li, B. Dong, E. Rigall, T. Zhou, J. Dong, and G. Chen, "Marine animal segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 4, pp. 2303–2314, Apr. 2022.

[6] S. Ge, X. Jin, Q. Ye, Z. Luo, and Q. Li, "Image editing by object-aware optimal boundary searching and mixed-domain composition," *Comput. Vis. Media*, vol. 4, no. 1, pp. 71–82, Mar. 2018.

[7] H. Mei et al., "Exploring dense context for salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 3, pp. 1378–1389, Mar. 2022.

[8] X. Hu, C.-W. Fu, L. Zhu, T. Wang, and P.-A. Heng, "SAC-Net: Spatial attenuation context for salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 3, pp. 1079–1090, Mar. 2021.

[9] L. Zhang, Q. Zhang, and R. Zhao, "Progressive dual-attention residual network for salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 9, pp. 5902–5915, Sep. 2022.

[10] R. Cong et al., "A weakly supervised learning framework for salient object detection via hybrid labels," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 2, pp. 534–548, Feb. 2023.

[11] R. Cong et al., "Global-and-local collaborative learning for co-salient object detection," 2022, *arXiv:2204.08917*.

[12] R. Cong et al., "Does thermal really always matter for RGB-T salient object detection?" *IEEE Trans. Multimedia*, early access, Oct. 21, 2022, doi: 10.1109/TMM.2022.3216476.

[13] D. Zhang, H. Tian, and J. Han, "Few-cost salient object detection with adversarial-paced learning," in *Proc. NIPS*, vol. 33, 2020, pp. 12236–12247.

[14] D.-P. Fan, G.-P. Ji, G. Sun, M.-M. Cheng, J. Shen, and L. Shao, "Camouflaged object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 2777–2787.

[15] K. Chen et al., "Hybrid task cascade for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jul. 2019, pp. 4974–4983.

[16] N. Liu, J. Han, and M.-H. Yang, "PiCANet: Learning pixel-wise contextual attention for saliency detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3089–3098.

[17] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "BASNet: Boundary-aware salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7479–7489.

[18] T. Zhao and X. Wu, "Pyramid feature attention network for saliency detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3085–3094.

[19] J. Zhao, J.-J. Liu, D.-P. Fan, Y. Cao, J. Yang, and M.-M. Cheng, "EGNet: Edge guidance network for salient object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8779–8788.

[20] J. Wei, S. Wang, and Q. Huang, "F^3Net: Fusion, feedback and focus for salient object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 12321–12328.

[21] Z. Chen, Q. Xu, and R. Cong, "Global context-aware progressive aggregation network for salient object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 10599–10606.

[22] R. Cong et al., "CIR-Net: Cross-modality interaction and refinement for RGB-D salient object detection," *IEEE Trans. Image Process.*, vol. 31, pp. 6800–6815, 2022.

[23] R. Cong, Y. Zhang, L. Fang, J. Li, Y. Zhao, and S. Kwong, "RRNet: Relational reasoning network with parallel multiscale attention for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5613311.

[24] N. Liu, N. Zhang, K. Wan, L. Shao, and J. Han, "Visual saliency transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 4722–4732.

[25] H. Mei, G.-P. Ji, Z. Wei, X. Yang, X. Wei, and D.-P. Fan, "Camouflaged object segmentation with distraction mining," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 8772–8781.

[26] J. Ren et al., "Deep texture-aware features for camouflaged object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 3, pp. 1157–1167, Mar. 2023.

[27] J. Zhu, X. Zhang, S. Zhang, and J. Liu, "Inferring camouflaged objects by texture-aware interactive guidance network," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 4, pp. 3599–3607.

[28] Y. Sun, G. Chen, T. Zhou, Y. Zhang, and N. Liu, "Context-aware cross-level fusion network for camouflaged object detection," in *Proc. 30th Int. Joint Conf. Artif. Intell.*, Aug. 2021, pp. 1025–1031.

[29] G. Chen, S.-J. Liu, Y.-J. Sun, G.-P. Ji, Y.-F. Wu, and T. Zhou, "Camouflaged object detection via context-aware cross-level fusion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 10, pp. 6981–6993, Oct. 2022.

[30] N. Kajiura, H. Liu, and S. Satoh, "Improving camouflaged object detection with the uncertainty of pseudo-edge labels," in *Proc. ACM Multimedia Asia*, Dec. 2021, pp. 1–7.

[31] Q. Zhai, X. Li, F. Yang, C. Chen, H. Cheng, and D.-P. Fan, "Mutual graph learning for camouflaged object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 12997–13007.

[32] J. Wagemans et al., "A century of gestalt psychology in visual perception: I. Perceptual grouping and figure–ground organization," *Psychol. Bull.*, vol. 138, no. 6, pp. 1172–1217, 2012.

[33] W. Köhler, "Gestalt psychology," *Psychologische Forschung*, vol. 31, no. 1, pp. 18–30, 1967.

[34] K. Koffka, *Principles of Gestalt Psychology*. Evanston, IL, USA: Routledge, 2013.

[35] Y. Lv et al., "Simultaneously localize, segment and rank the camouflaged objects," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11591–11601.

[36] A. Li, J. Zhang, Y. Lv, B. Liu, T. Zhang, and Y. Dai, "Uncertainty-aware joint salient object and camouflaged object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10071–10081.

[37] Y. Pang, X. Zhao, T.-Z. Xiang, L. Zhang, and H. Lu, "Zoom in and out: A mixed-scale triplet network for camouflaged object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 2160–2170.

[38] X. Zhang, C. Zhu, S. Wang, Y. Liu, and M. Ye, "A Bayesian approach to camouflaged moving object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 9, pp. 2001–2013, Sep. 2017.

[39] H. Bi, C. Zhang, K. Wang, J. Tong, and F. Zheng, "Rethinking camouflaged object detection: Models and datasets," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 9, pp. 5708–5724, Sep. 2022.

[40] W. Shi et al., "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1874–1883.

[41] Z. Tian, T. He, C. Shen, and Y. Yan, "Decoders matter for semantic segmentation: Data-dependent decoding enables flexible feature aggregation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3126–3135.

[42] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, and J. Sun, "Meta-SR: A magnification-arbitrary network for super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1575–1584.

[43] J. Wang, K. Chen, R. Xu, Z. Liu, C. C. Loy, and D. Lin, "CARAFE: Content-aware reassembly of features," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3007–3016.

[44] S.-H. Lee and S.-H. Bae, "SRF-GAN: Super-resolved feature GAN for multi-scale representation," 2020, *arXiv:2011.08459*.

[45] Y. Bai, Y. Zhang, M. Ding, and B. Ghanem, "SOD-MTGAN: Small object detection via multi-task generative adversarial network," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 206–221.

[46] J. Rabbi, N. Ray, M. Schubert, S. Chowdhury, and D. Chao, "Small-object detection in remote sensing images with end-to-end edge-enhanced GAN and object detector network," *Remote Sens.*, vol. 12, no. 9, p. 1432, May 2020.

[47] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, "Perceptual generative adversarial networks for small object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1222–1230.

[48] T. R. Shaham, T. Dekel, and T. Michaeli, "SinGAN: Learning a generative model from a single natural image," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4570–4580.

[49] J. Noh, W. Bae, W. Lee, J. Seo, and G. Kim, "Better to follow, follow to be better: Towards precise supervision of feature super-resolution for small object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9725–9734.

[50] Y. Li, W. Cai, Y. Gao, C. Li, and X. Hu, "More than encoder: Introducing transformer decoder to upsample," 2021, *arXiv:2106.10637*.

[51] Q. Zheng, X. Qiao, Y. Cao, and R. W. H. Lau, "Distraction-aware shadow detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5167–5176.

[52] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 234–250.

[53] H. Xiao, J. Feng, Y. Wei, M. Zhang, and S. Yan, "Deep salient object detection with dense connections and distraction diagnosis," *IEEE Trans. Multimedia*, vol. 20, no. 12, pp. 3239–3251, Dec. 2018.

[54] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan, and W. Hu, "Distractor-aware Siamese networks for visual object tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 101–117.

[55] Q. Huang, C. Wu, C. Xia, Y. Wang, and C.-C.-J. Kuo, "Semantic segmentation with reverse attention," in *Proc. Brit. Mach. Vis. Conf.*, 2017, pp. 4–7.

[56] W. Wang et al., "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 568–578.

[57] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," 2016, *arXiv:1606.08415*.

[58] D.-P. Fan et al., "PraNet: Parallel reverse attention network for polyp segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2020, pp. 263–273.

[59] F. Yang et al., "Uncertainty-guided transformer reasoning for camouflaged object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 4146–4155.

[60] G.-P. Ji, D.-P. Fan, Y.-C. Chou, D. Dai, A. Liniger, and L. Van Gool, "Deep gradient learning for efficient camouflaged object detection," *Mach. Intell. Res.*, vol. 20, no. 1, pp. 92–108, Jan. 2023.

[61] G.-P. Ji, L. Zhu, M. Zhuge, and K. Fu, "Fast camouflaged object detection via edge-based reversible re-calibration network," *Pattern Recognit.*, vol. 123, Mar. 2022, Art. no. 108414.

[62] P. Skurowski, H. Abdulameer, J. Błaszczyk, T. Depta, A. Kornacki, and P. Kozieł, "Animal camouflage analysis: Chameleon database," *Unpublished Manuscript*, vol. 2, no. 6, p. 7, 2018.

[63] T.-N. Le, T. V. Nguyen, Z. Nie, M.-T. Tran, and A. Sugimoto, "Anabranch network for camouflaged object segmentation," *Comput. Vis. Image Understand.*, vol. 184, pp. 45–56, Jul. 2019.

[64] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4548–4557.

[65] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, and A. Borji, "Enhanced-alignment measure for binary foreground map evaluation," 2018, *arXiv:1805.10421*.

[66] D.-P. Fan, G.-P. Ji, X. Qin, and M.-M. Cheng, "Cognitive vision inspired object segmentation metric and loss function," *Scientia Sinica Inf.*, vol. 51, no. 9, p. 1475, Sep. 2021.

[67] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 248–255.

[68] L. Wang, Y. Wang, Z. Lin, J. Yang, W. An, and Y. Guo, "Learning a single network for scale-arbitrary super-resolution," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 4801–4810.

[69] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

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

[71] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.

[72] Z. Zhou et al., "UNet++: Redesigning skip connections to exploit multiscale features in image segmentation," *IEEE Trans. Med. Imag.*, vol. 39, no. 6, pp. 1856–1867, Dec. 2019.

[73] Y. Fang, C. Chen, Y. Yuan, and K.-Y. Tong, "Selective feature aggregation network with area-boundary constraints for polyp segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2019, pp. 302–310.

[74] D. Jha et al., "Kvasir-SEG: A segmented polyp dataset," in *Proc. Int. Conf. Multimedia Modeling*, 2020, pp. 451–462.

[75] J. Bernal et al., "WM-DOVA maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians," *Comput. Med. Imag. Graph.*, vol. 43, pp. 99–111, Jul. 2015.

[76] D. Vázquez et al., "A benchmark for endoluminal scene segmentation of colonoscopy images," *J. Healthcare Eng.*, vol. 2017, pp. 1–9, Jan. 2017.

[77] J. Silva, A. Histace, O. Romain, X. Dray, and B. Granado, "Toward embedded detection of polyps in WCE images for early diagnosis of colorectal cancer," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 9, no. 2, pp. 283–293, 2014.

[78] N. Tajbakhsh, S. R. Gurudu, and J. Liang, "Automatic polyp detection in colonoscopy videos using an ensemble of convolutional neural networks," in *Proc. IEEE 12th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2015, pp. 79–83.

[79] P. Yan et al., "Semi-supervised video salient object detection using pseudo-labels," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7284–7293.

[80] C. Yang, H. Lamdouar, E. Lu, A. Zisserman, and W. Xie, "Self-supervised video object segmentation by motion grouping," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7177–7188.

[81] X. Cheng et al., "Implicit motion handling for video camouflaged object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 13864–13873.

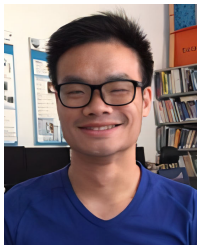
[82] H. Lamdouar, C. Yang, W. Xie, and A. Zisserman, "Betrayed by motion: Camouflaged object discovery via motion segmentation," in *Proc. Asian Conf. Comput. Vis.*, 2020, pp. 488–503.

Haozhe Xing is currently pursuing the master's degree with the School of Engineering and Technology, Fudan University, Shanghai, China. His research interests include computer vision, camouflaged object detection, and salient object detection.

Shuyong Gao is currently pursuing the Ph.D. degree with the School of Computer Science, Fudan University, Shanghai, China. His research interests include computer vision, multimodal information processing, image segmentation, and salient object detection.

Yan Wang received the Ph.D. degree from Fudan University, China, in January 2023. He is currently a Post-Doctoral Researcher with the Fudanroilab, Academy for Engineering and Technology, Fudan University. His research interests include affective computing, multimedia analysis, and unsupervised domain adaptation learning.

Xujun Wei is currently pursuing the master's degree with the School of Engineering and Technology, Fudan University, Shanghai, China. His research interests include computer vision, few-shot object detection, and medical image segmentation.



Hao Tang received the master's degree from the School of Electronics and Computer Engineering, Peking University, China, and the Ph.D. degree from the Multimedia and Human Understanding Group, University of Trento, Italy. He was a Visiting Scholar with the Department of Engineering Science, University of Oxford. He is currently a Post-Doctoral Researcher with the Computer Vision Laboratory, ETH Zurich, Switzerland. His research interests include deep learning, machine learning, and their applications to computer vision.

Wenqiang Zhang received the Ph.D. degree in mechanical engineering from Shanghai Jiao Tong University, China, in 2004. He is a Professor with the School of Computer Science, Fudan University. His current research interests include computer vision and robot intelligence.