Pattern Recognition 172 (2026) 112637

Contents lists available at ScienceDirect

Pattern Recognition

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

Partitioned observation network for camouflaged object detection

Jinxia Zhang[a,b], Yin Yuan[a], Xuwen Zhu[a], Yang Hu[a], Kaihua Zhang[a,*]

[a] Key Laboratory of Measurement and Control of CSE, Ministry of Education, School of Automation, Southeast University, Nanjing, 210096, China
[b] Advanced Ocean Institute of Southeast University, Nantong, 226010, China

ARTICLE INFO

Keywords:
Camouflaged object don
Image segmentation
Human visual system

ABSTRACT

Camouflaged Object Detection (COD) aims to delineate the boundaries between the Camouflaged Objects (COs) and their surrounding backgrounds explicitly. However, an excessive focus on the CO boundaries may make the features insufficient for the low-frequency information, thereby leading to the loss of semantics and incomplete segmentation. This issue can be naturally addressed by perceptual models of the human visual system, which first prioritize the CO within the central vision, and subsequently extend attention to the CO in the peripheral vision. Inspired by this, we propose a Partitioned Observation Network (PONet) for COD. The PONet comprises two branches that separately focus on the low-frequency semantics of the CO's central region and the high-frequency textures of the CO's peripheral region. To fully extract and reinforce the low-frequency semantics of the CO's central region, a Global Semantic Enhancement (GSE) module is designed within the central region branch. Through stacking the GSE modules, the low-frequency information in the features is progressively refined and supplemented, achieving the rendering of the CO's central region. To capture the high-frequency features of the CO's peripheral region and to accentuate the contrast between the CO and its surrounding backgrounds, we design a Local Detail Extraction (LDE) module and a weighted peripheral region loss within the peripheral region branch. Guided by the semantics of the central region, the high-resolution details of the target are progressively restored, enhancing the discriminative high-frequency features of the peripheral region. Finally, to complement and integrate the low-frequency and high-frequency information from both regions, a Region Adaptive Fusion (RAF) module is proposed to achieve a complete "*observation*" of the CO. The PONet is evaluated on four challenging benchmarks, which achieves superior performance over 35 state-of-the-art methods under four widely-used evaluation metrics.

1. Introduction

Camouflage concerns color similarity or disruptive textural patterns between camouflaged objects and their surrounding backgrounds. Camouflaged Object Detection (COD) is a highly researched topic to counteract camouflage strategies and distinguish objects from their backgrounds. It has significant implications and extensive practical applications in various fields, including pest detection in agriculture, enemy reconnaissance in the military, and polyp segmentation in medicine.

Early COD methods rely on handcrafted low-level features such as texture characteristics to separate foreground and background. Some studies use salient object detection models directly to detect COs. However, the effectiveness of these approaches is quite limited due to the weak representation capability. A large-scale benchmark COD10K proposed by Fan et al. [1] propels the transition of COD methods from traditional computer vision to deep learning era. COD encounters challenges, particularly in preserving boundaries. Some methods incorporate auxiliary cues like gradient [4] and multi-view assistance [11] to get prior knowledge, compensating for the loss of boundary features. A lot of methods design boundary extraction modules [3] or utilize features in reverse [6] to supplement the missing boundaries. However, these approaches still obtain unclear boundaries, sacrifice some semantic information or amplify background interference while intensively focusing on the boundaries. According to central vision and peripheral vision in human visual attention [27], the object of interest in the eye's central vision is observed first. To observe the details in the peripheral vision, attention needs to be shifted to this region. Taking inspiration from this mechanism, a Partitioned Observation Network (PONet) is proposed to explicitly model and synergize the partitioned regions of camouflaged object: the central region, which is the main part of the target with a rich presence of low-frequency information, and the peripheral region, which is the neighborhood region near the target boundary with a substantial amount of high-frequency information. As shown in the features in Fig. 1, PONet aims to extract the low-frequency semantics of the central region and the high-frequency details of the peripheral region separately and achieve complete segmentation by integrating the

* Corresponding author.
E-mail address: zhkhua@gmail.com (K. Zhang).

https://doi.org/10.1016/j.patcog.2025.112637
Received 13 December 2024; Received in revised form 20 September 2025; Accepted 18 October 2025
Available online 24 October 2025

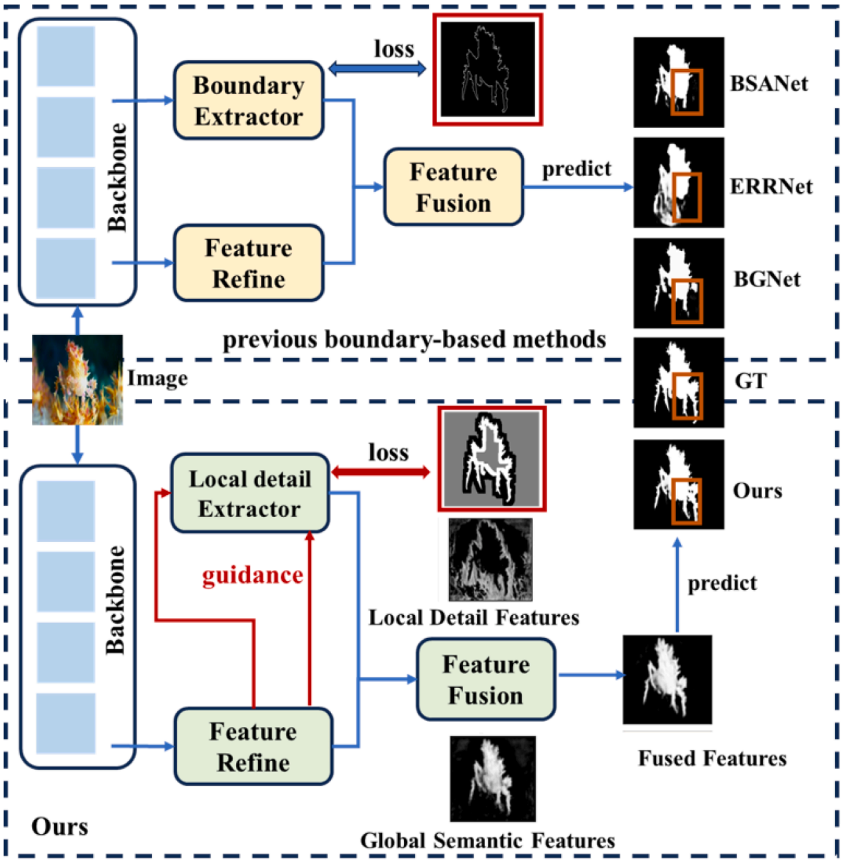


Fig. 1. Comparing with previous boundary-based methods, our method can make better use of low-frequency target semantics and high-frequency local details, thus obtaining complete segmentation results.

two regions. PONet diverges from boundary-based methods by avoiding direct extraction of high-frequency boundary features from the backbone. Guided by semantic features, our method restores refined features' high-frequency details, thus preserving semantic information in the features. Furthermore, PONet concentrates constraints on peripheral regions, maximizing the contrast between the target and the background.

Specifically, our PONet consists of two main branches: the central region branch and the peripheral region branch, specializing in the low-frequency semantics of the central region and high-frequency details of the peripheral region respectively. The central region branch employs cross-scale connections to capture multi-scale semantic information and enhances the semantics through the Global Semantic Enhancement (GSE) module. The stacked GSE modules can help progressively enrich and enhance semantic information of the target's central region. In the peripheral region branch, we propose the Local Detail Extraction (LDE) module for extracting local details under the guidance of semantic information. Through this, the features of the peripheral region are progressively refined. A weighted peripheral region loss is proposed to ensure a balanced consideration of the targets with various sizes and shapes. Ultimately, the features of different regions from these two branches are adaptively fused with a Region Adaptive Fusion (RAF) module to completely segment the target from the indistinguishable background. To sum up, our contributions are as follows:

1. A novel PONet is designed to implement partitioned focus on the target's central region and peripheral region, and explicitly model the multi-region synergy between these two regions for camouflaged object detection.
2. A GSE module and an LDE module are put forward to enhance low-frequency semantic information in the central region and extract high-frequency detailed features in the peripheral region. An RAF module is proposed to jointly utilize these two discriminative cues.
3. A weighted peripheral region loss is proposed to constrain the peripheral regions of objects with different sizes and maximize the contrast between the target and the background.
4. Extensive experimental results demonstrate that our method outperforms 35 state-of-the-art approaches on three benchmark datasets.

2. Related work

Traditional pattern recognition based COD. Camouflaged object detection (COD) aims to identify camouflage patterns and completely separate the target from the background. In the early stage, traditional visual techniques were employed, including handcrafted patterns such as motion clues as well as heuristic priors like color, texture, and intensity. Boult et al. [28] developed a background subtraction technique with two thresholds to detect camouflaged objects. Nagabhushan and Bhajantri [29] proposed a method with a co-occurrence matrix and a Canny edge detector to analyze texture and boundaries. Later, some studies applied the salient object detection model to detect camouflaged objects. However, the effectiveness of these methods was limited.

Bionics-based COD. Recently, the focus of some studies shifted to bionics-based approaches. Inspired by animal hunting, Fan et al. [1] and Mei et al. [2] first located and then recognized camouflaged objects. SegMaR [10], also bio-inspired, iteratively refined rough predictions by sampling and zooming in on the predicted locations. Peng et al. [11] mimicked human observation from different viewpoints, introducing auxiliary inputs of varying sizes or views. Li et al. [12], Liu et al. [13] and Chen et al. [14] developed attention modules or multi-scale fusion modules to enhance target semantics and fuse multi-scale contextual information. Yang et al. [35] proposed a method that can effectively

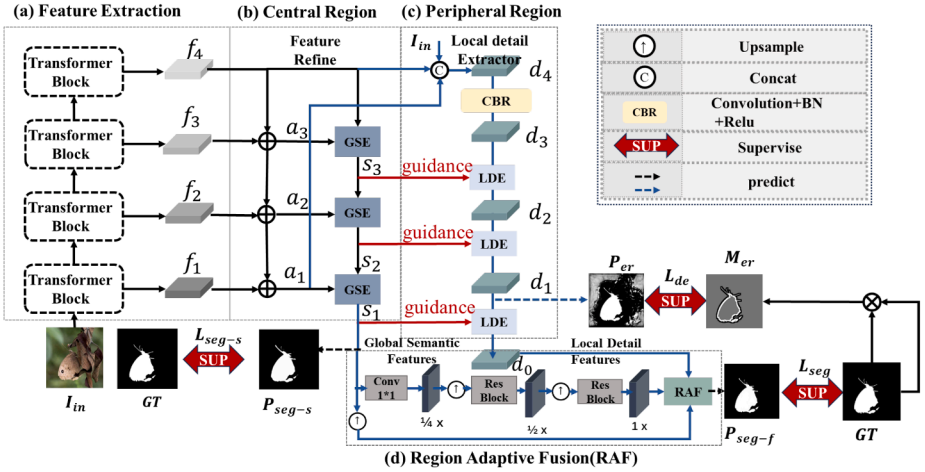


Fig. 2. The overall architecture of the proposed PONet: (a) the feature extraction part with a Transformer backbone; (b) the central region branch consists of GSE modules; (c) the peripheral region branch with LDE modules; (d) RAF module for features from the two branches.

improve the accuracy of camouflaged object detection through multi-angle feature fusion and self-similarity constraints. These works have demonstrated that fully leveraging biological cognitive characteristics helps improve the performance of camouflaged object detection.

Auxiliary information-based COD. Some studies focus on utilizing auxiliary information, such as boundaries and gradients to enhance the segmentation of camouflaged objects. Sun et al [3] explored extra object-related boundary semantics to guide representation learning of COD. Ji et al. [36] introduced the edge-based reversible re-calibration network, which leverages boundary information through selective edge aggregation and reversible re-calibration units. Similarly, Ji et al. [5] and Zhu et al. [6] extracted boundaries and progressively refined boundary features to get comprehensive priors. Zhai et al. [16] decomposed images into feature maps for specific tasks to coarsely locate targets and capture their boundary details with graphs. He et al. [17] utilized high-order differential equation solvers to reconstruct precise and complete boundary prediction maps. Wang et al. [31] and Xiang et al. [7] explored the contributions of depth information in segmenting camouflaged objects. Zhao et al. [47] proposed a background-free camouflaged image generation method using latent knowledge retrieval and diffusion modeling to generate diverse, realistic camouflage images without manual input. Luo et al. [48] used 2D prompts and discrimination loss in an encoder-decoder framework to enhance model adaptability and integrate multimodal information for camouflaged object detection. More recently, Yin et al. [49] proposed AdaptCOD, which leverages a context adaptive partition and background retrieval to refine boundaries. Yue et al. [50] proposed PRBE-Net, which progressively integrates region and boundary cues with tailored modules. Zhao et al. [51] proposed FLR-Net, a three-stage framework that leverages filtering, localization, and refinement to enhance boundary cues. Nevertheless, these approaches show limitations in effectively modeling the intricate interactions between auxiliary information and global semantics, often resulting in sub-optimal segmentation in challenging scenarios.

Transformer-based COD. Vision Transformers can make full use of the global information of the image and sparked new ideas for COD. Yang et al. [15] enhanced the model's reasoning capabilities and mined high-level semantics with Transformer. Hu et al. [18] proposed a high-resolution iterative feedback network to exploit the high-resolution information and refine the low-resolution features with high-resolution knowledge. Liu et al. [19] fine-tuned pre-trained models with visual cues to enforce the tunable parameters focusing on the explicit visual content from each individual image. However, these approaches continue to face challenges, including incomplete segmentation, inadequate delineation of boundary regions, and excessive interference. In this paper, inspired by human visual attention, a novel method is developed via partitioned observation of the target's central and peripheral regions. A GSE module is proposed to enhance low-frequency semantics in the central region and an LDE module is proposed to extract high-frequency details in the peripheral region under the guidance of semantic information, which are then effectively fused with the proposed Region Adaptive Fusion module.

3. Proposed method

In this section, we first present the overall architecture of the proposed PONet, then illustrate the details of each module and the total loss of the network.

3.1. Overall architecture

The overall architecture of the proposed PONet is illustrated in Fig. 2. Given an input image $I_{in} \in \mathbb{R}^{3\times H\times W}$, PVT-v2 [26] is employed as the backbone to extract multi-scale features of I_{in}. The channel numbers of these features are reduced to 64 through 1×1 convolutions. These 64-dimension features of the scale i are denoted as $\{f_i \in \mathbb{R}^{64\times \frac{H}{2^{i+1}}\times \frac{W}{2^{i+1}}}\}_{i=1}^{4}$.

Subsequently, in the central region branch, we perform up-to-down connections between different scales of features f_i to get cross-scale features $\{a_i\}_{i=1}^{3}$. Following this, a stack of GSE modules are proposed to fuse the features from two adjacent scales and produce semantic-enhanced features denoted as $\{s_i\}_{i=1}^{3}$. And s_1 is convolved into a single channel as the output P_{seg-s} of the central region branch.

In the peripheral region branch, we employ f_4 as global feature and a_1 as local feature. They are upsampled to the same resolution as the input image I_{in} and concatenated with I_{in} as input d_4, which is convolved to obtain high-resolution feature d_3 of the target region from I_{in}. Next, $\{s_i\}_{i=1}^{3}$ are used as semantic guidance to assist in the extraction of detailed features in the peripheral region through LDE modules. This process generates the peripheral detail feature denoted as d_1, which is then convolved into a single channel as the peripheral region prediction P_{er}.

Finally, the semantic feature s_1 is directly resized and iteratively resized respectively through upsample layers and residual blocks to the same resolution as the detail feature d_0. These upsampled semantic features and d_0 are adaptively fused by the Region Adaptive Fusion (RAF) module to generate the ultimate prediction map P_{seg-f}. P_{er} is supervised by a peripheral region map with our peripheral region loss while P_{seg-s} and P_{seg-f} are supervised by ground truth with weighted binary cross entropy (BCE) loss and weighted intersection over union (IoU) loss.

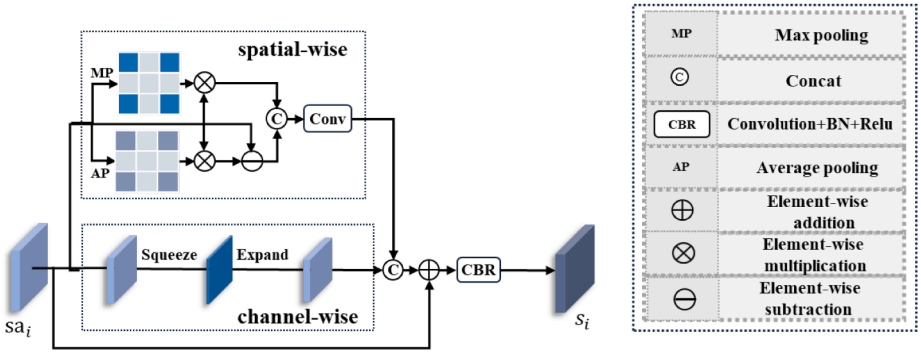


Fig. 3. Illustration of GSE.

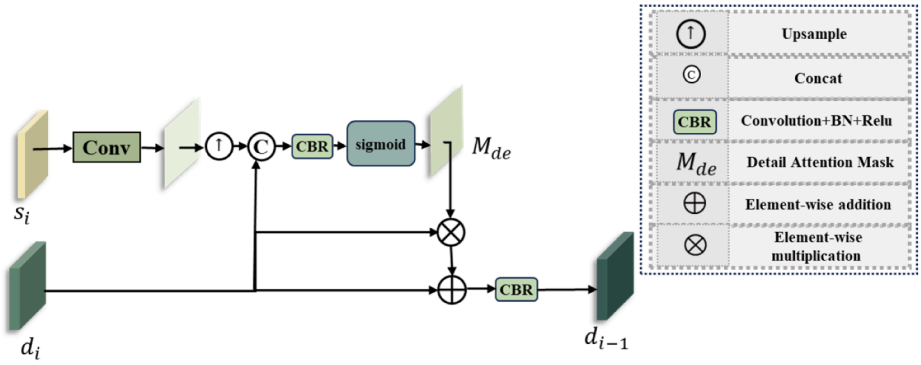


Fig. 4. Illustration of LocaL Detail Extraction (LDE).

3.2. Global semantic ement module

Rich context and semantic information of the target are crucial to segmentation results [14]. In this paper, GSE modules are proposed to extract and enhance the low-frequency semantics of the central region to get the accurate location of the targets, as shown in Fig. 3. Specifically, the output feature of the upper-scale s_{i+1} and the current-scale feature a_i are concatenated as input $\{sa_i\}_{i=1}^{3}$. Specially, for the first GSE, f_4 is selected as s_4. Then sa_i passes through a residual block with two branches: spatial-wise branch and channel-wise branch.

For COD, the network is expected to have a greater response to the features of the target region. So in the spatial-wise branch, a maximum pooling layer is used to locate the target. To obtain more discriminative features, an average pooling layer is used to obtain similar patterns between the target and background and the difference between the input feature sa_i and the average-pooled feature is computed to remove the similar patterns from sa_i and reserve more discriminative features. Through the spatial-wise branch, the target region gets a larger response and becomes prominent in the feature maps. This process can be described as:

$$sa_i^{sp} = \mathcal{F}(C(\sigma(MP(sa_i)) \cdot sa_i, (1 - \sigma(AP(sa_i))) \cdot sa_i)), \tag{1}$$

where σ represents sigmoid. $\mathcal{F}$ denotes a 3×3 convolution layer together with Batch Normalization and Relu while C means concatenation. MP and AP represent max pooling and average pooling.

For the channel-wise branch, we use pointwise convolutions with a channel compression ratio of 0.25 and a channel expansion ratio of 4, which can aggregate information along the channels of features and enrich local features with very little computational overhead.

The above two branches enhance the features of the target's central region from both spatial and channel aspects. Then the global semantic enhanced features s_i are obtained as follows:

$$s_i = \mathcal{F}(C(\mathcal{PW}_{r=4}(\mathcal{PW}_{r=0.25}(sa_i)), sa_i^{sp}) + sa_i), \tag{2}$$

where $\mathcal{PW}_r$ represents the pointwise convolution with a compression or expansion ratio r. After concatenation, an 1×1 convolution is applied to make sure the concatenated feature has the same size as sa_i.

3.3. Local detail extraction

To compensate for the loss of features and minimize background interference near the target, we design a high-resolution peripheral region branch. The peripheral region is the neighborhood of the target boundary. The peripheral region branch enhances the focus on this region to fully explore the features, eliminate background interference near the target, and strengthen the contrast between the target and the background.

To locate the target and recover texture high-frequency details from I_{in}, f_4 is used as guidance to indicate the approximate location of the target and a_1 is employed to offer more details. The features f_4 and a_1 are upsampled to the same resolution with the input image I_{in} and concatenated with I_{in} as input d_4, which is convolved to obtain high-resolution detail feature d_3 of the target region from I_{in}. After that, the detail features are refined with a stack of Local Detail Extraction (LDE) modules under the guidance of the semantic features s_i.

As Fig. 4 shows, our LDE module takes the semantic feature s_i and the corresponding detail feature d_i as inputs. First, s_i is convolved and upsampled and then concatenated with the detail feature d_i. After that, a detail attention mask M_{de} is generated to make the detail features pay more attention to the peripheral region. The calculation process of M_{de} can be expressed as:

$$M_{de} = \sigma(\mathcal{F}(C(\mathcal{U}(\mathcal{F}(s_i)), d_i))), \tag{3}$$

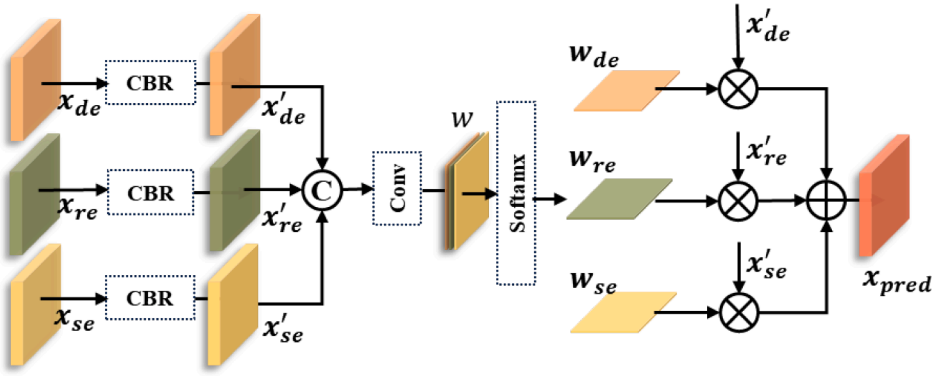


Fig. 5. Illustration of Region Adaptive Fusion (RAF).

where $\mathcal{U}$ means upsample while $\mathcal{F}$ denotes Convolutional layers, Batch normalization, and Relu (abbreviated as CBR in Fig. 4). Then the local detail refined feature d_{i-1} can be computed as:

$$d_{i-1} = \mathcal{F}(d_i + M_{de} \cdot d_i). \tag{4}$$

The details of the peripheral regions can be supplemented during the continuous iteration process. Since we use the outputs of three GSE modules $\{s_i\}_{i=1}^{3}$ as the guidance information for three LDE modules, the interference of non-target regions can also be reduced.

3.4. Region adaptive fusion

To fully utilize and integrate the features from the central region branch and the peripheral region branch, we design a RAF module, as shown in Fig. 5.

The three inputs of the module are x_{de}, x_{re} and x_{se}, where x_{de} is the high-resolution texture feature d_0 from the peripheral region branch, x_{re} is obtained by progressively upsampling s_1 with upsample layer and residual blocks, and x_{se} is obtained by directly upsampling the feature s_1 from the central region branch to the same resolution as x_{de}, as shown in Fig. 2.

The three features are dimensionally reduced through Convolutional layers, Batch normalization, and Relu (CBR) to get x'_{de}, x'_{re}, and x'_{se}. Then they are concatenated and convolved with 1×1 convolution to have the feature w with three channels. These three channels are the mapping of x'_{de}, x'_{re}, and x'_{se} respectively, and integrate the information from three input features x_{de}, x_{re}, and x_{se}. We apply softmax on these three mapping channels and split them along the channel dimension to obtain the weights w_{de}, w_{re}, and w_{se}. They perform a weighted fusion of features x'_{de}, x'_{re}, and x'_{se} to generate the final feature x_{pred}, expressed as:

$$x_{pred} = w_{de}x_{de'} + w_{re}x_{re'} + w_{se}x_{se'}. \tag{5}$$

The feature x_{pred} is then convolved through 1×1 convolution to obtain the final output prediction P_{seg-f}. This RAF module ensures the adaptive fusion of semantic features and detail features, making the features complementary to each other and ensuring the consistency of the features.

3.5. Loss function

Our supervision strategy is divided into three parts. The first part $\mathcal{L}_{seg-s}$ is the supervision for the global semantic prediction P_{seg-s} of the central region branch. The second part $\mathcal{L}_{de}$ is the supervision for the peripheral region prediction P_{er}, and the third part $\mathcal{L}_{seg-f}$ is the constraint on the final prediction P_{seg-f}.

For the supervision of the central region $\mathcal{L}_{seg-s}$, we adopt weighted binary cross-entropy loss and weighted IoU loss. It is set as:

$$\mathcal{L}_{seg-s} = \mathcal{L}^{w}_{BCE}(P_{seg-s}, GT) + \mathcal{L}^{w}_{IoU}(P_{seg-s}, GT). \tag{6}$$

Note that since P_{seg-s} is generated at 1/16 of the input, the ground truth is correspondingly downsampled to match its spatial dimensions for loss computation.

For the supervision of the peripheral region, we propose a peripheral region loss, which calculates the L1 distance between the predicted image and the ground truth in the peripheral region. The peripheral region is defined as the band surrounding the edge of the camouflaged object, as illustrated in Fig. 6. To obtain the peripheral region, morphological operations are applied to the ground truth. Specifically, a 30×30 kernel is used for both dilation and erosion, each applied once. The difference between the dilated and eroded masks generates a narrow band, i.e. the peripheral region around the object boundary, corresponding to an approximately 60-pixel-wide region. The peripheral region PR can be calculated as follows:

$$PR = \text{Dilate}(GT, 30 * 30, 1) - \text{Erode}(GT, 30 * 30, 1), \tag{7}$$

where GT denotes ground truth. Dilate(GT,30*30,1) represents a dilation operation using a 30×30 kernel with one iteration. Erode(GT,30*30,1) represents an erosion operation using a 30×30 kernel with one iteration. In contrast, the central region is defined as the other area in the image.

Based on the peripheral region PR, the peripheral region supervision mask M_{er} is then calculated as:

$$M_{er} = GT \odot PR, \tag{8}$$

where $\odot$ denotes element-wise multiplication within the peripheral region. Note that the peripheral region supervision mask M_{er} includes only the pixels within the peripheral region, which are marked as black or white, as illustrated in Fig. 6. Pixels outside the peripheral region, which do not require constraints, are marked as gray.

This mask is then used to guide the weighted peripheral region loss, enforcing stronger supervision on the peripheral region while reducing background interference.

Considering the prior that the size of the camouflaged object varies, a weighted peripheral region loss $\mathcal{L}^{w}_{er}$ is designed to alleviate the problem:

$$\mathcal{L}_{de} = \mathcal{L}^{w}_{er}(P_{er}, M_{er}) = \frac{\sum_{i,j}^{H,W} \left| P_{ij} \cdot M_{ij} - M_{ij} \right|}{\sum_{i,j}^{H,W} M_{ij}/HW}, \tag{9}$$

where the inverse proportion of the peripheral region in the image M_{er} is utilized as the weight. It performs different constraint strengths for targets of different sizes. By constraining the pixels of the boundary vicinity, we suppress background interference near the target while excavating the peripheral region details. Fig. 6 shows the calculation process of $\mathcal{L}_{de}$ and we mark the regions that do not require constraints in gray.

For the final prediction $\mathcal{L}_{seg-f}$, we employ weighted BCE loss and weighted IoU loss, the same as $\mathcal{L}_{seg-s}$. The overall loss $\mathcal{L}_{seg}$ can be expressed as:

$$\mathcal{L}_{seg} = \gamma\mathcal{L}_{seg-s} + \beta\mathcal{L}_{de} + \mathcal{L}_{seg-f}. \tag{10}$$

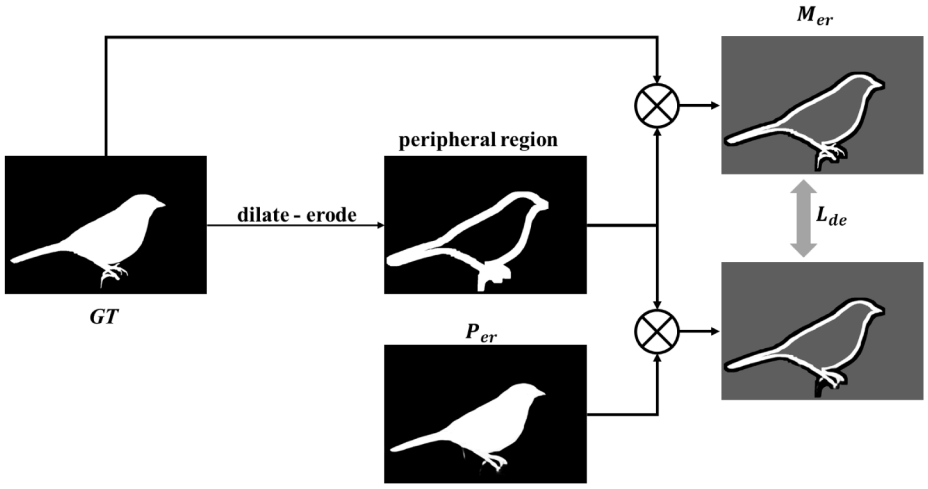


Fig. 6. Details of M_{er} and $\mathcal{L}_{de}$.

Since the resolution of the semantic feature is 1/16 of the final prediction, we set γ to 1/16 to impose slight constraints on it.To balance the constraint strengths of the peripheral region loss and other losses, we set β to 3.

4. Experiments

4.1. Experiment setup

DataSet. We evaluate our method on four benchmark datasets: CAMO [20], COD10K [1], NC4K[9] and CHAMELLON[52]. CAMO contains 1250 camouflaged images (1,000 for training and 250 for testing). COD10K includes 5066 camouflaged images (3,040 for training and 2026 for testing) covering 5 super-classes and 69 subclasses. NC4K includes 4121 images downloaded from the Internet. CHAMELLON contains 76 camouflaged images which are collected from the Internet by searching "camouflaged animal" as the keyword. We follow previous works to use the training set of CAMO and COD10K for training (4,040 images) and others for testing.

Evaluation Metrics. Following [1], four standard metrics are used to comprehensively evaluate the model performance: Structure measure (S_α) [30], weighted F-measure (F_β^ω) [23], Mean Absolute Error (MAE), and mean E-measure (E_ϕ^{mn}) [24]. Structure measure (S_α) is adopted to evaluate region-aware and object-aware structural similarity. Weighted F-measure (F_β^ω) is a comprehensively reliable measure of both weighted precision and weighted recall. Mean Absolute Error (MAE) evaluates the element-wise difference between the normalized prediction and the ground truth. Mean E-measure (E_ϕ^{mn}) focuses on evaluating the overall and local accuracy of camouflaged object detection.

Implementation details. We implement our model on PyTorch. PVT-v2 [26], pre-trained on ImageNet, is utilized as the backbone. The input image is resized to 384 × 384. To enhance model generalization, we apply several data augmentation techniques, including random horizontal and vertical flipping, random rotations within ±10 degrees, color jittering (adjusting brightness, contrast, saturation, and hue), and random cropping.Adam optimizer is set. The learning rate is set to 4e-5 and decays follow a step curve. During training, the batch size is set to 8 and the maximum epoch is set to 100. All the experiments are running with an Nvidia GeForce RTX 3090 GPU.

4.2. Comparisons with state-of-the-arts

4.2.1. Qualitative evaluation

Our results across various scenarios are compared with previous SOTA methods in Fig. 7. In the first three rows, for targets with structures and colors similar to the environment, our method perfectly discriminates the target and its details (highlighted by red rectangles in GT) from the environment. In row 4, for occluded targets, HitNet segments the target completely but loses the occlusion details marked by a red rectangle in GT. By contrast, our PONet recognizes the occluded region while ensuring the integrity of the target. In row 5, for large targets, PONet captures more target semantic information and preserves superior boundary details compared to other methods. In row 6, for small targets, our segmentation results demonstrate more accurate target positioning, and the details are also richer, such as the antennae of the insect. In the last row, during multi-target segmentation, other methods can segment the target with a lower degree of camouflage, but the more camouflaged one cannot be recognized completely, such as the ears and the legs of the right deer. Our method can accurately segment both targets with fine-grained details. Overall, our method exhibits excellent performance in detecting camouflaged targets and capturing fine-grained details.

4.2.2. Quantitative evaluation

Table 1 lists the quantitative comparisons between our PONet and 31 SOTA COD methods across four benchmark datasets. The results demonstrate that PONet achieves highly competitive performance. While several competing models surpass PONet in specific metrics, they often come with substantially higher computational costs. For instance, COMPrompter [41] requires 94.86M parameters and 369.01G FLOPs, whereas PONet only needs 62.68M parameters and 34.93G FLOPs. Similarly, both FSEL [44] and PNet [45] exhibit relatively high computational complexity. Moreover, PNet was trained with an additional 6665 images with box annotations, further elevating its resource consumption, while FSEL was trained with larger input images, which also increases its computational demands. Likewise, HitNet [18] delivers strong performance on the CHAMELLON dataset (76 test images) yet shows relatively weaker performance on the other three datasets (each with more test images). Notably, this mixed performance still comes at the cost of significantly higher FLOPs, coupled with a larger 704 × 704 input resolution requirement (which further increases computational burden). The above experimental comparisons demonstrate that PONet not only maintains superior performance but also significantly reduces computational overhead, making it a more efficient and practical solution.

To present the evaluation metrics more intuitively, we also plot Precision-Recall and Fmeasure-Threshold curves to compare the testing performances of some SOTA methods and our method across 3 benchmarks, as illustrated in Fig. 8. The closer the Precision-Recall curve is to the upper-right corner, the better the performance is. The

Table 1
Quantitative results of our method and other 31 state-of-the-art COD methods on four benchmark datasets. "↑(↓)" means that higher (lower) values indicate better performance. The best results are in bold, the second-ranked results are emphasized with green color and the third-ranked results are emphasized with blue color.

Model	Backbone	Params	Flops	CAMO(250)				COD10K(2026)				NC4K(4121)				CHAMELLON(76)			
				$S_\alpha\uparrow$	$F_\beta^w\uparrow$	$MAE\downarrow$	$E_\beta^m\uparrow$	$S_\alpha\uparrow$	$F_\beta^w\uparrow$	$MAE\downarrow$	$E_\beta^m\uparrow$	$S_\alpha\uparrow$	$F_\beta^w\uparrow$	$MAE\downarrow$	$E_\beta^m\uparrow$	$S_\alpha\uparrow$	$F_\beta^w\uparrow$	$MAE\downarrow$	$E_\beta^m\uparrow$
SINet [1]($CVPR_{20}$)	ResNet-50	48.946M	19.475G	0.751	0.606	0.100	0.771	0.771	0.551	0.051	0.806	0.808	0.723	0.058	0.872	0.869	0.740	0.044	0.891
PFNet [2]($CVPR_{21}$)	ResNet-50	18.977M	46.498G	0.782	0.695	0.085	0.852	0.800	0.660	0.036	0.890	0.829	0.745	0.053	0.898	0.882	0.810	0.033	0.942
MGL [16]($CVPR_{21}$)	ResNet-50	73.73M	378.2G	0.775	0.673	0.088	0.847	0.814	0.666	0.035	0.865	0.833	0.739	0.053	0.893	0.893	0.813	0.030	0.923
LSR [9]($CVPR_{21}$)	ResNet-50	28.720M	17.423G	0.787	0.696	0.080	0.838	0.804	0.673	0.037	0.880	0.840	0.766	0.048	0.895	0.842	0.794	0.046	0.896
UJSC [8]($CVPR_{21}$)	ResNet-50	–	–	0.800	0.728	0.073	0.859	0.809	0.684	0.035	0.884	0.842	0.771	0.047	0.898	0.894	0.848	0.030	0.943
UGCT [15]($ICCV_{21}$)	ResNet-50	–	–	0.785	0.686	0.086	0.823	0.818	0.667	0.035	0.853	0.839	0.747	0.052	0.874	0.888	0.796	0.031	0.918
C2FNet [14]($IJCAI_{21}$)	ResNet-50	18.063M	25.214G	0.796	0.719	0.080	0.854	0.813	0.686	0.036	0.890	0.838	0.762	0.049	0.897	0.893	0.845	0.028	0.947
SINet-v2 [21] ($TPAMI_{22}$)	ResNet-50	–	–	0.820	0.743	0.070	0.882	0.815	0.680	0.037	0.887	0.847	0.770	0.048	0.903	0.888	0.816	0.030	0.942
BSANet [6]($AAAI_{22}$)	Res2Net-50	28.79M	39.64G	0.794	0.717	0.079	0.851	0.818	0.699	0.034	0.891	0.841	0.771	0.048	0.897	0.895	0.841	0.027	0.946
OCENet [22]($WACV_{22}$)	ResNet-50	–	–	0.802	0.723	0.080	0.852	0.827	0.707	0.033	0.894	0.853	0.785	0.045	0.902	0.901	0.843	0.028	0.940
ERRNet [5](PR_{22})	ResNet-50	67.798M	40.050G	0.779	0.679	0.085	0.842	0.786	0.630	0.043	0.867	0.827	0.737	0.054	0.887	0.877	0.805	0.036	0.927
BGNet [3]($IJCAI_{22}$)	Res2Net-50	–	–	0.812	0.749	0.073	0.870	0.831	0.722	0.033	0.901	0.851	0.788	0.044	0.907	–	–	–	–
SegMaR [10]($CVPR_{22}$)	ResNet-50	–	–	0.815	0.753	0.071	0.874	0.833	0.724	0.034	0.899	0.841	0.781	0.046	0.896	0.906	0.860	0.025	0.954
ZoomNet [11]($CVPR_{22}$)	ResNet-50	32.382M	203.496G	0.820	0.752	0.063	0.895	0.838	0.729	0.029	0.888	0.853	0.784	0.043	0.896	0.902	0.845	0.023	0.958
DGNet [4] (MIR_{23})	EfficientNet	21.02M	2.77G	0.839	0.769	0.057	0.901	0.822	0.693	0.033	0.896	0.857	0.784	0.042	0.911	–	–	–	–
CINet [12] ($NeuCom_{23}$)	ResNet-50	26.79M	20.93G	0.827	0.763	0.066	0.888	0.830	0.710	0.033	0.904	0.855	0.789	0.043	0.910	0.905	0.843	0.028	0.947
Bi-RRNet [13](PR_{23})	Van-samll	14.95M	12.65G	0.843	0.802	0.054	0.909	0.840	0.746	0.026	0.912	0.861	0.810	0.038	0.917	0.896	0.852	0.022	0.960
DaCOD [31]($ACMMM_{23}$)	ResNet-50	–	–	0.855	0.796	0.051	–	0.840	0.729	0.028	–	0.874	0.814	0.035	–	0.898	0.856	0.029	0.954
FEDER [17]($CVPR_{23}$)	ResNet-50	–	–	0.827	0.801	0.064	0.893	0.837	0.756	0.028	0.913	0.859	0.832	0.041	0.915	0.894	0.855	0.028	0.947
EVP [19]($CVPR_{23}$)	SegFormer-B4	3.20M	–	0.846	0.777	0.059	0.895	0.843	0.742	0.029	0.907	–	–	–	–	0.871	0.795	0.036	0.917
FSPNet [25]($CVPR_{23}$)	ViT	–	–	0.856	0.799	0.050	0.899	0.851	0.735	0.026	0.895	0.879	0.816	0.035	0.915	-	-	-	–
HitNet [18]($AAAI_{23}$)	PVT-v2	–	54G	0.844	0.801	0.057	0.902	0.868	0.798	0.024	0.932	0.870	0.825	0.039	0.921	**0.922**	**0.903**	**0.018**	**0.970**
CamoFocus-E1 [33]($WACV_{24}$)	PVT-v2	–	–	0.830	0.770	0.062	0.893	0.830	0.719	0.030	0.899	0.855	0.790	0.042	0.912	0.901	0.846	0.024	0.940
ICEG [34]($ICLR_{24}$)	ResNet-50	–	–	0.810	0.789	0.068	–	0.826	0.747	0.030	–	0.849	0.814	0.044	–	0.899	0.858	0.027	0.950
GenSAM [32]($AAAI_{24}$)	SAM	–	–	0.719	0.659	0.113	0.775	0.775	0.681	0.067	0.838	–	–	–	–	0.764	0.780	0.090	0.807
DSAM [43]($ACMMM_{24}$)	SAM	121.93M	380.14G	0.832	0.794	0.061	0.913	0.846	0.760	0.033	0.921	0.871	0.826	0.040	0.932	–	–	–	–
.RISNet [46]($CVPR_{24}$)	PVT-v2	26.6M	248.5G	0.870	0.827	0.050	0.922	0.873	0.799	0.025	0.931	0.882	0.834	0.037	0.925	–	–	–	–
PRNet [42]($TCSVT_{24}$)	PVT-v2	24.21M	14.05G	0.860	0.848	0.050	0.915	0.859	0.793	0.025	0.925	0.879	0.858	0.035	0.929	0.909	0.876	0.024	0.957
FSEL [44]($ECCV_{24}$)	PVT-v2	67.13M	54.73G	**0.885**	0.851	0.040	**0.942**	0.873	0.800	0.021	0.928	0.892	0.853	0.030	0.941	–	–	–	–
PNet [45]($ECCV_{24}$)	PVT-v2	61.00M	45.00G	0.882	**0.870**	**0.039**	**0.942**	**0.901**	**0.857**	**0.016**	**0.960**	0.906	**0.888**	**0.024**	**0.958**	0.908	0.886	0.021	0.964
COMPrompter [41](($SCIS_{25}$))	SAM	94.86M	369.01G	0.882	0.858	0.044	**0.942**	0.889	0.821	0.023	0.949	**0.907**	0.876	0.030	0.955	0.906	0.857	0.026	0.955
PONet(Ours)	PVT-v2	62.68M	34.93G	0.874	0.833	0.045	0.929	0.874	0.792	0.022	0.934	0.892	0.848	0.030	0.938	0.911	0.858	0.023	0.954

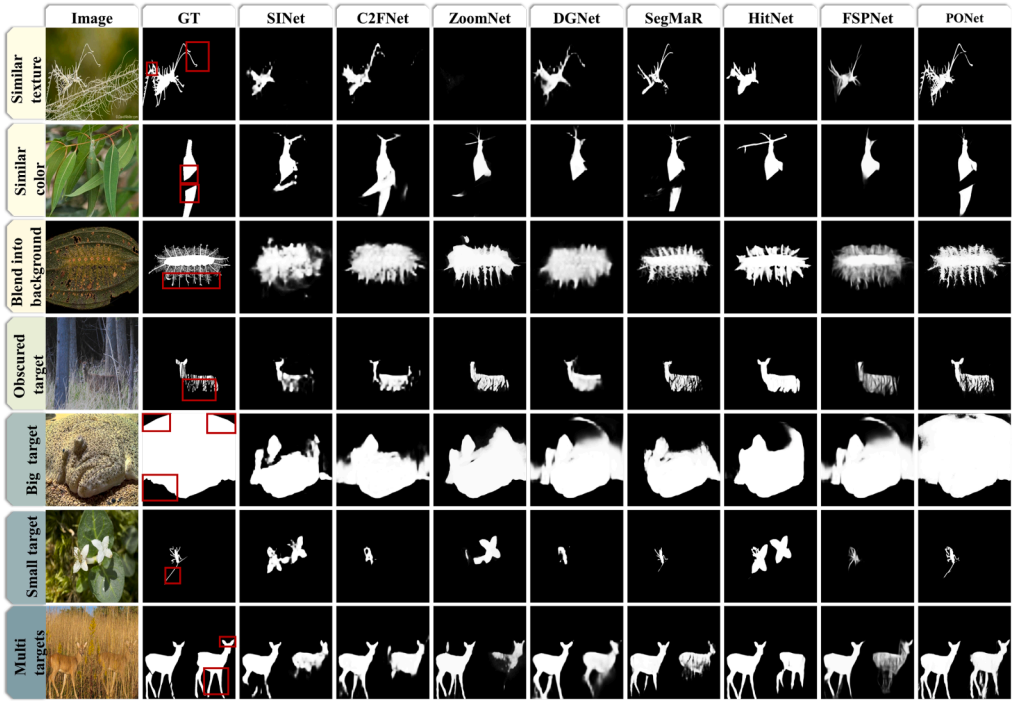


Fig. 7. Visual comparison of the proposed model with eight state-of-the-art COD methods. Our method is capable of accurately segmenting various camouflaged objects in difficult senses.

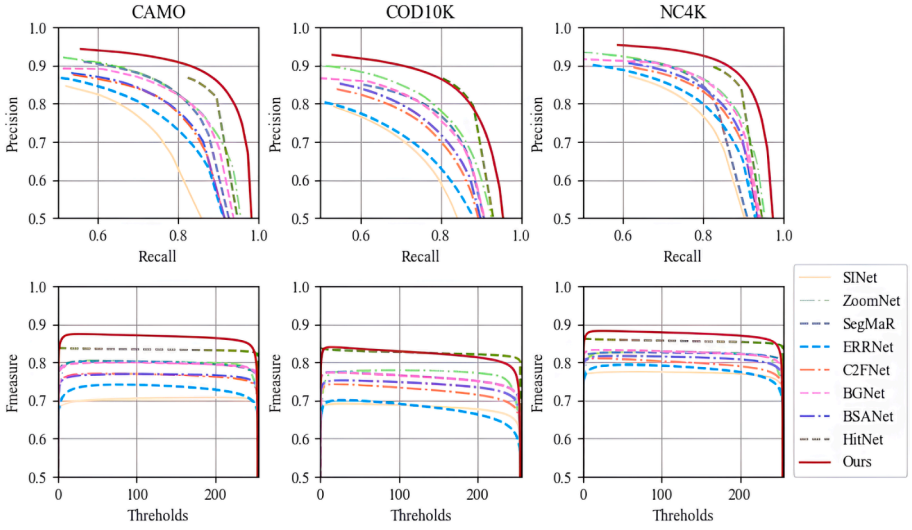


Fig. 8. Precision-recall and Fmeasure-threshold curves to compare the testing performances of some SOTA methods and our method under varying thresholds across 3 benchmarks.

higher the F-measure curve is, the better the model works. These curves provide a comprehensive perspective on model behaviors under varying threshold configurations. As shown in the figure, our method consistently yields superior curves, highlighting its robustness and accuracy.

Comparison with SOD Methods. To further validate the effectiveness of our method, we applied several SOD models, such as EGNet, BASnet, S2MA, and VST++ to the COD dataset. Experimental results in Table 2 show that these models, originally designed for SOD tasks, perform worse in camouflaged object detection scenarios compared to our model specifically designed for COD. These results highlight the differences between COD and SOD and demonstrate the superiority of our method in addressing the unique challenges of COD.

4.3. Ablation studies

4.3.1. Ablation of different modules

We conduct ablation studies to demonstrate the effect of different modules, including the GSE modules, the LDE modules, the proposed weighted Peripheral Region Loss (PRL), and the RAF module. The results are shown in Table 3. The baseline A only uses the backbone network and several convolutional layers for prediction. We install the components one by one on the baseline model to evaluate their performance. It can be seen from the table that each component improves the results of the model, which shows that our components are all effective.

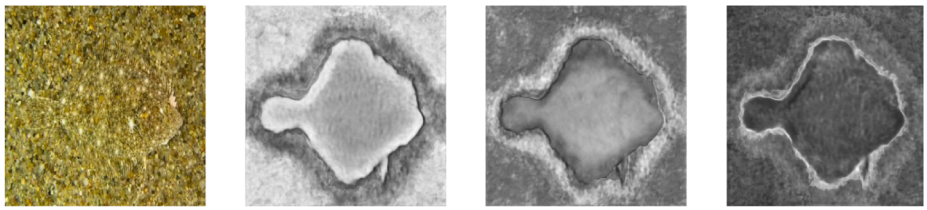
Effect of GSE. In the proposed model, the GSEs are used to enrich semantic information of the features. In Table 3, Model B adds GSEs based

Table 2
Quantitative results of our method and 4 SOD methods on three benchmark datasets. "↑(↓)" means that higher (lower) values indicate better performance. The best results are in bold.

Model	Pub.year	CAMO(250)				COD10K(2026)				NC4K(4121)			
		$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$MAE \downarrow$	$E_\beta^m \uparrow$	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$MAE \downarrow$	$E_\beta^m \uparrow$	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$MAE \downarrow$	$E_\beta^m \uparrow$
EGNet [37]	$ICCV_{19}$	0.601	0.446	0.149	0.514	0.613	0.424	0.086	0.524	0.707	0.533	0.095	0.614
BASNet [38]	$CVPR_{19}$	0.580	0.438	0.152	0.450	0.587	0.401	0.098	0.445	0.667	0.576	0.103	0.457
S2MA [40]	$TPAMI_{21}$	0.606	0.460	0.135	0.554	0.653	0.487	0.076	0.593	0.720	0.574	0.078	0.685
VST++ [39]	$TPAMI_{24}$	0.618	0.480	0.126	0.634	0.677	0.516	0.063	0.661	0.738	0.581	0.068	0.77
PONet(Ours)	–	**0.874**	**0.833**	**0.045**	**0.929**	**0.874**	**0.792**	**0.022**	**0.934**	**0.892**	**0.848**	**0.030**	**0.938**

Table 3
Ablation studies of different modules on three large-scale test datasets. "↑(↓)" means that higher (lower) values indicate better performance. GSE means global semantic enhancement module. LDE stands for local detail extraction module. PRL stands for weighted peripheral region loss. RAF is the abbreviation for region adaptive fusion.

model	Params	FLOPs	CAMO		COD10K		NC4K	
			$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$S_\alpha \uparrow$	$F_\beta^w \uparrow$
A. baseline	62.04M	12.37G	0.854	0.769	0.821	0.656	0.860	0.761
B. A + GSE	62.26M	15.70G	0.870	0.821	0.864	0.760	0.887	0.831
C. A + LDE	62.06M	20.13G	0.871	0.828	0.867	0.780	0.889	0.841
D. A + GSE + LDE	62.28M	23.46G	0.872	0.827	0.870	0.776	0.890	0.840
E. A + GSE + LDE + PRL	62.64M	28.68G	0.874	0.830	0.872	0.787	0.891	0.845
F. A + GSE + LDE + PRL + RAF	62.68M	34.93G	**0.874**	**0.833**	**0.874**	**0.792**	**0.892**	**0.848**



(a) Original image (b) w_{se} (c) w_{re} (d) w_{de}

Fig. 9. Heatmaps of the learned weight maps for the adaptive weighting mechanism of RAF. The first image depicts a camouflaged fish in the sea. From left to right, the subsequent images show w_{se}, w_{re}, and w_{de}, which are weight maps corresponding to the directly upsampled semantic feature x_{se}, the iteratively refined semantic feature x_{re}, and the high-resolution detail feature x_{de}, respectively.

on the baseline and the performance has been greatly improved. This indicates that the GSE module we designed is very effective in enhancing the features of COs, while only introducing a marginal increase in parameters and GFLOPs. This demonstrates that the semantic enhancement is achieved at a very low computational cost.

Effect of LDE. The LDE modules are designed to mine and supplement details information in the features. The LDE modules are added to Model C in Table 3, which can help to supplement the detail features with semantic features directly. Notably, this performance improvement is achieved with minimal parameter growth relative to the baseline, further validating the efficiency of our module design.

Effect of PRL and RAF. Model E incorporates the proposed weighted peripheral region loss (PRL), which effectively constrains the prediction of COs. Model F fuses the detail features and semantic features via RAF, yielding the most optimal performance. Notably, the introduction of PRL and RAF only induces a slight increase in the number of parameters, indicating that our final design gains significant performance improvement with only a modest parameter increase.

4.3.2. The adaptive weighting of RAF

To gain deeper insights into the adaptive weighting mechanism of the RAF module, we visualize the heatmaps of the learned weight maps for the three input branches in Eq. (5): the directly upsampled semantic feature x_{se}, the iteratively refined semantic feature x_{re}, and the high-resolution detail feature x_{de}. These weight maps, denoted as w_{se}, w_{re}, and w_{de}, reflect the dynamic contribution of each feature component to the final prediction, as illustrated in Fig. 9. Higher brightness corresponds to larger weight values.

The first image depicts a camouflaged fish in the sea, where the fish's texture closely resembles the background, presenting a challenging scenario. From left to right, the subsequent images show w_{se}, w_{re}, and w_{de}, respectively. w_{se} emphasizes the central region of the camouflaged object and the background, where low-frequency semantics information prevails. The strong activation in this region confirms that directly upsampled semantic features play a crucial role in modeling the coarse semantic prior of the target. Additionally, w_{re} highlights the central region of objects while more finely highlights the background area surrounding camouflaged objects. This behavior stems from w_{re} capturing global and mid-level structural information, which aids in preserving shape consistency in ambiguous cases. Conversely, w_{de} focuses predominantly on the object's boundaries and fine details, exhibiting high responses along the edges. This indicates that the RAF module acknowledges the importance of high-resolution textures from the peripheral region in localizing camouflaged boundaries.

4.3.3. Effect of loss functions

We have also conducted an additional ablation study to evaluate the individual contributions of each loss function used in our training framework, namely L_{seg-s}, L_{de} and L_{seg-f}, shown in Table 4. Specifically, the first, second, and third rows show the results of removing L_{seg-s}, L_{de} and L_{seg-f} alone from the overall method. As expected, the removal of L_{seg-f}, which directly supervises the final prediction, results in substantial performance degradation across all datasets, validating

Table 4
Ablation study on loss functions. We remove each loss term alone from the overall method and evaluate the corresponding performance across three benchmark datasets. "↑(↓)" means that higher (lower) values indicate better performance.

Setting	CAMO				COD10K				NC4K			
	$S_\alpha \uparrow$	$F_\beta \uparrow$	MAE ↓	$E_\phi \uparrow$	$S_\alpha \uparrow$	$F_\beta \uparrow$	MAE ↓	$E_\phi \uparrow$	$S_\alpha \uparrow$	$F_\beta \uparrow$	MAE ↓	$E_\phi \uparrow$
w/o L_{seg-s}	0.869	0.823	0.048	0.923	0.869	0.780	0.025	0.930	0.885	0.836	0.032	0.931
w/o L_{de}	0.870	0.825	0.047	0.925	0.870	0.782	0.024	0.929	0.888	0.839	0.032	0.933
w/o L_{seg-f}	0.443	0.289	0.369	0.472	0.351	0.135	0.452	0.347	0.419	0.243	0.394	0.433
PONet(Ours)	0.874	0.833	0.045	0.929	0.874	0.792	0.022	0.934	0.892	0.848	0.030	0.938

Table 5
Comprehensive comparisons with SOTA methods.

model	backbone	Input size	Paras	MACs	FPS	COD10K			
						$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$MAE \downarrow$	$E_\phi^{mn} \uparrow$
SINet	ResNet50	352^2	48.95M	19.46G	185	0.776	0.631	0.043	0.864
PONet-res	ResNet50	352^2	26.45M	15.27G	213	**0.793**	**0.649**	**0.039**	**0.867**
BSANet	Res2Net50	384^2	32.58M	29.75G	173	0.818	0.699	0.034	0.891
PONet-res2	Res2Net50	384^2	26.61M	19.40G	165	**0.830**	**0.707**	**0.033**	**0.892**
DGNet	EffiNet-B4	352^2	18.11M	1.28G	277	0.822	0.693	0.033	0.896
PONet-effi	EffiNet-B4	352^2	18.24M	5.90G	137	**0.836**	**0.709**	**0.032**	**0.899**
HitNet	PVT-v2-b2	704^2	25.73M	56.25G	53	0.868	0.798	0.024	0.932
PONet-pvt	PVT-v2-b2	704^2	25.56M	58.39G	45	**0.887**	**0.818**	**0.021**	**0.940**

Table 6
Evaluation on lightweight backbone.

Model	Backbone	CAMO(250)		COD10K(2026)		NC4K(4121)	
		$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$S_\alpha \uparrow$	$F_\beta^w \uparrow$
PONet(Ours)	**MobileNet-s**	**0.726**	**0.598**	**0.761**	**0.581**	**0.789**	**0.674**
DGNet[4]	**MobileNet-s**	**0.735**	**0.587**	**0.729**	**0.517**	**0.779**	**0.638**
PRNet[42]	**MobileNet-s**	**0.634**	**0.56**	**0.729**	**0.664**	**0.749**	**0.727**

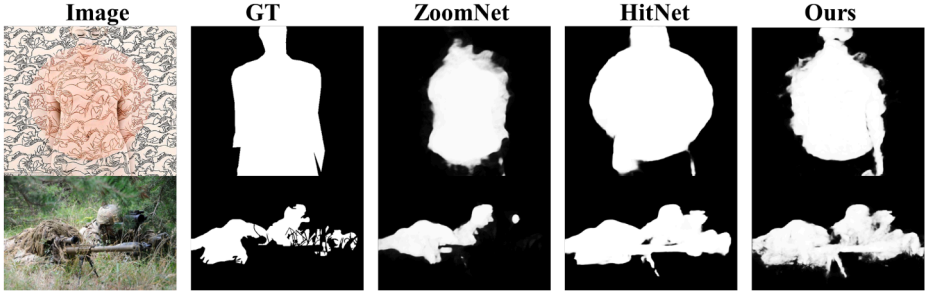


Fig. 10. Visualization of some failure cases.

its crucial role. The absence of L_{seg-s} supervision also causes a notable performance decline, as L_{seg-s} regulates the semantic guidance feature s_1. This feature is critical for guiding the LDE module and facilitating feature fusion within the RAF module. Comparing the third row with the last row reveals that L_{de} supervision further enhances performance. With explicit detail supervision, the method more accurately segments boundaries between camouflaged objects and the background.

4.3.4. Comprehensive evaluations

To explore factors that affect performance, the comprehensive evaluations on Parameters, MACs, FPS, and performance on COD10K are shown in Table 5. Our method switches the backbone and the input size based on the configurations of different SOTA methods. It reveals that the performance of our method outperforms SOTA methods with the configurations of any backbone and input size, which demonstrates the effectiveness of the proposed partitioned observing strategy, the innovative design of our modules, and the designed peripheral region loss.

We also evaluated different approaches by replacing the original backbone with MobileNet-s. Although the performance of our method with MobileNet-s lags behind PVT-v2, it still achieves competitive results when compared to other state-of-the-art methods which adopt the same lightweight backbone, as demonstrated in Table 6. These results indicate that leveraging lightweight architectures to balance performance and efficiency remains a critical challenge in this field, and we believe this direction warrants further exploration in future research.

4.4. Limitation of the proposed method

Despite the promising performance of PONet on different benchmark datasets, we recognize certain limitations in its ability to detect camouflaged targets that exhibit high visual similarity with their surroundings, particularly in terms of color and texture, as shown in Fig. 10. This limitation is particularly pronounced in scenarios involving artistic or military camouflage, where distinguishing the target from the background is inherently difficult. Notably, this problem remains unsolved by other SOTA methods. Further investigations are needed in the future to address this specific challenge.

5. Conclusion

In this paper, we introduce a novel Partitioned Observation Network (PONet) to achieve partitioned focus on the target's central region and peripheral region and explicitly model the multi-region synergy between these two regions for camouflaged object detection. We designed a global semantic enhancement (GSE) module to fully utilize the discriminative semantic features of the central regions by enhancing and refining the semantics. We also designed a local detail extraction (LDE) module, which continuously mines high-resolution features of peripheral regions under the guidance of semantic information. A weighted peripheral region loss is proposed to constrain the detailed features of the peripheral region. Finally, we use the region adaptive fusion (RAF) module to adaptively fuse the two parts of "*observation*". Experiments show that our method outperforms the previous 35 SOTA methods.

This approach demonstrates that effectively utilizing and modeling the cognitive characteristics of human vision contribute to the enhancement of camouflaged object detection performance. While our PONet achieves commendable results across different benchmark datasets, we recognize certain limitations in its ability to detect objects exhibiting extremely high visual similarity with their backgrounds, particularly in challenging scenarios such as artistically painted or militarily camouflaged settings. This challenge remains unresolved by other state-of-the-art methods as well. Future work will aim to enhance the detection performance of highly camouflaged targets, particularly in dynamic environments. This will involve refining our model's ability to handle a wider range of camouflage techniques, as well as exploring solutions for real-time detection and improved computational efficiency.

CRediT authorship contribution statement

Jinxia Zhang: Writing – review & editing, Writing – original draft, Methodology, Funding acquisition, Data curation, Conceptualization; **Yin Yuan:** Writing – original draft, Methodology; **Xuwen Zhu:** Writing – review & editing, Visualization, Validation; **Yang Hu:** Writing – review & editing, Visualization, Validation; **Kaihua Zhang:** Writing – review & editing, Visualization, Supervision.

Data availability

Data will be made available on request.

Declaration of interests

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.

Acknowledgement

This work was supported by National Natural Science Fund of China (Grant number 62573123), Research Fund for Advanced Ocean Institute of Southeast University, Nantong (GP202411), and the Fundamental Research Funds for the Central Universities. We also thank the Big Data Computing Center of Southeast University for providing the facility support on the numerical calculations in this paper.

Appendix A. Appendix A. Subclass Results on COD10K

To comprehensively and in detail compare different methods, we conducted quantitative evaluations on each subclass of the COD10K dataset, with results presented in Table A.1. Given the length of the table, it is included in this appendix. The results show that PONet achieves consistently superior performance across most subclasses, indicating that our dual-branch design with a low-frequency semantic branch and a high-frequency texture branch is effective in handling diverse camouflage scenarios.

Table A.1
Results of S_α for each subclass in COD10K. The best and second performing methods of each category are highlighted in bold and emphasized with green color, respectively.

Subclass	SINet	BGNet	DGNet	ZoomNet	EVP	FSPNet	Ours
Amphibian/Frog	0.785	0.861	0.843	0.876	0.874	0.852	**0.89**
Amphibian/Toad	0.85	0.887	0.857	0.885	0.887	**0.893**	0.883
Aquatic/BatFish	0.892	0.893	0.885	0.89	0.897	0.907	**0.915**
Aquatic/ClownFish	0.69	0.757	0.793	0.813	0.849	0.851	**0.855**
Aquatic/Crab	0.864	0.83	0.839	0.835	0.852	0.864	**0.891**
Aquatic/Crocodile	0.861	0.786	0.832	0.822	**0.874**	0.857	0.867
Aquatic/CrocodileFish	0.734	0.731	0.81	0.805	**0.907**	0.846	0.899
Aquatic/Fish	0.767	0.819	0.852	0.847	0.872	0.854	**0.882**
Aquatic/Flounder	0.784	0.881	0.899	0.88	0.907	0.922	**0.932**
Aquatic/FrogFish	0.738	0.856	0.905	0.925	0.887	0.925	**0.939**
Aquatic/GhostPipeFish	0.77	0.823	0.833	0.82	0.861	0.872	**0.892**
Aquatic/LeafySeaDragon	0.538	0.671	0.683	0.64	0.677	**0.782**	0.752
Aquatic/Octopus	0.741	0.901	0.897	0.86	0.908	0.885	**0.911**
Aquatic/Pagurian	0.604	0.691	0.707	0.701	0.69	0.71	**0.756**
Aquatic/PipeFish	0.728	0.811	0.821	0.791	0.817	0.828	**0.839**
Aquatic/ScorpionFish	0.74	0.808	0.85	0.843	0.862	0.851	**0.888**
Aquatic/SeaHorse	0.799	0.809	0.834	0.821	0.844	0.851	**0.875**
Aquatic/Shrimp	0.718	0.762	0.777	0.781	0.794	0.819	**0.843**
Aquatic/Slug	0.732	0.772	0.779	0.807	**0.835**	0.696	0.766
Aquatic/StarFish	0.799	0.869	0.902	0.912	0.916	0.889	**0.939**
Aquatic/Stingaree	0.784	0.757	0.836	0.851	0.884	0.901	**0.911**
Aquatic/Turtle	0.774	0.809	0.866	0.857	**0.891**	0.883	0.875
Flying/Bat	**0.907**	0.857	0.882	0.9	0.876	0.875	**0.907**
Flying/Bee	0.727	0.824	0.837	0.801	0.818	0.68	**0.853**
Flying/Beetle	0.917	0.941	0.932	0.918	0.902	0.932	**0.946**
Flying/Bird	0.807	0.863	0.875	0.882	0.874	0.873	**0.898**
Flying/Bittern	**0.905**	0.853	0.88	0.889	0.877	0.865	**0.905**
Flying/Butterfly	0.844	0.901	0.902	0.88	0.892	0.885	**0.926**
Flying/Cicada	0.872	0.895	0.904	0.908	0.906	0.909	**0.934**
Flying/Dragonfly	0.896	0.853	0.875	0.892	0.871	0.886	**0.901**
Flying/Frogmouth	**0.976**	0.948	0.962	0.963	0.954	0.947	0.963
Flying/Grasshopper	0.801	0.856	0.88	0.87	0.859	0.874	**0.891**
Flying/Heron	0.812	0.868	0.863	0.844	0.867	0.866	**0.897**
Flying/Katydid	0.711	0.842	0.851	0.82	0.842	0.853	**0.882**
Flying/Mantis	0.797	0.79	0.804	0.81	0.799	0.835	**0.849**
Flying/Mockingbird	0.811	0.895	0.89	0.879	0.886	0.875	**0.922**
Flying/Moth	0.863	0.932	0.945	0.928	0.925	0.941	**0.946**
Flying/Owl	0.832	0.883	0.892	0.869	0.892	0.875	**0.906**
Flying/Owlfly	0.772	0.836	0.854	0.842	0.879	0.872	**0.903**
Other/Other	0.635	0.807	0.889	0.887	0.889	0.899	**0.916**
Terrestrial/Ant	0.587	0.711	0.715	0.671	0.689	**0.743**	0.716
Terrestrial/Bug	0.82	0.87	0.885	0.848	0.875	0.874	**0.895**
Terrestrial/Cat	0.712	0.778	0.812	0.781	0.811	0.823	**0.846**
Terrestrial/Caterpillar	0.704	0.767	0.826	0.773	0.787	0.813	**0.854**
Terrestrial/Centipede	0.707	0.704	0.78	0.763	0.761	0.791	**0.793**
Terrestrial/Chameleon	0.759	0.824	0.861	0.827	0.839	0.845	**0.886**
Terrestrial/Cheetah	0.776	0.843	0.856	0.824	0.845	0.851	**0.865**
Terrestrial/Deer	0.8	0.777	0.815	0.808	0.801	0.798	**0.823**
Terrestrial/Dog	0.669	0.692	0.735	0.758	0.761	**0.786**	0.77
Terrestrial/Duck	**0.809**	0.751	0.789	0.781	0.787	0.784	**0.809**
Terrestrial/Gecko	0.825	0.857	0.902	0.896	0.894	0.908	**0.92**
Terrestrial/Giraffe	**0.897**	0.837	0.872	0.83	0.809	0.846	**0.897**
Terrestrial/Grouse	0.927	0.937	0.949	0.942	0.94	0.942	**0.958**
Terrestrial/Human	0.742	0.779	0.819	0.787	0.786	0.797	**0.847**
Terrestrial/Kangaroo	0.737	0.806	0.865	0.785	0.794	0.802	**0.88**
Terrestrial/Leopard	0.832	0.836	0.86	0.841	0.844	0.851	**0.882**
Terrestrial/Lion	0.82	0.816	0.839	0.846	0.848	**0.859**	0.839
Terrestrial/Lizard	0.8	0.851	0.868	0.846	0.852	0.853	**0.886**
Terrestrial/Monkey	0.804	0.897	0.902	0.897	0.883	**0.913**	0.911
Terrestrial/Rabbit	0.804	0.86	0.891	0.885	0.884	0.887	**0.913**
Terrestrial/Reccoon	0.738	0.845	0.859	0.809	0.833	0.791	**0.872**
Terrestrial/Sciuridae	0.806	0.883	0.911	0.892	0.897	0.856	**0.922**
Terrestrial/Sheep	0.49	0.481	0.531	0.573	**0.687**	0.493	0.493
Terrestrial/Snake	0.796	0.845	0.86	0.855	0.846	0.854	**0.873**
Terrestrial/Spider	0.736	0.794	0.816	0.792	0.799	0.808	**0.842**
Terrestrial/StickInsect	0.756	0.754	0.77	0.748	0.733	0.762	**0.792**
Terrestrial/Tiger	0.662	0.728	0.744	0.727	0.733	0.734	**0.755**
Terrestrial/Wolf	0.77	0.755	0.774	0.763	0.737	0.749	**0.831**
Terrestrial/Worm	0.672	0.8	0.819	0.815	0.779	0.812	**0.833**
Overall	0.771	0.831	0.822	0.838	0.843	0.851	**0.874**
Ranking	7	5	6	4	3	2	**1**

References

[1] D.-P. Fan, G.-P. Ji, G. Sun, M.-M. Cheng, J. Shen, L. Shao, Camouflaged object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 8772–8781.

[2] H. Mei, G.-P. Ji, Z. Wei, X. Yang, X. Wei, D.-P. Fan, Camouflaged object segmentation with distraction mining, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 8768–8777.

[3] L. Sun, S. Wang, C. Chen, T.-Z. Xiang, Boundary-guided camouflaged object detection, in: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), the International Joint Conference on Artificial Intelligence (IJCAI), 2022, pp. 1335–1341.

[4] G.-P. Ji, D.-P. Fan, Y.-C. Chou, D. Dai, A. Liniger, L.V. Gool, Deep gradient learning for efficient camouflaged object detection, Mach. Intell. Res. 20 (1) (2023) 92–108.

[5] G.-P. Ji, L. Zhu, M. Zhuge, K. Fu, Fast camouflaged object detection via edge-based reversible re-calibration network, Pattern Recognit. 123 (2022) 108414.

[6] H. Zhu, P. Li, H. Xie, X. Yan, D. Liang, D. Chen, M. Wei, J. Qin, I can find you! boundary-guided separated attention network for camouflaged object detection, Proc. AAAI Conf. Artif. Intell. (AAAI) 36 (2022) 3608–3616.

[7] M. Xiang, J. Zhang, Y. Lv, A. Li, Y. Zhong, Y. Dai, Exploring depth contribution for camouflaged object detection, 2021. arXiv preprint arXiv:2106.13217.

[8] A. Li, J. Zhang, Y. Lv, B. Liu, T. Zhang, Y. Dai, Uncertainty-aware joint salient object and camouflaged object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 10071–10081.

[9] Y. Lv, J. Zhang, Y. Dai, A. Li, B. Liu, N. Barnes, D.-P. Fan, Simultaneously localize, segment and rank the camouflaged objects, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 11591–11601.

[10] Q. Jia, S. Yao, Y. Liu, X. Fan, R. Liu, Z. Luo, Segment, magnify and reiterate: detecting camouflaged objects the hard way, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 4713–4722.

[11] Y. Pang, X. Zhao, T.-Z. Xiang, L. Zhang, H. Lu, Zoom in and out: a mixed-scale triplet network for camouflaged object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 2160–2170.

[12] X. Li, H. Li, H. Zhou, M. Yu, D. Chen, S. Li, J. Zhang, Camouflaged object detection with counterfactual intervention, Neurocomputing 553 (2023) 126530.

[13] Y. Liu, K. Zhang, Y. Zhao, H. Chen, Q. Liu, Bi-RRNet: bi-level recurrent refinement network for camouflaged object detection, Pattern Recognit. 139 (2023) 109514.

[14] G. Chen, S.-J. Liu, Y.-J. Sun, G.-P. Ji, Y.-F. Wu, T. Zhou, Camouflaged object detection via context-aware cross-level fusion, IEEE Trans. Circuits Syst. Video Technol. 32 (2022) 6981–6993.

[15] F. Yang, Q. Zhai, X. Li, R. Huang, A. Luo, H. Cheng, D.-P. Fan, Uncertainty-guided transformer reasoning for camouflaged object detection, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), the IEEE International Conference on Computer Vision (ICCV), 2021, pp. 4146–4155.

[16] Q. Zhai, X. Li, F. Yang, C. Chen, H. Cheng, D.-P. Fan, Mutual graph learning for camouflaged object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 12997–13007.

[17] C. He, K. Li, Y. Zhang, L. Tang, Y. Zhang, Z. Guo, X. Li, Camouflaged object detection with feature decomposition and edge reconstruction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 22046–22055.

[18] X. Hu, S. Wang, X. Qin, H. Dai, W. Ren, D. Luo, Y. Tai, L. Shao, High-resolution iterative feedback network for camouflaged object detection, Proceedings of the AAAI Conference on Artificial Intelligence (AAAI) 37 (2023) 881–889.

[19] W. Liu, X. Shen, C.-M. Pun, X. Cun, Explicit visual prompting for low-level structure segmentations, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 19434–19445.

[20] T.-N. Le, T.V. Nguyen, Z. Nie, M.-T. Tran, A. Sugimoto, Anabranch network for camouflaged object segmentation, Comput. Vision Image Understanding 184 (2019) 45–56.

[21] D.-P. Fan, G.-P. Ji, M.-M. Cheng, L. Shao, Concealed object detection, IEEE Trans. Pattern Anal. Mach. Intell. 44 (10) (2021) 6024–6042.

[22] J. Liu, J. Zhang, N. Barnes, Modeling aleatoric uncertainty for camouflaged object detection, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2022, pp. 1445–1454.

[23] R. Margolin, L. Zelnik-Manor, A, How to evaluate foreground maps?, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 248–255.

[24] D.-P. Fan, G.-P. Ji, X. Qin, M.-M. Cheng, Cognitive vision inspired object segmentation metric and loss function, Sci. Sin. Inf. 51 (6) (2021) 681–700.

[25] Z. Huang, H. Dai, T.-Z. Xiang, S. Wang, H.-X. Chen, J. Qin, H. Xiong, Feature shrinkage pyramid for camouflaged object detection with transformers, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 5557–5566.

[26] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, L. Shao, Pyramid vision transformer: a versatile backbone for dense prediction without convolutions, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), the IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 568–578.

[27] A. Lleras, S. Buetti, Z.J. Xu, Incorporating the properties of peripheral vision into theories of visual search, Nat. Rev. Psychol. 1 (10) (2022) 590–604.

[28] T.E. Boult, R.J. Micheals, X. Gao, M. Eckmann, Into the woods: visual surveillance of noncooperative and camouflaged targets in complex outdoor settings, Proc. IEEE 89 (10) (2001) 1382–1402.

[29] N.U. Bhajantri, P. Nagabhushan, Camouflage defect identification: a novel approach, in: Proceedings of the International Conference on Information Technology (ICIT), the International Conference on Information Technology (ICIT), 2006, pp. 145–148.

[30] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, A. Borji, Structure-measure: a new way to evaluate foreground maps, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 4548–4557.

[31] Q. Wang, J. Yang, X. Yu, F. Wang, P. Chen, F. Zheng, Depth-aided camouflaged object detection, in: Proceedings of the ACM International Conference on Multimedia (MM '23), the ACM International Conference on Multimedia (MM '23)New York, NY, USA, 2023, pp. 3297–3306.

[32] J. Hu, J. Lin, S. Gong, W. Cai, Relax image-specific prompt requirement in SAM: a single generic prompt for segmenting camouflaged objects, Proc. AAAI Conf. Artif. Intell. (AAAI) 38 (2024) 12511–12518.

[33] A. Khan, M. Khan, W. Gueaieb, A.E. Saddik, G.D. Masi, F. Karray, Camofocus: enhancing camouflage object detection with split-feature focal modulation and context refinement, in: 2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 2024, pp. 1423–1432.

[34] C. He, K. Li, Y. Zhang, Y. Zhang, C. You, Z. Guo, X. Li, M. Danelljan, F. Yu, Strategic preys make acute predators: enhancing camouflaged object detectors by generating camouflaged objects, in: Proceedings of the International Conference on Learning Representations (ICLR), the International Conference on Learning Representations (ICLR), 2024.

[35] H. Yang, Y. Zhu, K. Sun, H. Ding, X. Lin, Camouflaged object detection via dual-branch fusion and dual self-similarity constraints, Pattern Recognit. 157 (2024) 110895.

[36] G. Ji, L. Zhu, M. Zhuge, K. Fu, Fast camouflaged object detection via edge-based reversible re-calibration network, Pattern Recognit. 123 (2022) 108414.

[37] J. Zhao, J. Liu, D. Fan, Y. Cao, J. Yang, M. Cheng, EGNet: edge guidance network for salient object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[38] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, M. Jagersand, BASNet: boundary-aware salient object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 7471–7481.

[39] N. Liu, Z. Luo, N. Zhang, J. Han, VST : efficient and stronger visual saliency transformer, IEEE Trans. Pattern Anal. Mach. Intell. 46 (11) (2024) 7300–7316. https://doi.org/10.1109/TPAMI.2024.3388153

[40] N. Liu, N. Zhang, L. Shao, J. Han, Learning selective mutual attention and contrast for RGB-D saliency detection, IEEE Trans. Pattern Anal. Mach. Intell. 44 (12) (2022) 9026–9042. https://doi.org/10.1109/TPAMI.2021.3122139

[41] X. Zhang, Z. Yu, L. Zhao, COMPrompter: reconceptualized segment anything model with multiprompt network for camouflaged object detection, Sci. China Inf. Sci. 68 (1) (2025) 112104.

[42] X. Hu, X. Zhang, F. Wang, Efficient camouflaged object detection network based on global localization perception and local guidance refinement, IEEE Trans. Circuits Syst. Video Technol. 34 (7) (2024) 5452–5465.

[43] Z. Yu, X. Zhang, L. Zhao, Exploring deeper! segment anything model with depth perception for camouflaged object detection, in: Proceedings of the ACM International Conference on Multimedia, the ACM International Conference on Multimedia, 2024, pp. 4322–4330.

[44] Y. Sun, C. Xu, J. Yang, Frequency-spatial entanglement learning for camouflaged object detection, in: European Conference on Computer Vision(ECCV), Cham, Switzerland, Springer Nature, 2024, pp. 343–360.

[45] J. Zhang, R. Zhang, Y. Shi, Learning camouflaged object detection from noisy pseudo label, in: European Conference on Computer Vision(ECCV), Cham, Switzerland, Springer Nature, 2024, pp. 158–174.

[46] L. Wang, J. Yang, Y. Zhang, Depth-aware concealed crop detection in dense agricultural scenes, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR), 2024, pp. 17201–17211.

[47] P. Zhao, P. Xu, P. Qin, D.-P. Fan, Z. Zhang, G. Jia, B. Zhou, J. Yang, LAKE-RED: camouflaged images generation by latent background knowledge retrieval-augmented diffusion, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR), 2024, pp. 4092–4101.

[48] Z. Luo, N. Liu, W. Zhao, X. Yang, D. Zhang, D.-P. Fan, F. Khan, J. Han, VSCode: general visual salient and camouflaged object detection with 2D prompt learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR), the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR), 2024, pp. 17169–17180.

[49] B. Yin, X. Zhang, L. Liu, M.-M. Cheng, Y. Liu, Q. Hou, Camouflaged object detection with adaptive partition and background retrieval, Int. J. Comput. Vision(IJCV) 133 (7) (2025) 4877–4893. https://doi.org/10.1007/s11263-025-02406-6

[50] G. Yue, S. Wu, T. Zhou, G. Li, J. Du, Y. Luo, Q. Jiang, Progressive region-to-boundary exploration network for camouflaged object detection, IEEE Trans. Multimed. 27 (2025) 236–248. https://doi.org/10.1109/TMM.2024.3521761

[51] Y. Zhao, Q. Zhang, Y. Li, FLRNet: a bio-inspired three-stage network for camouflaged object detection via filtering, localization and refinement, Neurocomputing 626 (2025) 129523. https://doi.org/10.1016/j.neucom.2025.129523

[52] P. Skurowski, H. Abdulameer, J. Błaszczyk, T. Depta, A. Kornacki, P. Kozieł, Animal Camouflage Analysis: Chameleon Database, Unpubl. Manuscr. 2 (6) (2018) 7 .