Graph Guided Edge-Aware Learning for Camouflaged Object Detection via Swin Transformer

WANG Xiaoyi[1,2,3], *LI Mingyan*[3], *LI Bin*[3], *WU Fanlu*[3], *ZHANG Mingqiang*[3], *WANG Guan*[1,2], *ZHU Rui*[1,2], *CAI Hua*[1], *FU Qiang*[1,2*]

1. National and Local Joint Engineering Research Center of Space Optoelectronics Technology, Changchun University of Science and Technology, Changchun 130022, P. R. China; 2. College of Opto-electronic Engineering, Changchun University of Science and Technology, Changchun 130022, P. R. China; 3. Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, P. R. China

(Received 26 March 2026; revised 26 June 2026; accepted 9 July 2026)

Abstract: Compared with generic object detection, camouflaged object detection (COD) is more difficult due to its high similarity to the background. Most COD methods used convolutional neural network (CNN) in the past. In contrast, we note the great potential of Transformer in vision tasks, so we use Swin Transformer as the backbone to generate more robust multi-level features. Considering the complex characteristics between the camouflaged object and the background, we propose an edge-aware module (EAM) to generate a fine edge prior. In addition, the feature aggregation module (FAM) is used to fuse multi-level features, and the internal connections between edge points are investigated by feeding edge feature pixels into a graph convolutional network (GCN) to purify uncertain edge points, then guiding multi-level feature fusion operation. We reduce indistinguishable noise by cascading several FAMs and get prediction maps with sharp edges. Experimental results show that the proposed model can run at the real-time speed of 46 frames per second (FPS) on a single NVIDIA 2080Ti GPU. Comparing with 12 state-of-the-art COD methods on four public datasets, the proposed method outperforms other approaches.

Key words: camouflage object detection (COD); Swin Transformer; graph convolutional network (GCN); edge-aware learning

CLC number: TP391.41 **Document code:** A **Article ID:** 1005-1120(2026)04-0575-17

0 Introduction

Camouflage originally refers to a method that animals use to hide themselves or to deceive other animals, and the ability to camouflage usually affects the survival of these animals. Different from salient object detection (SOD), the goal of camouflaged object detection (COD) is to detect objects in environments when the objects are hidden in the background. Such objects blend in with the background using the strategies such as distraction[1], color matching[2], motion camouflage[3] and disruptive coloration[4]. Animals try to conceal themselves from predators to increase their chances of surviving. Inspired by this zoology mechanism, humans also apply camouflage to a variety of occasions. Soldiers conceal themselves by donning camouflage gear in challenging battlefield settings to reduce the risk of detection and attack. Simultaneously, people are also starting to focus on how to employ visual information to disrupt the camouflage.

Camouflage is common in a range of natural scenes and can deceive humans. Detecting a camouflaged object is more challenging than detecting tasks that require clear contrasts between the target and the background. The main reasons are as fol-

*Corresponding author, E-mail address: cust_fuqiang@163.com.

How to cite this article: WANG Xiaoyi, LI Mingyan, LI Bin, et al. Graph guided edge-aware learning for camouflaged object detection via Swin Transformer[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2026, 43(4): 575-591.

http://dx.doi.org/10.16356/j.1005-1120.2026.04.007

lows: (1) Confusing colors and textures; (2) blurred boundaries between edges and the background; (3) deception of human visual perception by objects[5]. While COD is difficult, it is still necessary. COD technology has great potential for application, such as defect detection (e.g., road crack detection), medical image segmentation (e.g., polyp segmentation, lung infection segmentation), and more downstream image analysis. Although several effective frameworks for general object detection have been proposed, they cannot achieve satisfactory performance for COD. Furthermore, early COD systems relied heavily on modules like receptive field blocks (RFBs)[6-8] to gather contextual information and detect possible regions. By widening the receptive field, these approaches gather adequate detailed information, but it is difficult to identify easily confused pixels. Some approaches[9-10] considered local information (e.g., boundaries), but they merely combine it with features and may cause misguidance.

Drawing on the team's preliminary research efforts[11-15], to address the above problems, we propose a novel network. It gradually generates refined prediction maps in a cascaded manner, supplements details via an edge-prior-based graph convolutional network (GCN), and shows outstanding performance on multiple datasets.

First of all, we use the Swin Transformer[16] as the backbone to generate more powerful feature representations and then connect the atrous spatial pyramid pooling (ASPP) module[17] at the final layer to generate the initial prediction prior which contains multi-scale contextual information. Second, we offer an edge-aware module (EAM) that aggregates the first three levels' features and generates accurate edge representations via residual connections with spatial attention and channel attention[18]. Then, we combine the current layer feature, higher-level feature with edge prior. Inspired by Ref.[19], we apply a GCN to refine the feature map's edge pixels with reliable edge prediction. After cascading multiple modules, we acquire prediction maps with finer boundary information. Note that we use all levels of features extracted from the backbone for more accurate results. We conduct comprehensive experiments on four benchmark datasets to validate the effectiveness of the proposed method, and experimental findings show that the model is effective. The main contributions are as follows:

(1) We design an EAM, which provides accurate global edge priors for subsequent feature aggregation operations.

(2) We develop a feature aggregation module (FAM) to combine different hierarchical features and utilize an edge point-based GCN to refine detailed information. Furthermore, we cascade three FAMs to progressively generate prediction maps of different scales.

(3) We further conduct comparative experiments on four commonly used datasets. Compared with 12 state-of-the-art approaches, the proposed method outperforms other competitors and can run at the real-time speed of 46 frames per second (FPS).

1 Related Work

1.1 Camouflaged object detection

Traditional methods mainly focus on low-level features of images (such as color, texture, gradient) to evaluate the differences between the objects and the background. Galun et al.[20] proposed a bottom-up aggregation framework that combines the structural features of textures with filter responses to differentiate objects through different statistical measures. Pan et al.[21] proposed a 3D convexity-based method to eliminate the influence of the background noise and identify the camouflaged target. Bhajantri et al.[22] employed gray-level co-occurrence matrix (GLCM)-based texture features for camouflage detection. Yin et al.[23] used optical flow to detect moving camouflaged targets. These methods only consider part of the image features and are useful in certain situations. It cannot, however, offer superior results in situations where the foreground and background are comparable.

Pioneers have achieved excellent achievements with convolutional neural networks (CNNs). Le et al.[24] built a new dataset of camouflaged images for benchmarking and presented an end-to-end network

that includes a segmentation branch and another classification branch. The classification branch is used to predict the probability that an image contains a camouflaged object, which is subsequently utilized to enhance segmentation performance in the segmentation branch. Fan et al.[6] proposed the SINet model which contains two modules: The search module (SM) locates the camouflaged objects, and the identification module (IM) is used to accurately detect it. Mei et al.[25] introduced the concept of distraction into the camouflaged object segmentation task and developed a new distraction mining approach for distraction identification and removal to help the accurate segmentation of camouflaged objects. Zhai et al.[26] decoupled the feature maps into two specific tasks: One for roughly localizing objects and the other for accurately predicting edge details and iteratively reasoning about their higher-order relationships through the graphs. After this, Fan et al.[7] improved the network and provided three well-designed submodules, namely texture enhanced module, neighbor connection, and group reversal attention blocks to achieve better results. Li et al.[27] proposed joint training of SOD and COD tasks and use of contradictory information to simultaneously improve the performance of both tasks. Zhan et al.[28] employed multi-scale image features to detect concealed cracks on road surfaces.

1.2 Edge-aware learning

Existing methods demonstrate that object edge prediction is more challenging. Besides, correct edge perception is beneficial to performance. Zhao et al.[9] use the complementarity of salient edge information and salient object information to more precisely detect prominent objects and their boundaries. Feng et al.[29] designed a boundary-enhanced loss (BEL) to explore refined boundaries. Qin et al.[30] proposed a new loss function, which fused binary cross-entropy (BCE) loss, structural similarity (SSIM) and intersection-over-union (IOU) loss, for effectively segmentation and accurate boundary prediction. Chen et al.[31] proposed a contour loss, which leveraged the contour to guide the model to obtain more discriminative features, and enhanced local feature representation while preserving object boundaries. He et al.[19] proposed a refined differential module to generate edge cues and then used a point-based GCN to model the global boundary feature learning to detect glass-like objects.

1.3 Vision Transformer

Vaswani et al.[32] first presented a Transformer structure fully based on attention for the subject of natural language processing (NLP). Recently, ViT[33] has applied the Transformer to the field of computer vision and obtained promising results. DETR[34] used the Transformer decoder for end-to-end object detection, considerably simplifying the handcrafted processes. PVT[35] used a progressive shrinking pyramid to reduce the computation and easily embed it into various downstream tasks. GLCNet[36] integrated the local detail extraction capability of CNNs with the global context modeling strength of Transformers to capture robust feature representations under complex remote sensing backgrounds. VOLO[37] introduced outlook attention to efficiently encode finer-level features and context into tokens. DPT[38] extracted tokens from the various stages of the Vision Transformer and used a convolutional decoder to combine them into full resolution representations for finer-grained and more globally predictions. Swin Transformer[16] designed the shifted windows to bring more efficiency by self-attention and achieve cross-window connection. With comprehensive consideration of computation and performance, we choose Swin Transformer as the backbone.

2 The Proposed Method

2.1 Overview

The broad framework of the proposed model is depicted in Fig.1. We first provide an input image as $\mathcal{I}: \mathcal{H} \times \mathcal{W} \times 3$, including one or more object camouflage in the background. The backbone network Swin Transformer[16] is used to generate four multi-scale features, which can be denoted as $\{f_i, i \in \{1, 2, 3, 4\}\}$, respectively. Following the last level feature f_4, a cascaded ASPP[17] module

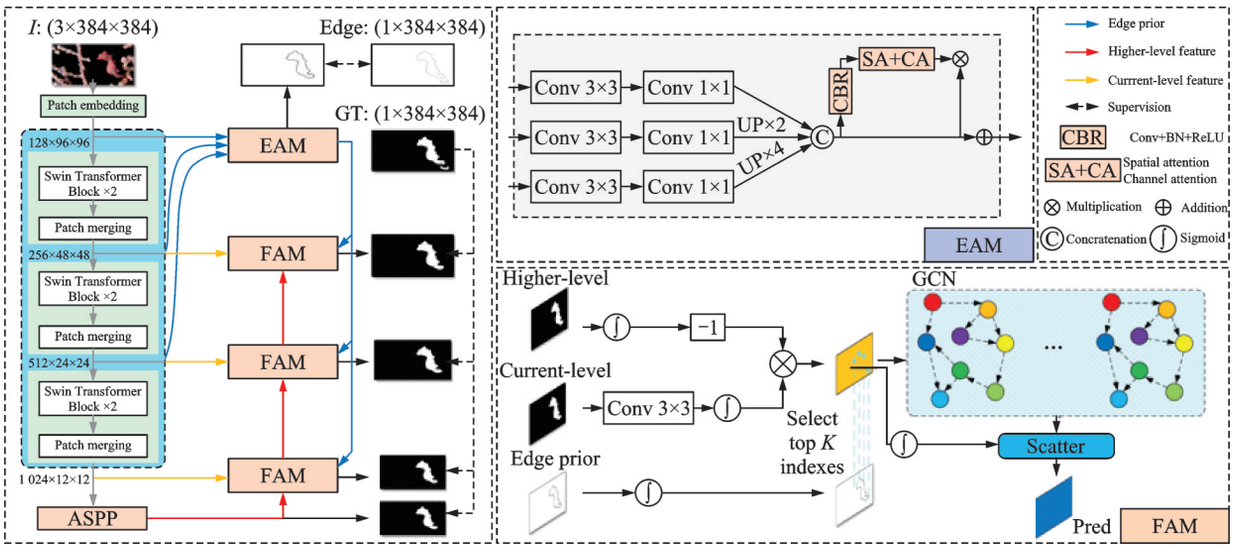

Fig.1 The overall pipeline of the proposed model with three main modules

with four dilation rates $d\in\{1, 6, 12, 18\}$ generates the initial prediction priors $\mathcal{P}_4$.

Then, we design an aggregation structure named EAM to obtain edge priors. New research[39] shows that lower-level features from shallow layers often retain fine-grained information used to construct objects. Thus, we select three-layer shallower features, which are fed to EAM to generate an edge prior. Furthermore, we use the edge ground truth to learn more valuable edge cues to direct the subsequent network. The process can be formulated as

$$E=\mathrm{EAM}(f_1, f_2, f_3) \tag{1}$$

where E denotes the edge prior map produced by EAM, which is supervised by the edge ground truth and serves as guidance for the main detection branch. The detail of EAM is given in Section 2.3.

In order to gradually refine from abstract high-level features to accurate predictions, we propose FAM, which locates potential camouflage objects and mines details by strategically combining multiple prior features, and gradually improves predictions in a way that cascades multiple modules for enhanced reliability. The process can be formulated as

$$P_{i-1}=\mathrm{FAM}(P_i, f_i, E) \tag{2}$$

where P_i denotes feature from previous level FAM (P_4 from ASPP) and f_i the feature from current layer, $i\in\{2, 3, 4\}$. The detail of FAM is given in Secion 2.4.

Finally, the network outputs high-quality prediction maps and edge predictions, which are jointly supervised by the ground truth. The details of loss functions are declared in Section 2.5.

2.2 Swin Transformer backbone

Swin Transformer improves efficiency by self-attention computation within the non-overlapping shifted windows. At the same time, cross-window connections are realized by shifting the window[16]. This architecture has flexibility at different scales, and enables local modeling capabilities while enabling long-range dependence. These properties make it compatible with many vision tasks. Considering factors such as complexity and performance, we choose Swin-B as the feature extraction backbone network.

After input the image, the Swin-B backbone generates a list of feature maps $\{f_i,\ i\in\{1,\ 2,\ 3,\ 4\}\}$, respectively. The feature list $\{f_1, f_2, f_3\}$ is involved in subsequent edge extraction, and $\{f_2, f_3, f_4\}$ participates in feature aggregation in a cascaded manner to generate refined global predictions. In particular, f_4 dilates multi-scale contextual information via ASPP to obtain an initial prior feature, which is defined as

$$P_4=\mathrm{CBR}\Big(\mathrm{Cat}(f_4, F_1^{\mathrm{ASPP}}(f_4), F_6^{\mathrm{ASPP}}(f_4), F_{12}^{\mathrm{ASPP}}(f_4), F_{18}^{\mathrm{ASPP}}(f_4))\Big) \tag{3}$$

where $\mathrm{Cat}(\cdot)$, $\mathrm{CBR}(\cdot)$, and F_i^{ASPP} denote concate-

nation, Conv + BN + ReLU operation and atrous convolution with dilation rate i, respectively.

2.3 Edge-aware module

Deep features are generally recognized to carry more semantic information, whereas shallow features provide more detailed information. As a result, we select three-level shallower features to produce edge features. First, $\{f_1, f_2, f_3\}$ is performed Conv 3×3, Conv 1×1, and upsampling operations to make them have the same size, and then concatenate them to get a rough edge representation.

$$E_r = \mathrm{Cat}\Big(\mathrm{Conv}(f_1), \mathrm{Up}_2(\mathrm{Conv}(f_2)), \mathrm{Up}_4(\mathrm{Conv}(f_3))\Big) \tag{4}$$

where $\mathrm{Up}_i(\cdot)$ means $i\times$ upsampling operation.

Then, E_r through CBR, spatial attention[18] and channel attention[18] (SA + CA) operations, gets the final edge prior after a residual connection.

$$E = E_r \times \mathrm{SA}(\mathrm{CA}(\mathrm{CBR}(E_r))) + E_r \tag{5}$$

Specifically, we can define the channel attention $\mathrm{CA}(\cdot)$ as

$$\mathrm{CA}(x) = \sigma(W_1(\mathrm{avg}^c(x)) + W_2(\mathrm{max}^c(x))) \tag{6}$$

where x denotes the input; $\mathrm{avg}^c(\cdot)$ and $\mathrm{max}^c(\cdot)$ are the average-pooling and max-pooling, respectively; W_1 and W_2 share parameters, and each branch consists of a 1×1 convolution that reduces the number of channels by a factor of 16, followed by a ReLU activation and another 1×1 convolution that restores the channel dimension; σ denotes the sigmoid function.

The spatial attention operation $\mathrm{SA}(\cdot)$ is

$$\mathrm{SA}(x) = \sigma(f^{7\times7}(\mathrm{avg}^s(x) + \mathrm{max}^s(x))) \tag{7}$$

where $\mathrm{avg}^s(\cdot)$ and $\mathrm{avg}^s(\cdot)$ denote the average-pooling and max-pooling operation cross the channel, $f^{7\times7}$ is the convolution with the size of 7×7.

The resulting edge prior will be used to participate in feature aggregation to enhance details.

2.4 Feature aggregation module

A crucial step in obtaining a good feature representation is to mine the target region's hard pixels. After residual connection, the overlooked features can be examined by removing the currently anticipated region at the side output[40]. Specifically, we first perform a convolution operation on higher-level feature and current-level feature to ensure that the number of channels is consistent. After a sigmoid function, the current-level feature is multiplied with the reversed higher-level prediction.

$$F_i^r = \sigma(\mathrm{Conv}(f_i)) \odot (1 - (\sigma(\mathcal{P}_{i+1}))) \quad i\in\{1,2,3\} \tag{8}$$

where $\sigma(\odot)$ denotes the sigmoid function, "$\odot$" the multiple operation, f_i the current-level feature, and $\mathcal{P}_{i+1}$ the output of the previous cascaded block.

To make F_i^r have sharper edge, inspired by Ref.[19], we further feed edge pixels with high confidence into a graph convolution network[41] to learn their intrinsic connections. We first extract top-K high-confidence nodes $\mathcal{V}$, whose value are greater than a certain threshold from the known edge prior E, and sample the same number of points from F_i^r by their indices. In this way, we obtain the sampled set of feature matrix $\boldsymbol{X}^{K\times C}$ with the highest probability, where C denotes the number of channels. In this experiment, K is set to 128. Then we can formulate the graph convolution learning process as

$$\boldsymbol{Z} = \mathrm{ReLU}\left(\hat{\boldsymbol{A}}\boldsymbol{X}\boldsymbol{W}\right) \tag{9}$$

where $\hat{\boldsymbol{A}}^{K\times K}$ is the adjacency matrix representing the graph, $\boldsymbol{X}^{K\times C}$ the input feature matrix, and $\boldsymbol{W}^{C\times C}$ a learnable feature matrix. We construct $\hat{\boldsymbol{A}}$ using a single 1×1 convolution with learnable parameters. After graph convolution, we project the output point feature matrix $\boldsymbol{Z}$ back to F_i^r based on the index. After a 3×3 Conv operation, the enhanced feature is reduced to 1 channel and concatenated with F_i^r. Finally, we get the refined feature $\mathcal{P}_i$. The whole process can be described as

$$\mathcal{P}_{i-1} = F_i^r + \mathrm{Conv}(\mathrm{GCN}(F_i^r)) \quad i\in\{2,3,4\} \tag{10}$$

We finally consider the last $\mathcal{P}_1$ as the final prediction map.

2.5 Loss function

Same as previous research[6-7, 10], we combine two loss components, i.e., the weighted binary cross entropy (wBCE) loss[42] and weighted intersection-over-union (wIOU) loss[42], as our loss function. The two are used to calculate the global

loss and pixel-level loss, respectively. Unlike standard loss functions, the weighted loss focuses more on hard pixels instead of giving the same weights to each pixel. Furthermore, we perform deep supervision on the outputs of the three FAMs ($\{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\}$), the outputs of the EAM (E), and the outputs of the ASPP ($\mathcal{P}_4$). The entire loss function can be expressed as follows

$$L = \lambda_1 \mathcal{L}_{\text{edge}} + \lambda_2 \mathcal{L}_{\text{pred}} \tag{11}$$

$$\mathcal{L}_{\text{edge}} = \mathcal{L}_{\text{wBCE}}(E, E_{\text{g}}) \tag{12}$$

$$\mathcal{L}_{\text{pred}} = \sum_{i \in \{1,2,3,4\}} (\mathcal{L}_{\text{wBCE}}(\mathcal{P}_i, \mathcal{P}_{\text{g}}) + \mathcal{L}_{\text{wIOU}}(\mathcal{P}_i, \mathcal{P}_{\text{g}})) \tag{13}$$

where E_{g} is the edge ground truth and $\mathcal{P}_{\text{g}}$ the mask ground truth; note that we do not use $\mathcal{L}_{\text{wIOU}}$ for edge supervision because of the huge gap between the foreground region and background region which may lead to unstable descent of loss function; λ_1 is set to 10 and λ_2 is set to 1.

3 Experiment

3.1 Datasets and evaluation metrics

3.1.1 Datasets

We conduct experiments on four COD benchmark datasets. CHAMELEON[38] is a small-sample dataset which contains 76 images. CAMO[24] contains 1 000 images for training (CAMO-Train) and 250 images for the testing (CAMO-Test). COD10K[6] is the largest dataset to date, which is divided into five super-classes and 69 sub-class camouflaged object categories, and has 3 040 training images, and 2 026 testing images. NC4K[43] is the largest camouflaged object detection testing dataset, which contains 4 121 images, that can be used to evaluate the generalization ability of the model. Following the proposed model[6], we combine the training sets of COD10K-Train and CAMO-Train to build the final training set and evaluate our model on the others.

3.1.2 Evaluation metrics

In our experiment, we choose the output of the latest FAM module $\mathcal{P}_1$ as the final prediction map, because it incorporates more features from multiple branches. We use four widely-used evaluation metrics: Structure-measure (S_α)[44], mean enhanced-measure (E_Φ)[45], weighted F-measure (F_β^{w})[46], and mean absolute error (MAE)[47].

Structure-measure S_α evaluates region-level and object-level structural similarity between prediction $\mathcal{P}_1$ and ground truth $\mathcal{P}_{\text{g}}$, and the formula can be expressed as

$$S_\alpha = \alpha \cdot S_{\text{o}}(\mathcal{P}_1, \mathcal{P}_{\text{g}}) + (1-\alpha) \cdot S_{\text{r}}(\mathcal{P}_1, \mathcal{P}_{\text{g}}) \tag{14}$$

where S_{o} and S_{r} denote the object-level and region-level structural similarity, respectively. We set α to 0.5 as indicated in Ref.[44].

Mean E-measure E_Φ uses the enhanced matrix ϕ_{FM} to jointly capture image-level statistics and pixel-level matching information.

$$E_\Phi = \frac{1}{w \times h} \sum_{x=1}^{w} \sum_{y=1}^{h} \phi_{\text{FM}}(x, y) \tag{15}$$

Weighted F-measure F_β^{w} defines weighted Precision (Precision$^{\text{w}}$) and weighted Recall (Recall$^{\text{w}}$) to measure exactness and completeness.

$$F_\beta^{\text{w}} = (1+\beta^2) \times \frac{\text{Precision}^{\text{w}} \times \text{Recall}^{\text{w}}}{\beta^2 \times \text{Precision}^{\text{w}} + \text{Recall}^{\text{w}}} \tag{16}$$

where β^2 is set to 0.3 following Ref.[46].

MAE measures the pixel-level difference between the prediction $\mathcal{P}_1$ and the ground truth $\mathcal{P}_{\text{g}}$.

$$\text{MAE} = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} \left| \mathcal{P}_1(x, y) - \mathcal{P}_{\text{g}}(x, y) \right| \tag{17}$$

For a fair comparison, we use the standard evaluation with the same code to calculate the four metrics of all compared models.

3.2 Implementation details

We build the network on the PyTorch framework and train all the experiments on a single NVIDIA 2080Ti GPU. The weights pretrained on ImageNet-22k are used to establish the parameters of the Swin Transformer backbone. Before training, all camouflage pictures and ground truth maps are resized to 384×384 and no augmentation strategies are used. The batch size is set to 6. We use the Adam optimizer, and the initial learning rate is set to 1e−5. After every 30 epochs, the learning rate is divided by 10. The network is trained for a total of 60 epochs, which takes around 5.5 h. During testing, images are also resized to 384×384 and subsequently entered into the network. We only use the

output of the latest FAM as the final prediction map since it incorporates more features from multiple branches. The resulting prediction map is finally scaled to its original size using the bilinear interpolation operation to evaluate the results. The edge ground truth is automatically generated from the original binary masks using a Sobel gradient operator. The resulting gradient maps are used as supervisory signals for the EAM.

3.3 Ablation study

We design ablation experiments by stepwise decoupling each submodule to verify their validity.

3.3.1 Effectiveness of Swin Transformer backbone

We replace Swin Transformer backbone with some CNN backbones (e.g., ResNet-50[48], ResNet-101[48], Res2Net-50[49]) and Transformer backbone (e.g., PVT-M[35]) to verify the validity of the backbone. In all ablation experiments, we only replace the backbone network while keeping the training strategies, loss functions, and other settings strictly consistent (Table 1). The best scores are highlighted in bold.

Table 1 Effectiveness analysis of backbone network, including ResNet-50, ResNet-101, Res2Net-50, PVT-M and Swin-B

Backbone	CHAM[50]				CAMO[24]				COD10K[6]				NC4K[43]			
	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
ResNet-50[48]	0.883	0.938	0.812	0.030	0.780	0.840	0.683	0.083	0.784	0.865	0.633	0.042	0.826	0.889	0.740	0.053
ResNet-101[48]	0.888	0.941	0.810	0.029	0.789	0.849	0.700	0.080	0.822	0.872	0.676	0.038	0.838	0.899	0.757	0.050
Res2Net-50[49]	0.884	0.941	0.805	0.031	0.796	0.847	0.700	0.080	0.803	0.876	0.655	0.040	0.836	0.891	0.748	0.051
PVT-M[35]	0.888	0.942	0.814	0.029	0.833	0.912	0.796	0.057	0.838	0.096	0.712	0.030	0.865	0.920	0.788	0.043
Swin-B[16]	**0.892**	**0.949**	**0.823**	**0.026**	**0.882**	**0.934**	**0.832**	**0.041**	**0.858**	**0.929**	**0.757**	**0.025**	**0.888**	**0.938**	**0.833**	**0.032**

We find that the network performance using Swin Transformer is significantly better than others. We also give the visual comparison of different backbones in Fig.2. From the left to the right, there are input images, ground truth (GT), predictions of ours (Swin Transformer), ResNet-50, Res2Net-50, and PVT-M. From the first and second rows of Fig.2, we can find that the Transformer backbones have more accurate structural prediction than the CNN backbones. It demonstrates the superiority of transformers' long-distance dependencies. From the third row of Fig.2, it shows that camouflage objects with too small size will be ignored in CNN backbones. It may be caused by an excessively large receptive field. In general, Swin backbone is better than others.

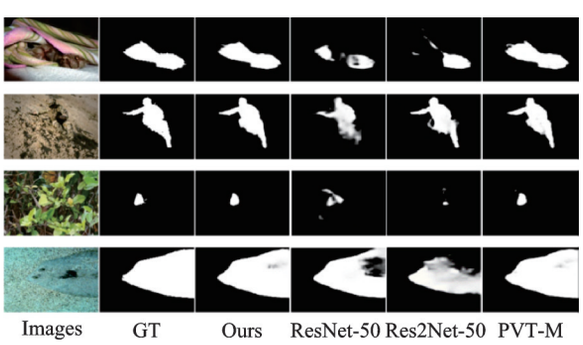

Fig.2 Visual comparison of different backbones

3.3.2 Effectiveness of FAM

To verify the effectiveness of FAM, we conduct ablation studies by gradually decoupling the pipeline of the input. We first remove the higher-level feature branch while keeping the two other branches. Then we add the higher-level guidance into the FAM, but only use a simple convolution on the $\mathcal{P}_4$ (as the higher-level feature of the first FAM) to reduce the number of channels instead of the ASPP. From Table 2, we find that our method wins about 0.018, 0.023, 0.022, 0.005 on S-measure, mean E-measure, weighted F-measure, and MAE, respectively, where w/o is the abbreviation of without. This shows the important role of the higher-level guidance. Moreover, compared with the method without the ASPP module, our full model gets much better scores. This demonstrates that connecting ASPP after the backbone to produce a prior is beneficial for subsequent feature aggregation.

Table 2 Effectiveness analysis of higher-level feature guidance

Ablation variant	CHAM[50]				CAMO[24]				COD10K[6]				NC4K[43]			
	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
w/o higher-level guidance	0.878	0.920	0.802	0.032	0.860	0.920	0.807	0.048	0.840	0.898	0.736	0.029	0.869	0.920	0.812	0.036
w/o ASPP	0.886	0.938	0.813	0.029	0.874	0.924	0.818	0.045	0.850	0.916	0.744	0.026	0.878	0.928	0.820	0.034
w/o FAM 1 & 2 & 3	0.883	0.937	0.803	0.031	0.877	0.924	0.819	0.046	0.849	0.920	0.738	0.031	0.878	0.925	0.820	0.038
w/o FAM 2 & 3	0.888	0.937	0.813	0.030	0.879	0.927	0.823	0.044	0.856	0.925	0.748	0.029	0.884	0.934	0.827	0.034
w/o FAM 3	**0.892**	0.940	0.819	0.027	0.881	0.928	0.827	0.044	**0.858**	0.926	0.751	0.027	0.885	0.934	0.827	0.034
Ours	**0.892**	**0.949**	**0.823**	**0.026**	**0.882**	**0.934**	**0.832**	**0.041**	**0.858**	**0.929**	**0.757**	**0.025**	**0.888**	**0.938**	**0.833**	**0.032**

We further justify the effectiveness of the FAM by substituting it with a lightweight alternative that performs cross-level feature fusion via edge-guided spatial weighting. Our ablation protocol involves removing the FAM from the deepest layer (testing high-level GCN necessity), the mid-high layers (testing redundancy), and all layers. Quantitative results in Table 2 reveal that the S-measure remains relatively stable during progressive removal, whereas pixel-wise precision drops drastically. The model maintains coarse structural awareness but loses fine details. Crucially, ablating the shallowest FAM leads to a catastrophic failure, underscoring that the refinement of high-resolution features via the FAM is indispensable for pixel-accurate saliency mapping.

We further conduct ablation studies on edge prior branch to verify its effect on feature aggregation.

3.3.3 Effectiveness of EAM

We completely remove the edge prior from the EAM to verify its effectiveness. From the column 3 and column 4 of Fig.3 we can find that the edge guidance enhances the edge details of the prediction map. We further preserve the edge prior but use a simple concatenation instead of GCN operations in the FAM. The column 5 of Fig.3 shows its ability for extracting hard pixels of edge. In addition, compared with the full network, Table 3 shows that sharper edge information supplement improves network performance.

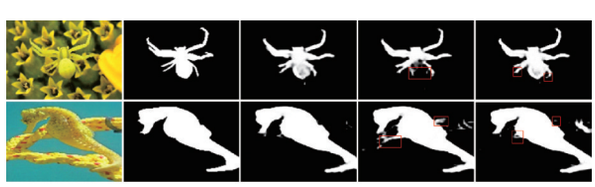

Fig.3 Effectiveness of edge guidance

Table 3 Effectiveness analysis of edge guidance and edge-based GCN operation

Ablation variant	CHAM[50]				CAMO[24]				COD10K[6]				NC4K[43]			
	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
w/o edge guidance	0.884	0.840	0.812	0.030	0.863	0.881	0.800	0.049	0.819	0.912	0.725	0.028	0.867	0.902	0.782	0.046
w/o GCN	0.889	0.940	0.816	0.028	0.878	0.929	0.827	0.043	0.854	0.920	0.749	0.026	0.881	0.930	0.821	0.035
Ours	**0.892**	**0.949**	**0.823**	**0.026**	**0.882**	**0.934**	**0.832**	**0.041**	**0.858**	**0.929**	**0.757**	**0.025**	**0.888**	**0.938**	**0.833**	**0.032**

3.3.4 Sensitivity analysis on the number of edge points K

To investigate the impact of Top-K edge pixel selection on model performance, we conduct the following ablation study. As presented in Table 4, selecting a small number of K leads to insufficient information within the graph structure, preventing the graph network from accurately refining continuous edges. Conversely, selecting an excessively large K introduces low-confidence edge points as noise, which disrupts the structural integrity of the predictions. On balance, we determine that $K=128$ is the optimal value for achieving robust performance.

3.3.5 Sensitivity analysis on loss weights

To investigate the impact of the loss weights on training, we conduct an ablation study by varying the edge loss coefficient $\lambda_1\in\{1, 5, 10, 15\}$ while keeping other training parameters constant. The quantitative results are summarized in Table 5.

Table 4 Ablation study on the number of edge points K

Number of edge point K	CHAM[50]				CAMO[24]				COD10K[6]				NC4K[43]			
	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
64	0.887	0.937	0.809	0.030	0.879	0.926	0.821	0.044	0.857	0.924	0.748	0.026	0.885	0.933	0.825	0.034
256	0.890	0.942	0.815	0.029	0.881	0.928	0.825	0.043	0.857	0.924	0.749	0.027	0.886	0.934	0.827	0.034
128	**0.892**	**0.949**	**0.823**	**0.026**	**0.882**	**0.934**	**0.832**	**0.041**	**0.858**	**0.929**	**0.757**	**0.025**	**0.888**	**0.938**	**0.833**	**0.032**

Table 5 Sensitivity analysis of the loss weight λ

Loss weight λ	CHAM[50]				CAMO[24]				COD10K[6]				NC4K[43]			
	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
$\lambda_1=1$, $\lambda_2=1$	0.889	0.940	0.814	0.029	0.880	0.930	0.827	0.042	0.856	0.926	0.750	0.026	0.887	0.935	0.828	0.034
$\lambda_1=5$, $\lambda_2=1$	0.890	0.945	0.818	0.028	0.882	0.931	0.829	0.043	**0.859**	0.927	0.753	0.026	**0.888**	**0.938**	0.831	0.034
$\lambda_1=10$, $\lambda_2=1$	**0.892**	**0.949**	**0.823**	**0.026**	**0.882**	**0.934**	**0.832**	**0.041**	0.858	**0.929**	**0.757**	**0.025**	**0.888**	**0.938**	**0.833**	**0.032**
$\lambda_1=15$, $\lambda_2=1$	0.887	0.938	0.816	0.031	0.878	0.930	0.826	0.044	0.856	0.925	0.750	0.026	0.882	0.931	0.823	0.037

Experimental results indicate that a small λ_1 causes a slight decrease in S_α, and the predictions maintain basic structural integrity. However, metrics such as E_Φ, F_β^w and M drop sharply. This suggests that an insufficient edge loss weight leads to severe pixel-level prediction bias. Conversely, when λ_1 is set too high, the model overemphasizes edge details at the expense of the overall pixel structure.

3.4 Comparison with the SOTA methods

We compare our method with 12 existing COD methods, including BASNet[30], EGNet[9], CPD[51], F^{3}Net[42], PraNet[8], SINet[6], PFNet[25], C2FNet[52], SINetV2[7], LSR[43], UGTR[53], ZoomNet[54], DTINet[55], and TPRNet[56]. For a fair comparison, we directly use the prediction maps provided by the authors and evaluate them with the same formula. If the prediction map is lacking, we generate the prediction maps using the author's pretrained model.

3.4.1 Quantitative results

We summarize the quantitative results of different baseline models on the four datasets (Table 6). The best scores are highlighted in bold. From the re-

Table 6 Quantitative results of different datasets (including CHAM, CAMO, COD10K and NC4K) on four evaluation metrics

Baseline	CHAM[50]				CAMO[24]				COD10K[6]				NC4K[43]			
	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\Phi\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
BASNet[30]	0.687	0.721	0.474	0.118	0.618	0.661	0.413	0.159	0.634	0.678	0.365	0.105	0.695	0.785	0.546	0.095
EGNet[9]	0.848	0.870	0.702	0.050	0.732	0.768	0.583	0.104	0.737	0.779	0.509	0.056	0.777	0.864	0.639	0.075
CPD[51]	0.853	0.866	0.706	0.052	0.726	0.729	0.550	0.115	0.747	0.770	0.508	0.059	0.787	0.852	0.696	0.072
F3Net[42]	0.848	0.917	0.744	0.047	0.711	0.780	0.564	0.109	0.739	0.819	0.544	0.051	0.780	0.848	0.656	0.070
PraNet[8]	0.860	0.907	0.760	0.044	0.769	0.824	0.663	0.094	0.789	0.861	0.629	0.045	0.822	0.877	0.724	0.059
SINet[6]	0.869	0.891	0.740	0.044	0.751	0.771	0.606	0.100	0.771	0.806	0.551	0.051	0.808	0.876	0.723	0.058
PFNet[25]	0.882	0.942	0.810	0.033	0.782	0.852	0.695	0.085	0.800	0.868	0.660	0.040	0.839	0.892	0.745	0.053
C2FNet[52]	0.888	0.946	0.828	0.032	0.820	0.864	0.752	0.066	0.813	0.890	0.636	0.036	0.838	0.898	0.762	0.049
SINetV2[7]	0.888	0.942	0.816	0.030	0.820	0.882	0.743	0.070	0.815	0.887	0.680	0.037	0.847	0.905	0.770	0.048
LSR[43]	0.890	0.948	0.820	0.030	0.787	0.852	0.696	0.080	0.804	0.889	0.673	0.037	0.840	0.907	0.770	0.048
UGTR[53]	0.888	0.918	0.794	0.031	0.785	0.859	0.686	0.086	0.818	0.850	0.667	0.035	0.839	0.892	0.746	0.052
ZoomNet[54]	**0.902**	**0.958**	**0.845**	**0.023**	0.820	0.892	0.752	0.066	0.838	0.911	0.729	0.029	0.853	0.912	0.784	0.043
TPRNet[55]	0.891	0.930	0.816	0.031	0.814	0.870	0.781	0.076	0.829	0.892	0.725	0.034	0.854	0.903	0.790	0.047
DTINet[56]	0.883	0.928	0.813	0.033	0.857	0.912	0.796	0.050	0.824	0.893	0.695	0.034	0.863	0.915	0.792	0.041
Ours	0.892	0.949	0.823	0.026	**0.882**	**0.934**	**0.832**	**0.041**	**0.858**	**0.929**	**0.757**	**0.025**	**0.888**	**0.938**	**0.833**	**0.032**

sults we can find that Zoom, as the latest method, outperforms other methods on each dataset, and our method outperforms it on three datasets (CAMO, COD10K, NC4K). Specifically, our method has a 4.5% improvement on S_α, 2.8% improvement on the mean E_ϕ, 6.4% improvement on the average F_β^w, and 0.004—0.024 decrease on MAE. Our method outperforms all competitors on three datasets (CAMO, COD10K, NC4K) and is the second best on one dataset (CHAMELEON). We further provide Precision-Recall and F-measure curves of our method and other 11 state-of-the-art (SOTA) models. As shown in Fig. 4, ours performs better than other models.

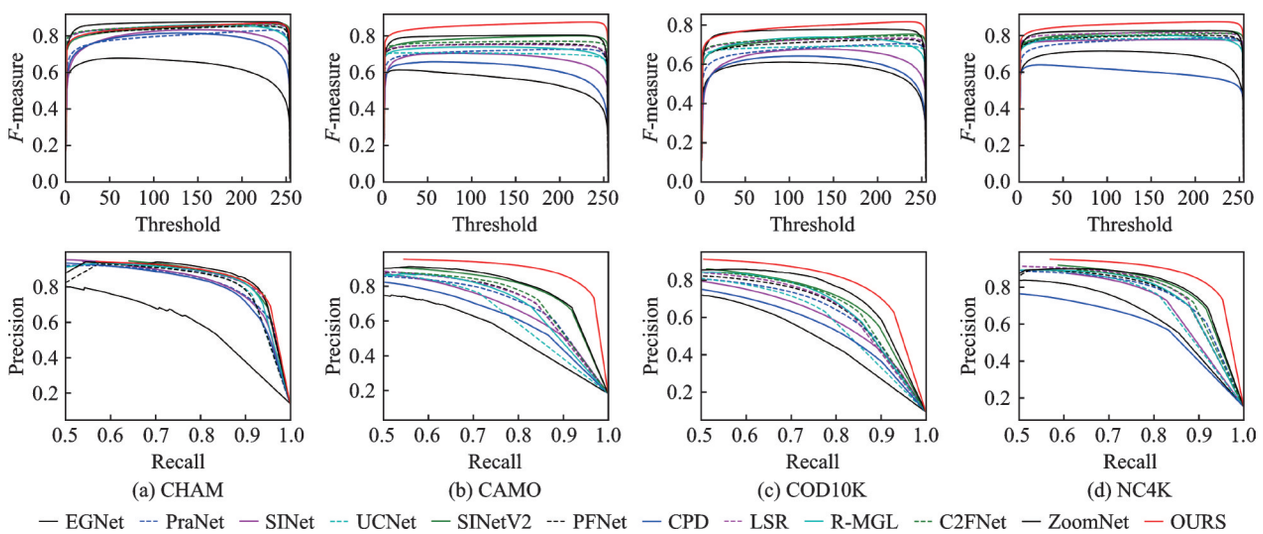

Fig.4 F-measure curves (the 1st row) and precision curves (the 2nd row) of 12 methods on four benchmark datasets

3.4.2 Visual comparison

Fig.5 is the result of comparing our method with other SOTA models. It can be observed that our method outperforms others in predicting the camouflage objects.

In particularly, our method achieves more robust edge representations when all methods accurately localize objects (e.g., the 6th row).

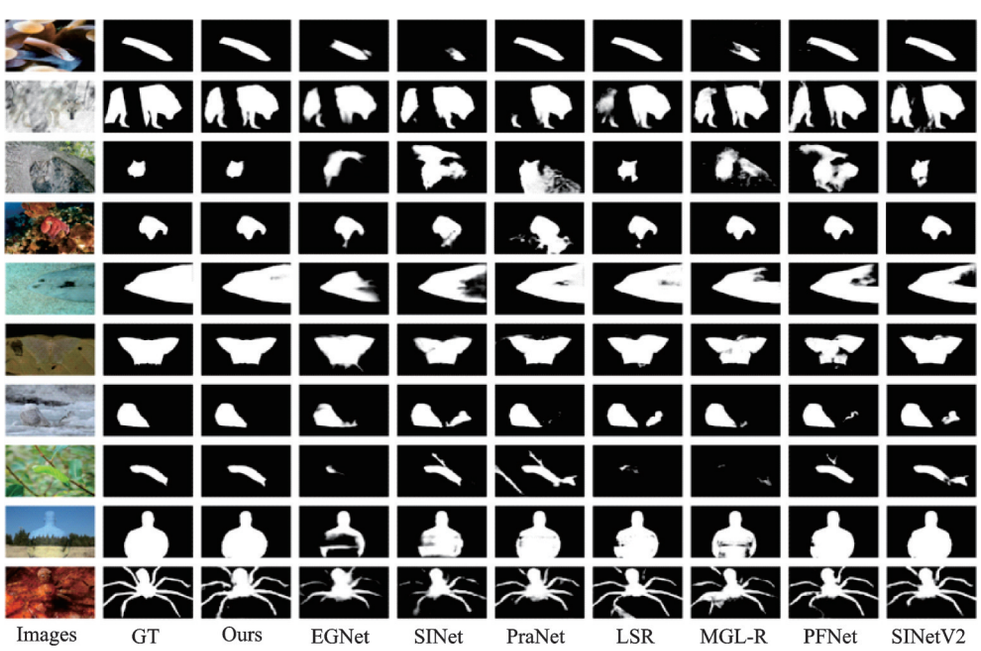

Fig.5 Vision comparison of the proposed model with SOTA methods

3.4.3 Analysis of speed and model complexity

Table 7 shows the speed and computational complexities of our method and seven other latest models. We do experiments on a NVIDIA 2080Ti GPU. The hyperparameters refer to the settings in the corresponding papers. Our model has 109.41M parameters and requires 48.56G floating-point operations (FLOPs). The complexity of the model is mainly in the Swin-B backbone part which has 88M parameters and costs 47G FLOPs. Despite the large number of model parameters, it can still run at the real-time speed of 46 FPS.

Table 7 Speed and model complexity analysis on multiple models

Method	Ours	SINet[6]	PFNet[25]	UGTR[53]	MGL-R[26]	LSR[43]	SINetV2[7]	ZoomNet[54]
Paramster	109.41M	48.95M	46.50M	48.87M	63.58M	50.95M	24.93M	32.38M
FLOPs	48.56G	19.42G	19.00G	1.00T	549.62G	66.64G	12.24G	203.40G
FPS	46	56	62	18	12	56	52	25

3.5 Failure cases

Although our method has achieved excellent accuracy, it may still fail in some challenging scenarios. We show partially failed cases in Fig.6. We argue that the main reasons for failure are:

(1) Although we employ edge guidance to reduce the effect of blurred bounds, it is still readily misled in scenarios with many obfuscated object boundaries (the 1st row of Fig.6).

(2) When the scale of the object is excessively small, the repeated downsampling operations lead to the neglect or even complete loss of object features, thereby preventing accurate localization (the 2nd row of Fig.6).

(3) While the GCN performs well on regions with continuous edges, it struggles with irregular and fragmented occlusions. In such scenarios, fine boundaries fail to generate a sufficient number of graph nodes. Meanwhile, the disconnected and distant small regions restrict the connectivity between nodes, preventing the model from accurately capturing the edges of multiple regions (the last row of Fig.6).

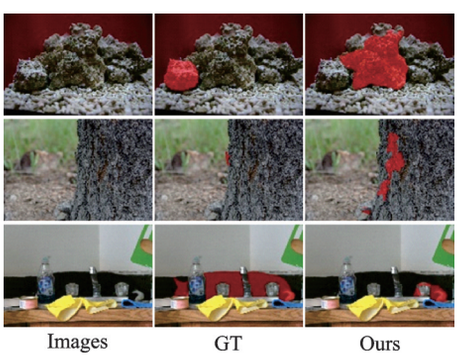

Fig.6 Several failure cases

In order to solve the above problems, we argue that it can be alleviated by fine-grained network structures, augmented datasets or high-resolution training datasets. These await our further research.

4 Discussion

In this study, we propose a novel COD framework based on the Swin Transformer, incorporating an edge-aware module and a graph-based feature aggregation strategy. The proposed method achieves superior performance over existing approaches on multiple public benchmark datasets. This section provides an in-depth analysis and discussion of the experimental results from five perspectives: Module effectiveness, trade-off between accuracy and efficiency, generalization ability, model limitations, and future research directions.

4.1 Effectiveness of the proposed modules

Ablation studies demonstrate that each component of the proposed network contributes significantly to performance improvements. Compared with traditional CNNs (e.g., ResNet, Res2Net) and lightweight Transformer backbones (e.g., PVT), Swin Transformer offers stronger capability in modeling long-range dependencies and capturing global context, thus providing a more robust feature representation for COD tasks. The EAM enhances boundary representations by aggregating low-level

features and applying spatial and channel attention mechanisms, leading to more accurate boundary prediction. The FAM, guided by graph convolution networks, effectively models the intrinsic relationships between high-confidence edge points, refining multi-level features in a cascaded manner and significantly improving the clarity and precision of the final predictions.

4.2 Trade-off between accuracy and efficiency

Although the proposed model has a relatively large number of parameters (109.41M) and higher computational complexity (48.56G FLOPs), it still achieves the real-time inference speed of 46 FPS on a single NVIDIA 2080Ti GPU. This indicates that the method balances detection accuracy and computational efficiency, making it suitable for practical applications.

4.3 Generalization ability

To further evaluate the generalization capability of the proposed model, we apply it to several downstream tasks that share similarities with COD, including polyp segmentation, lung infection detection, camouflaged people detection, and road crack detection. Experimental results demonstrate that our method consistently yields strong performance across these tasks, suggesting that the proposed approach is not only effective in COD but also transferable to medical imaging and industrial inspection applications.

4.4 Limitations

Despite its promising performance, the proposed method has several limitations. In scenarios with extremely blurred boundaries or very small-scale targets, the model may fail to accurately localize the object. Additionally, in heavily occluded scenes, the model can misclassify due to irregular shapes and ambiguous cues. Furthermore, the relatively complex network structure may pose challenges for deployment in resource-constrained environments.

4.5 Downstream applications

To further confirm our method's generalization performance, we do validation studies on the four downstream applications listed below.

(1) Polyp segmentation

Similarly to camouflaged objects, polyp segmentation in colonoscopy images is a challenging task. Following Ref.[8], we first retrain our model on ClinicDB[57] and Kvasir-SEG[58] training datasets to verify its generalization ability in the medical imaging field. We then use two unseen test datasets, namely ETIS[59] and ColonDB[60], to evaluate the results. Table 8 shows the results of our method compared with SOTA methods on four metrics (including Dice$^{\max}$, S_α, $E_\phi^{\max}$ and F_β^{w}). The visualization results can be seen in Fig.7(a).

Table 8 Quantitative results on ETIS[59] and ColonDB[60] test datasets

Baseline	CHAM[50]				CAMO[24]			
	Dice$^{\max}$ ↑	S_α ↑	$E_\phi^{\max}$ ↑	F_β^{w} ↑	Dice$^{\max}$ ↑	S_α ↑	$E_\phi^{\max}$ ↑	F_β^{w} ↑
UNet[61]	0.444	0.684	0.740	0.366	0.512	0.710	0.780	0.494
UNet++[62]	0.509	0.683	0.776	0.390	0.660	0.692	0.760	0.467
PraNet[8]	0.638	0.794	0.841	0.600	0.728	0.820	0.872	0.699
Ours	**0.732**	**0.860**	**0.881**	**0.704**	**0.785**	**0.858**	**0.910**	**0.765**

(2) Lung infection segmentation

To further validate the effect of our method on medical images, we retrain our model on the COVID-19 lung infection segmentation dataset[63]. From Fig.7(b), we can find that even if the contrast between the infected area and other parts in the CT image is extremely low, our method can still perform preliminary segmentation.

(3) Camouflage people detection

People will hide in some complicated situations due to particular camouflage. The study of persons dressed in certain camouflage patterns has crucial consequences for certain special vocations (e.g., field rescue, military). We validate our method on a

special camouflaged people dataset[64]. The visualization results are shown in Fig.7(c).

(4) Defect detection

Defects in industrial products endanger human safety. However, due to their irregularity and topological complexity, defects are frequently difficult to discover. We retrain our model using CrackForest[65], a road crack dataset. Despite the small number of sample (only 156 images), our approach produces good results (Fig.7(d)).

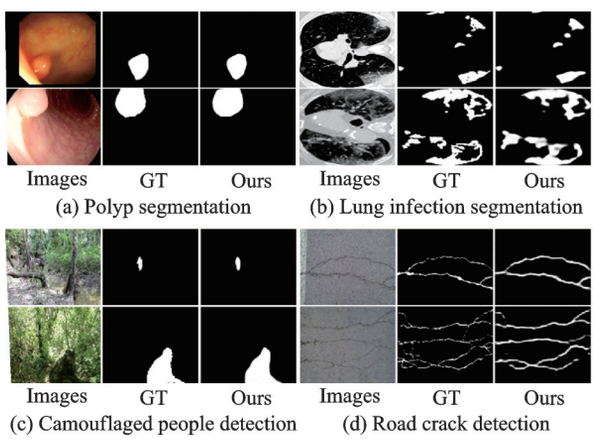

Fig.7 Visualization results for four downstream applications

5 Conclusions

This paper proposes a novel approach for the COD task. First, Swin Transformer is employed to generate robust feature representations. Furthermore, the EAM is adopted to yield more discriminative edges, while the FAM further refines edge details and enhances inter-layer information fusion. Compared with 12 state-of-the-art COD methods, the proposed method exhibits excellent performance and strong generalization ability on four benchmark datasets. Additionally, the method achieves satisfactory results in several experiments on downstream applications, which indicates its certain application value.

Future efforts can be directed toward the following aspects: (1) Integrating stronger multi-scale attention and structure-aware modules to enhance detection of small or highly camouflaged objects; (2) exploring tighter integration of graph neural networks and Transformer architectures to improve cross-level feature interaction; (3) designing more lightweight and efficient architectures for deployment on mobile or edge devices; (4) constructing more diverse and challenging synthetic camouflage datasets, and employing data augmentation techniques to improve model robustness and adaptability.

References

[1] VALLIN A, JAKOBSSON S, LIND J, et al. Prey survival by predator intimidation: An experimental study of peacock butterfly defence against blue tits[J]. Proceedings of the Royal Society B: Biological Sciences, 2005, 272(1569): 1203-1207.

[2] DUARTE R C, FLORES A A V, STEVENS M. Camouflage through colour change: Mechanisms, adaptive value and ecological significance[J]. Philosophical Transactions of the Royal Society B: Biological Sciences, 2017, 372(1724): 20160342.

[3] SRINIVASAN M V, DAVEY M. Strategies for active camouflage of motion[J]. Proceedings of the Royal Society of London Series B: Biological Sciences, 1995, 259: 19-25.

[4] SCHAEFER H M, STOBBE N. Disruptive coloration provides camouflage independent of background matching[J]. Proceedings Biological Sciences, 2006, 273(1600): 2427-2432.

[5] MERILAITA S, SCOTT-SAMUEL N E, CUTHILL I C. How camouflage works[J]. Philosophical Transactions of the Royal Society B: Biological Sciences, 2017, 372(1724): 20160341.

[6] FAN D P, JI G P, SUN G L, et al. Camouflaged object detection[C]//Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Seattle, WA, USA: IEEE, 2020: 2774-2784.

[7] FAN D P, JI G P, CHENG M M, et al. Concealed object detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44: 6024-6042.

[8] FAN D P, JI G P, ZHOU T, et al. PraNet: Parallel reverse attention network for polyp segmentation[C]//Proceedings of Medical Image Computing and Computer Assisted Intervention—MICCAI 2020. Cham: Springer, 2020: 263-273.

[9] ZHAO J X, LIU J J, FAN D P, et al. EGNet: Edge guidance network for salient object detection[C]//Pro-

ceedings of 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea: IEEE, 2019: 8778-8787.

[10] DING M D, QU A P, ZHONG H Q, et al. An enhanced Vision Transformer with wavelet position embedding for histopathological image classification[J]. Pattern Recognition, 2023, 140: 109532.

[11] ZHANG S, FU Q, DUAN J, et al. Low contrast target polarization recognition technology based on lifting wavelet[J]. Acta Optica Sinica, 2015, 35(2): 0211002.

[12] ZHANG S, PENG J, ZHAN J T, et al. Research of the influence of non-spherical ellipsoid particle parameter variation on polarization characteristic of light[J]. Acta Physica Sinica, 2016, 65(6): 064205.

[13] ZHANG Y, FU Q, LUO K M, et al. Analysis of two-color infrared polarization imaging characteristics for target detection and recognition[J]. Photonics, 2023, 10(11): 1181.

[14] WANG J Y, SHI H D, LIU J N, et al. Compressive space-dimensional dual-coded hyperspectral polarimeter (CSDHP) and interactive design method[J]. Optics Express, 2023, 31(6): 9886-9903.

[15] ZHANG S, ZHAN J T, FU Q, et al. Simulation research on sky polarization characteristics under complicated marine environment[J]. Acta Optica Sinica, 2020, 40(22): 2201001.

[16] LIU Z, LIN Y T, CAO Y, et al. Swin Transformer: Hierarchical Vision Transformer using shifted windows[C]//Proceedings of 2021 IEEE/CVF International Conference on Computer Vision (ICCV). Montreal, QC, Canada: IEEE, 2022: 9992-10002.

[17] CHEN L C, PAPANDREOU G, KOKKINOS I, et al. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(4): 834-848.

[18] WOO S, PARK J, LEE J Y, et al. CBAM: Convolutional block attention module[C]//Proceedings of Computer Vision—ECCV 2018. Cham: Springer, 2018: 3-19.

[19] HE H, LI X T, CHENG G L, et al. Enhanced boundary learning for glass-like object segmentation[C]//Proceedings of 2021 IEEE/CVF International Conference on Computer Vision (ICCV). Montreal, QC, Canada: IEEE, 2022: 15839-15848.

[20] GALUN M, SHARON E, BASRI R, et al. Texture segmentation by multiscale aggregation of filter responses and shape elements[C]//Proceedings of the Ninth IEEE International Conference on Computer Vision. Nice, France: IEEE, 2008: 716-723.

[21] PAN Y X, CHEN Y W, FU Q, et al. Study on the camouflaged target detection method based on 3D convexity[J]. Modern Applied Science, 2011, 5(4): 152-157.

[22] BHAJANTRI N U, NAGABHUSHAN P. Camouflage defect identification: A novel approach[C]//Proceedings of the 9th International Conference on Information Technology (ICIT'06). Bhubaneswar, India: IEEE, 2007: 145-148.

[23] YIN Jianqin, HAN Yanbin, HOU Wendi, et al. Detection of the mobile object with camouflage color under dynamic background based on optical flow[J]. Procedia Engineering, 2011, 15: 2201-2205.

[24] LE T N, NGUYEN T V, NIE Z L, et al. Anabranch network for camouflaged object segmentation[J]. Computer Vision and Image Understanding, 2019, 184: 45-56.

[25] MEI H Y, JI G P, WEI Z Q, et al. Camouflaged object segmentation with distraction mining[C]//Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Nashville, TN, USA: IEEE, 2021: 8768-8777.

[26] ZHAI Q, LI X, YANG F, et al. Mutual graph learning for camouflaged object detection[C]//Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Nashville, TN, USA: IEEE, 2021: 12992-13002.

[27] LI A X, ZHANG J, LV Y Q, et al. Uncertainty-aware joint salient object and camouflaged object detection[C]//Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Nashville, TN, USA: IEEE, 2021: 10066-10076.

[28] ZHAN Biheng, SONG Xiangyu, CHENG Jianrui, et al. Pavement crack extraction based on multi-scale convolutional neural network[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2025, 42(6): 749-766.

[29] FENG M Y, LU H C, DING E R. Attentive feedback network for boundary-aware salient object detection[C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, USA: IEEE, 2020: 1623-1632.

[30] QIN X B, ZHANG Z C, HUANG C Y, et al. BASNet: Boundary-aware salient object detection[C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, USA: IEEE, 2020: 7471-7481.

[31] CHEN Z X, ZHOU H J, LAI J H, et al. Contour-aware loss: Boundary-aware learning for salient object segmentation[J]. IEEE Transactions on Image Processing, 2021, 30: 431-443.

[32] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017). Long Beach, CA, USA: Neural Information Processing Systems, 2017.

[33] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An image is worth 16×16 words: Transformers for image recognition at scale[EB/OL]. (2020-10-22). https://arxiv.org/abs/2010.11929.

[34] CARION N, MASSA F, SYNNAEVE G, et al. End-to-end object detection with Transformers[C]//Proceedigns of Computer Vision—ECCV 2020. Cham: Springer, 2020: 213-229.

[35] WANG W H, XIE E Z, LI X, et al. Pyramid Vision Transformer: A versatile backbone for dense prediction without convolutions[C]//Proceedings of 2021 IEEE/CVF International Conference on Computer Vision (ICCV). Montreal, QC, Canada: IEEE, 2022: 548-558.

[36] WEI Kan, LI Ling, LIANG Shilin, et al. GLC-Net: Global-local collaborative network for remote sensing image segmentation[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2025, 42(5): 565-576.

[37] YUAN L, HOU Q B, JIANG Z H, et al. VOLO: Vision outlooker for visual recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(5): 6575-6586.

[38] RANFTL R, BOCHKOVSKIY A, KOLTUN V. Vision Transformers for dense prediction[C]//Proceedings of 2021 IEEE/CVF International Conference on Computer Vision (ICCV). Montreal, QC, Canada: IEEE, 2022: 12159-12168.

[39] ZHAO T, WU X Q. Pyramid feature attention network for saliency detection[C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, USA: IEEE, 2020: 3080-3089.

[40] CHEN S H, TAN X L, WANG B, et al. Reverse attention for salient object detection[C]//Proceedings of Computer Vision—ECCV 2018. Cham: Springer, 2018: 236-252.

[41] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks[EB/OL]. (2016-09-09). http://arxiv.org/abs/1609.02907.

[42] WEI J, WANG S H, HUANG Q M. F^3Net: Fusion, feedback and focus for salient object detection[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(7): 12321-12328.

[43] LV Y Q, ZHANG J, DAI Y C, et al. Simultaneously localize, segment and rank the camouflaged objects[C]//Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Nashville, TN, USA: IEEE, 2021: 11586-11596.

[44] CHENG M M, FAN D P. Structure-measure: A new way to evaluate foreground maps[J]. International Journal of Computer Vision, 2021, 129(9): 2622-2638.

[45] FAN D P, GONG C, CAO Y, et al. Enhanced-alignment measure for binary foreground map evaluation[C]//Proceedings of the 27th International Joint Conference on Artificial Intelligence. Stockholm, Sweden: ACM, 2018: 698-704.

[46] MARGOLIN R, ZELNIK-MANOR L, TAL A. How to evaluate foreground maps[C]//Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA: IEEE, 2014: 248-255.

[47] PERAZZI F, KRÄHENBÜHL P, PRITCH Y, et al. Saliency filters: Contrast based filtering for salient region detection[C]//Proceedings of 2012 IEEE Conference on Computer Vision and Pattern Recognition. Providence, RI, USA: IEEE, 2012: 733-740.

[48] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition[C]//Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA: IEEE, 2016: 770-778.

[49] GAO S H, CHENG M M, ZHAO K, et al. Res2Net: A new multi-scale backbone architecture[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 43(2): 652-662.

[50] SKUROWSKI P, ABDULAMEER H, BLASZCZYK J, et al. CHAMELEON: A dataset for camou-

flaged object detection[EB/OL]. (2024-12-02). https:doi.org/10.57702/t1qv79ld.

[51] WU Z, SU L, HUANG Q M. Cascaded partial decoder for fast and accurate salient object detection[C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, USA: IEEE, 2020: 3902-3911.

[52] SUN Y J, CHEN G, ZHOU T, et al. Context-aware cross-level fusion network for camouflaged object detection[EB/OL]. (2021-05-26). https://arxiv.org/abs/2105.12555.

[53] YANG F, ZHAI Q, LI X, et al. Uncertainty-guided Transformer reasoning for camouflaged object detection[C]//Proceedings of 2021 IEEE/CVF International Conference on Computer Vision (ICCV). Montreal, QC, Canada: IEEE, 2022: 4126-4135.

[54] PANG Y W, ZHAO X Q, XIANG T Z, et al. Zoom in and out: A mixed-scale triplet network for camouflaged object detection[C]//Proceedings of 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). New Orleans, LA, USA: IEEE, 2022: 2150-2160.

[55] LIU Z Y, ZHANG Z L, TAN Y C, et al. Boosting camouflaged object detection with dual-task interactive Transformer[C]//Proceedings of 2022 26th International Conference on Pattern Recognition (ICPR). Montreal, QC, Canada: IEEE, 2022: 140-146.

[56] ZHANG Q, GE Y L, ZHANG C, et al. TPRNet: Camouflaged object detection via transformer-induced progressive refinement network[J]. The Visual Computer, 2023, 39(10): 4593-4607.

[57] BERNAL J, SÁNCHEZ F J, FERNÁNDEZ-ESPARRACH G, et al. WM-DOVA maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians[J]. Computerized Medical Imaging and Graphics, 2015, 43: 99-111.

[58] JHA D, SMEDSRUD P H, RIEGLER M A, et al. Kvasir-SEG: A segmented polyp dataset[C]//Proceedings of MultiMedia Modeling. Cham: Springer, 2020: 451-462.

[59] SILVA J, HISTACE A, ROMAIN O, et al. Toward embedded detection of polyps in WCE images for early diagnosis of colorectal cancer[J]. International Journal of Computer Assisted Radiology and Surgery, 2014, 9(2): 283-293.

[60] BERNAL J, SÁNCHEZ J, VILARIÑO F. Towards automatic polyp detection with a polyp appearance model[J]. Pattern Recognition, 2012, 45(9): 3166-3182.

[61] RONNEBERGER O, FISCHER P, BROX T. U-Net: Convolutional networks for biomedical image segmentation[C]//Proceedings of Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015. Cham: Springer, 2015: 234-241.

[62] ZHOU Z W, RAHMAN SIDDIQUEE M M, TAJBAKHSH N, et al. UNet++: A nested U-Net architecture for medical image segmentation[C]//Proceedings of Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support. Cham: Springer, 2018: 3-11.

[63] FAN D P, ZHOU T, JI G P, et al. Inf-Net: Automatic COVID-19 lung infection segmentation from CT images[J]. IEEE Transactions on Medical Imaging, 2020, 39(8): 2626-2637.

[64] ZHENG Y F, ZHANG X W, WANG F, et al. Detection of people with camouflage pattern via dense deconvolution network[J]. IEEE Signal Processing Letters, 2019, 26(1): 29-33.

[65] SHI Y, CUI L M, QI Z Q, et al. Automatic road crack detection using random structured forests[J]. IEEE Transactions on Intelligent Transportation Systems, 2016, 17(12): 3434-3445.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China (No. 62127813); the Joint Funds of the National Natural Science Foundation of China (No.U2341226); and the Jilin Provincial Talent Special Project (No.20240602015RC).

Authors

The first author Mr. **WANG Xiaoyi** was born in 1992. He is a Ph.D. candidate in optoelectronic information engineering at Changchun University of Science and Technology. His primary research interests focus on intelligent processing of photoelectric images, detection of camouflaged targets, and imaging control of photoelectric payloads.

The corresponding author Prof. **FU Qiang** was born in 1984. He is an researcher and doctoral supervisor of Changchun University of Science and Technology. His research interests are optical transmission characteristics testing, multi-dimensional optical imaging, and multi-point and multi-functional space laser communication research.

Author contributions Mr. **WANG Xiaoyi** designed the study, interpreted the results, and wrote the manuscript. Mr. **LI Mingyan** complied the models and conducted the

analysis. Dr. **LI Bin** conducted the analysis. Dr. **WU Fanlu** interpreted the results. Dr. **ZHANG Mingqiang** interpreted the results. Mr. **WANG Guan**, Dr. **ZHU Rui**, Prof. **CAI Hua**, and Prof. **FU Qiang** interpreted the results. All authors commented on the manuscript draft and approved the submission.

Competing interests The authors declare no competing interests.

(Production Editor: XU Chengting)

基于Swin Transformer的图引导边缘感知学习伪装目标检测

王潇逸[1,2,3]，李明岩[3]，李 彬[3]，吴凡路[3]，张明强[3]，王 冠[1,2]，朱 瑞[1,2]，才 华[1]，付 强[1,2]

(1.长春理工大学空间光电技术国家地方联合工程研究中心,长春 130022,中国; 2.长春理工大学光电工程学院,长春 130022,中国; 3.中国科学院长春光学精密机械与物理研究所,长春 130033,中国)

摘要:与通用目标检测相比,伪装目标检测(Camouflaged object detection, COD)由于目标与背景具有极高的相似度,其检测难度更大。过去,大多数COD方法基于卷积神经网络(Convolutional neural network, CNN)构建。注意到Transformer在COD任务中蕴含的巨大潜力,因此采用Swin Transformer作为骨干网络,以生成更具鲁棒性的多层次特征。考虑到伪装目标与背景之间复杂的特征关联,本文提出了一种边缘感知模块(Edge-aware module, EAM),用以生成精细的边缘先验信息。并且设计了一组特征聚合模块(Feature aggregation module, FAM)来融合多层次特征;同时,通过将边缘特征像素输入图卷积网络(Graph convolutional network, GCN)来探究边缘点之间的内部联系,以此剔除不确定的边缘点,进而引导多层次特征的精细融合操作。通过级联多个FAM来逐步减少难以区分的噪声,从而获得边缘锐利的预测图。实验结果表明,所提出的模型在单张NVIDIA 2080Ti GPU上能够以实时速度运行(每秒46帧)。在4个公开数据集上与12种最先进的COD方法进行比较,本文方法表现出了优越的性能。

关键词:伪装目标检测;Swin Transformer;图卷积网络;边缘感知学习

研究亮点:

1.建立了一种融合Swin Transformer、边缘感知模块和图卷积关系推理的伪装目标检测方法,实现了多层特征与边缘先验信息的渐进式融合,有效提升了复杂背景下伪装目标的检测精度。

2.基于图卷积的边缘关系建模能够有效增强目标边界细节恢复能力,在多个公开数据集上均取得了优异检测性能,并在息肉分割、肺部感染分割等下游任务中表现出良好的泛化能力。