Camouflaged Object Detection with Adaptive Partition and Background Retrieval

Bowen Yin[1], Xuying Zhang[1], Li Liu[2⋆], Ming-Ming Cheng[1], Yongxiang Liu[2], and Qibin Hou[1⋆]

[1] VCIP, CS, Nankai University, Tianjin, China
[2] Academy of Advanced Technology Research of Hunan, Changsha, China

Abstract. Recent works confirm the importance of local details for identifying camouflaged objects. However, how to identify the details around the target objects via background cues lacks in-depth study. In this paper, we take this into account and present a novel learning framework for camouflaged object detection, called AdaptCOD. To be specific, our method decouples the detection process into three parts, namely localization, segmentation, and retrieval. We design a context adaptive partition strategy to dynamically select a reasonable context region for local segmentation and a background retrieval module to further polish the camouflaged object boundaries. Despite the simplicity, our method enables even a simple COD model to achieve great performance. Extensive experiments show that AdaptCOD surpasses all existing state-of-the-art methods on three widely-used camouflaged object detection benchmarks. Code is publicly available at https://github.com/HVision-NKU/AdaptCOD.

Keywords: Camouflaged object detection · adaptive partition · background retrieval.

1 Introduction

Camouflaged object detection (COD), which aims to identify camouflaged objects from visually similar surroundings, is an emerging research in the computer vision community. It also facilitates a wide range of valuable real-life applications in different fields, *e.g.*, species discovery in biological research [14], surface defect detection in industry manufacture [54, 28], and polyp segmentation in medical diagnosis [12]. In COD scenes [4, 25], the target objects usually hide in complex and cluttered surroundings, which makes the model difficult to accurately locate and segment them. Besides, a number of factors, *e.g.*, extremely similar backgrounds, a wide variety of object sizes, and irregular object edges, also contribute to the difficulty of this task. Fig. 1 provides some samples of camouflaged objects.

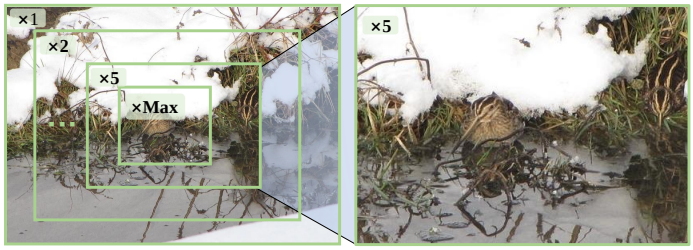
⋆ Corresponding authors.

Fig. 1. Visual comparisons between our AdaptCOD and two recent state-of-the-art methods for camouflaged object detection, *e.g.*, SegMaR [24] and DTINet [35]. As can be observed, although recent state-of-the-art methods can locate the target, they still struggle to distinguish extremely similar occluding regions and surroundings.

The dominant COD methods are based on either Transformers [35, 30, 57], or convolutional neural networks [24, 45, 76], or both, over recent years. It is noteworthy that most methods focus on the design of novel network architectures to improve segmentation performance, while the role of the context information around the camouflaged objects is rarely explored. As an exceptional case, SegMaR [24] reveals that there is no need to identify the camouflaged objects by observing the whole image. It presents a multi-stage model that leverages an iterative magnification refinement strategy to gradually refine the camouflaged objects by only looking at an attentive region. Despite its good performance, SegMaR only focuses on studying how to iteratively polish the regions closely around the targets, while neglecting to analyze how to properly leverage the context information and the background to improve the segmentation results.

Taking the aforementioned analysis into account, in this paper, we start by investigating how large the context regions should be for better identifying the camouflaged objects. As depicted in Fig. 2, we randomly select an image from the camouflaged object detection dataset, gradually increase its resolution, and crop it to a fixed size around the target center. We found that the segmentation performance initially rises as the size of the camouflaged object increases, but later degrades. This phenomenon indicates that a suitable amount of background context information is crucial for identifying the target object and how to judge the background region selection has a large impact on the performance of COD models.

Inspired by these observations, we present a novel locate-segment-polish learning framework, termed AdaptCOD, for the identification of camouflaged objects, as illustrated in Fig. 3. First, the whole image is fed to a simple network to generate a coarse prediction map of the camouflaged targets. Based on the rough prediction map, we design a context adaptive partition strategy to split the entire image into the candidate foreground region and the remaining



zoom ratio	×1	×2	×3	×5	×10	×Max
S_m	0.845	0.873	0.888	0.918	0.891	0.879
wF_m	0.741	0.779	0.798	0.851	0.812	0.795

Fig. 2. Performance of our COD model on a randomly picked image with varying zoom ratio. The model is well-trained on the standard training set and the parameters are fixed. '×5' means we increase the image size using a zoom ratio of 5 and then crop the result to a fixed size around the target center. '×Max' means the largest zoom ratio.

background region. Then, the candidate foreground region is fed into the simple network again to capture more details around the camouflaged targets. To further polish the segmentation results, we propose a background retrieval module using the background region as the query to correct the wrong predictions.

To validate the effectiveness of our AdaptCOD, we conduct extensive ablation experiments on three popular COD benchmarks (NC4K [38], COD10K [10], and CAMO [27]). Remarkably, our AdaptCOD achieves new state-of-the-art records among all these benchmarks compared to recent edge-cutting methods, *i.e.*, EINet [30] and TPRNet [72]. In particular, on the COD10K-test dataset, our method achieves 0.819 weighted F-measure [39] and 0.892 S-measure [8] scores, while the results for the second-best model TransCoop [59] are 0.779 and 0.863, respectively. Furthermore, the visualization results also show the superiority of our AdaptCOD over existing COD methods.

We summarize the main contributions of this paper as follows:

- We unveil that it is important for COD methods to make use of appropriate context information to generate high-quality segmentation results.
- We present a novel locate-segment-polish framework for COD research, termed AdaptCOD, which contains a context adaptive partition strategy and a background retrieval module.
- We show that our AdaptCOD can greatly improve the performance of previous COD methods. In addition, with the help of our AdaptCOD, even a simple network can also achieve great performance.

2 Related Work

2.1 Camouflaged Object Detection

Camouflaged object detection is a challenging task due to the subtle differences between the target and the background. To solve this issue, a series of works [44,

23, 76, 45, 71, 24, 41, 11] have been successively proposed to improve the accuracy of the segmentation results. To be specific, these methods can be divided into three categories according to the strategies they adopted. 1) Multi-scale feature aggregation: ZoomNet [45] processes the input image at three scales and unifies the scale-specific appearance features at different scales. PreyNet [71] splits the identification process for camouflaged objects into initial detection and predator learning according to the process of predation. 2) Multi-stage refinement strategy: In SINetV2 [10], the neighbor connection decoder and group-reversal attention are designed to further promote accurate segmentation of camouflaged objects. 3) Multi-task training: SLSR [38] proposes a joint-learning framework by introducing camouflaged ranking or learning from salient objects to camouflaged objects.

Recently, Transformers derived from NLP [56, 7] have achieved great success in a series of visual tasks, including classification [51, 15, 64, 5], object detection [66], segmentation [60, 31], super-resolution [77], and multi-modal learning [73, 61]. They also inspired a lot of work to apply this architecture to the COD task. UGTR [65] utilizes a probabilistic representational model to learn the uncertainties of the camouflaged objects using the Transformer framework. DTINet [35] proposes a dual-task interactive Transformer that can segment both the camouflaged objects and their detailed borders. EINet [30] designs a decoder with neighbor and hop connections to progressively refine the camouflaged objects. Most of these methods focus on the design of novel network architectures. Despite the good performance, they mostly ignore the importance of the context information around camouflaged objects.

2.2 Magnification Strategy

To acquire fine-grained foreground predictions for camouflaged objects, it is necessary to perceive enough details around the target [26, 53, 78, 3]. Recently, a representative type of work has achieved this by sampling several sub-regions at finer scales from the whole image. For instance, TASN [75] designs an attention-based sampler to increase the proportion of the attended regions while maintaining the image size to the original scale. Later, PA-KRN [63] uses this sampler method to highlight foreground object regions with a higher resolution based on body-attention maps. More recently, SegMaR [24] iteratively magnifies the regions around the targets according to the attention map for accurate segmentation of the camouflaged targets, especially for those small ones. These works bring positive gains for accurate localization and segmentation of objects. Nevertheless, they only rely on the attention map to amplify the objects, ignoring the use of context and background information. Different from them, our framework adaptively partitions the image regions according to the perceived global context, which is able to achieve better performance at lower computations.

2.3 Discriminative Content Retrieval

The core of segmenting camouflaged objects with fine details is to retrieve the discrepant positions in a confusing scene [74, 32, 49, 37, 70]. A large number of works [17, 13, 43, 69] have been devoted to exploiting more discriminative context to improve the identification ability of their methods. For example, PFNet [41] develops a distraction mining strategy to separately process the features of the foreground and background to remove false predictions. DDS [34] proposes diverse deep supervisions to enhance the awareness of edges. BGNet [53] and BSANet [78] utilize boundary features to segment camouflaged objects more finely. OPNet [42] proposes a dual-focus module to integrate local and global representations for accurate positioning of the camouflaged objects. These methods implicitly exploit the discriminative contexts from the whole image and still struggle to capture the details of camouflaged objects. On the contrary, in this paper, we use background representations as discriminative cues to explicitly polish the target-around background regions, which can achieve better segmentation results.

3 Proposed Framework

In this section, we first briefly introduce the overview of the proposed AdaptCOD and then sequentially describe the core components of AdaptCOD, namely global localization, local segmentation, and background retrieval. Finally, we give the training loss functions used in this paper.

3.1 Overview

The overall architecture of our proposed AdaptCOD is illustrated in Fig. 3. The entire pipeline can be separated into three parts: global localization, local segmentation, and background retrieval. The first localization part aims to roughly locate the camouflaged objects via global context modeling. The input image is sent to the shared network to obtain a coarse prediction, as shown on the top left of Fig. 3. Based on the coarse prediction, we use an adaptive partition strategy to generate a partition mask, which separates the image into a foreground region and a background region. In the segmentation part, we split the foreground region into fine-grained patches and utilize the shared network to extract features with fine details for accurate segmentation. The retrieval part takes the partition mask and the feature maps from the first two parts as input. We utilize our proposed background retrieval module to refine local features by measuring the distance between local features and background features and then use the refined features to generate the final prediction.

3.2 Global Localization

In the global localization part, we feed an input image with a spatial size of $H \times W$ into a shared network built on an encoder-decoder architecture. The

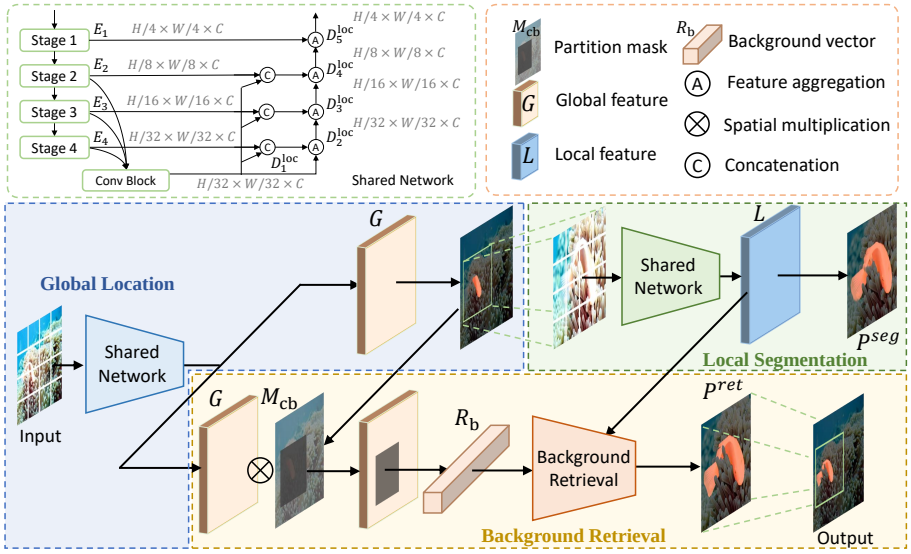


Fig. 3. Overall architecture of our AdaptCOD framework. First, we feed the whole image into a shared network for coarse prediction to locate the target. The shared network adopts pretrained PVTv2-b4 [58] as the backbone, as same as HitNet [19] and EINet [30]. Then, we adaptively partition the image into the background and candidate foreground regions and process the latter region by feeding it into the shared network for more local details. Finally, we use the background retrieval strategy to further enhance the discrimination capacity of the model. 'Feature aggregation' is illustrated in Eqn. 3.

adopted decoder structure is built on FPN [33] and is similar to the popular camouflaged object detection methods [10, 67], but only composed of basic convolutional layers. Specifically, multi-scale visual features are extracted from the encoder with four stages. We reduce the channel number of these features to $C = 128$ with 1×1 convolutions to achieve a better trade-off between efficiency and performance. The resulting features $\{E_i\}_{i=1}^{4}$ with spatial size $\frac{H}{2^{1+i}} \times \frac{W}{2^{1+i}}$ are sent to the decoder to generate segmentation maps. In particular, the features from the last three stages are sent to a convolutional block, where they are fused together and processed by two convolutional layers to produce the features D_1^{loc} that contain a more comprehensive understanding of the image. This process can be defined as follows:

$$D_1^{loc} = \mathcal{F}_{\text{conv5}\times5}(\mathcal{F}_{\text{conv3}\times3}([\mathcal{F}_{\text{resize}}(E_2), \mathcal{F}_{\text{resize}}(E_3), E_4)), \tag{1}$$

where the $\mathcal{F}_{\text{resize}}$ is the bilinear interpolation function for resolution matching with E_4 and $\mathcal{F}_{\text{convk}\times\text{k}}$ means the convolution operation with kernel size $k \times k$. The fused features $D_1^{loc} \in \mathbb{R}^{H/32 \times W/32 \times C}$ are used to enhance the features $\{E_i\}_{i=2}^{4}$ as follows:

$$E_i^{loc} = \mathcal{F}_{\text{conv3}\times3}([\mathcal{F}_{\text{resize}}(D_1^{loc}), E_i]), i \in \{2, 3, 4\}\,, \tag{2}$$

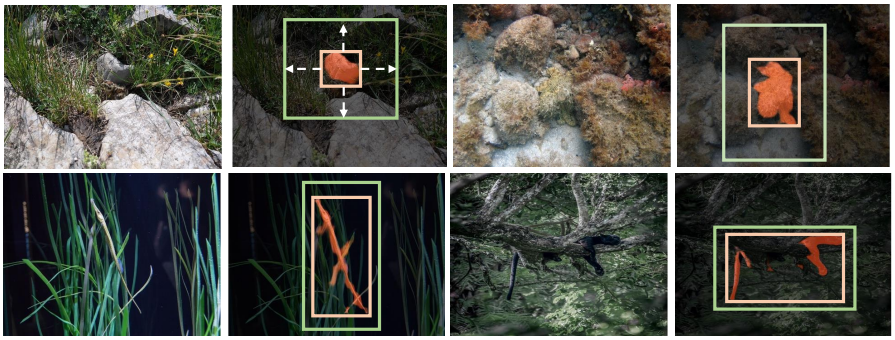


Fig. 4. Illustration of our context adaptive expansion strategy. The inner rectangle represents the bounding box of the predicted target, while the outer one is expanded from the inner one through the context adaptive expansion strategy.

where E_i^{loc} are the enriched visual features and $E_1^{loc} = E_1$. Following FPN [33], we also use a top-down manner to decode the visual features across multiple stages as shown in Fig. 3. The feature aggregation procedures in the top-down structure are formulated as follows:

$$D_j^{loc} = \mathcal{F}_{\text{conv3}\times3}(\mathcal{F}_{\text{resize}}(D_{j-1}^{loc}) \cdot E_{6-j}^{loc} + E_{6-j}^{loc}), j \in [2, 5]. \tag{3}$$

The decoder features D_i^{loc} can be converted to predictions P_i^{loc} via a 3×3 convolution followed by a bilinear interpolation operation.

Considering that the last decoder features D_5^{loc} contain rich global context of the whole image, we adopt them as the global features, denoted as G. Meanwhile, the corresponding prediction P_5^{loc} is also utilized in the local segmentation part.

3.3 Local Segmentation

As demonstrated in Sec. 1, how to adjust the context region around the camouflaged objects has a great impact on the model performance. Here, we present an adaptive partition strategy that can automatically select a reasonable context region for local segmentation.

Given the coarse prediction $P_5^{loc} \in [0, 1]^{H \times W}$, we can easily attain the bounding box enclosing the predicted targets. We denote the height and width of the bounding box as h and w, respectively. We can get a new rectangle region of height h_e and width w_e formulated as follows:

$$\begin{cases} h_e &= h \cdot (1 + \gamma \times r_t), \\ w_e &= w \cdot (1 + \gamma \times r_t), \\ r_t &= \sum P_5^{loc}/(h \times w), \end{cases} \tag{4}$$

where γ is a fixed scaling parameter controlling the magnitude of the context adaptive expansion, and r_t refers to the ratio of the predicted foreground area

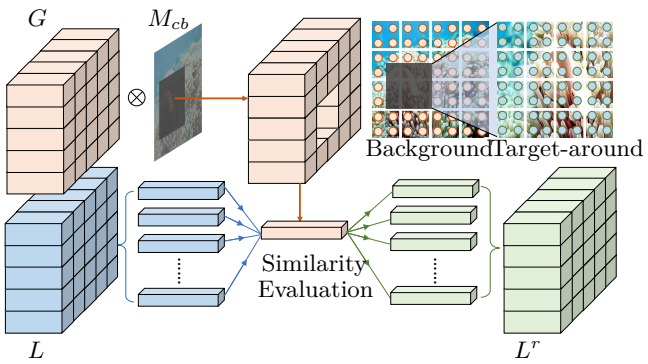


Fig. 5. Illustration of the background retrieval module. We first use the partition mask to exclude the foreground-around clues in the global feature to obtain pure background information. Then, we retrieve background points in the local feature by measuring the similarity between it and the background representation.

to the initial bounding box. This strategy can be explained as follows. Given a bounding box with a fixed size, when the proportion of the target objects r_t gets larger, the region of the background context tends to be smaller. In this case, more background regions outside the bounding box should be used for accurate segmentation.

To provide a more comprehensible explanation, some expanded examples are shown in Fig. 4. As can be seen in the first row, the background area in the initial bounding box is small; thus our strategy expands it at a relatively large rate. The situations in the second row are just the opposite. Therefore, our context adaptive partition strategy can dynamically expand the bounding box enclosing the target objects to a proper one.

Based on the expanded box, we split the entire image into a candidate foreground R_{cf} corresponding to the region inside the box and a background region R_{cb} that is outside the box. The background mask $M_{cb} \in \{0,1\}^{H\times W}$ can be easily attained as follows:

$$M_{cb}(i,j) = \begin{cases} 0, (i,j) \in R_{cf} \\ 1, (i,j) \in R_{cb}, \end{cases} \tag{5}$$

which will be used in background retrieval, illustrated in the next subsection.

For the remaining foreground region, we crop it from the original image as a local image and reshape it to the spatial size $H \times W$. Then, we send the local image to the shared network to generate segmentation results $\{P_i^{seg}\}_{i=1}^{5}$ with fine details. Following the same principle as global localization, we choose the decoder features in the last stage, denoted as D_5^{seg}, as the local features L with shape $H/4 \times W/4 \times C$.

3.4 Background Retrieval

Due to the subtle differences between the camouflaged objects and the background context, clearly recognizing the boundaries of the camouflaged objects is still challenging even though the local segmentation part is used. We observe that some background regions are still falsely predicted as foreground, degrading the model performance. To further polish the regions around the object boundaries, we propose to distinguish the camouflaged targets from their close background surroundings through a background retrieval module.

The basic idea is depicted in Fig. 5. In this module, we first use the global features $G \in \mathbb{R}^{H/4\times W/4\times C}$ and the background mask $M_{cb} \in \{0,1\}^{H\times W}$ to generate an indexing vector $R^b \in \mathbb{R}^{1\times C}$ via a masked average pooling (MAP) function. This process is formulated as follows:

$$R^b = \frac{\sum(\mathcal{F}_{\text{resize}}(M_{cb}) \odot G)}{\sum \mathcal{F}_{\text{resize}}(M_{cb})}, \tag{6}$$

where $\odot$ represents the element-wise multiplication operation and $\sum(\cdot)$ accumulates the feature values along the spatial dimension.

Given the indexing vector R^b, we use it to retrieve the background regions from the local features. This can be simply done by measuring the cosine similarity distance between R^b and each position in $L \in \mathbb{R}^{H/4\times W/4\times C}$. Then, the resulting feature $L^r \in \mathbb{R}^{H/4\times W/4\times C}$ can be attained by:

$$L^r(i,j) = L(i,j)(1 - \frac{R^b L^T(i,j)}{\|R^b\|\|L(i,j)\|})/2, \tag{7}$$

where $\|\cdot\|$ represents L_2 norm function. Finally, we obtain the fine-level prediction P^{ret} with the same spatial shape as R_{cf} from L^r via a 3×3 convolution followed by a bilinear interpolation function and paste it back as the final prediction to its original location, as illustrated in Fig. 3.

3.5 Training Objectives

The complete training process spans over 75 epochs. In particular, we first train AdaptCOD on the standard COD training set for localization. Following [18, 62], we add side supervision at each feature level. Given the predictions generated by the localization part $P^{loc} = \left\{P_i^{loc}\right\}_{i=1}^{5}$ and the corresponding ground truth T, the training loss for the first 60 epochs can be defined as:

$$\mathcal{L}(P^{loc}, T) = \sum_{i=1}^{5} \mathcal{L}_{bce}(P_i^{loc}, T) + \mathcal{L}_{iou}(P_i^{loc}, T), \tag{8}$$

where $\mathcal{L}_{bce}$ and $\mathcal{L}_{iou}$ represent the BCE loss [6] and the IoU loss [40], respectively.

Afterward, we jointly train the whole framework. Given the prediction maps from the three parts $\left\{P_i^{loc}\right\}_{i=1}^{5}$, $\{P_i^{seg}\}_{i=1}^{5}$, and P^{ret}, the training loss for the last 15 epochs can be written as:

$$\mathcal{L} = \mathcal{L}(P^{loc}, T) + \mathcal{L}(P^{seg}, T_l) + \mathcal{L}(P^{ret}, T_l), \tag{9}$$

where T_l denotes the ground truth within the expanded bounding box.

4 Experiments

4.1 Experiment Setup

Datasets. We evaluate the proposed AdaptCOD method on the three most popular benchmark datasets in COD research, *i.e.*, CAMO [27], COD10K [10], and NC4k [38]. The CAMO dataset is composed of 2500 samples, where 1,250 images contain camouflaged objects and the remaining 1,250 images do not. The COD10k dataset comprises 5,066 images containing camouflaged objects, 3,000 background images, and 1,934 non-camouflaged images. NC4K is a large-scale dataset with 4,121 images, and all of them contain camouflaged objects. Following prior works [11, 24, 45], we utilize 1,000 images from CAMO and 3,040 images from COD10K for training and leave the rest images for test.

Metrics. Following the protocols in prior studies [45, 24, 76], we adopt four widely used metrics to perform evaluations for the predictions, including structure measure (S_m) [8], mean absolute error (M) [47], weighted F-measure (wF) [39], and adaptive E-measure (αE) [9]. To be specific, the mean absolute error represents the absolute disparity between the prediction map and GT. The structure-measure assesses the region-aware and the object-aware structural resemblance between predictions and GT. The weighted F-measure provides an all-encompassing measure of recall and precision. The adaptive E-measure evaluates element-wise similarity and image-level statistics. In addition, we also plot precision-recall (PR) curves and F_β-threshold (F_β) curves to further evaluate the performance of the COD models.

Hyperparameter Details. Following HitNet [19, 30], we employ PVTv2-b4 [58] pre-trained on the ImageNet dataset as the backbone, and we set γ in our adaptive expansion strategy to 0.4. During the training phase, we set the batch size to 16, utilize the SGD optimizer, and initialize the learning rate to 1e-2, which decreases gradually via the cosine annealing method [36]. In the inference phase, the whole images are first resized to 384×384 and fed into the shared network in AdaptCOD for the localization of camouflaged objects. Then, in the segmentation part, the context-reasonable region is cropped from the original image and resized to 384×384. This local image is fed to the shared network again to generate local features. Finally, these features are refined by eliminating the false predictions via background retrieval. All experiments in this paper are conducted with the PyTorch library [46] on an NVIDIA GeForce RTX 3090 GPU.

Computational Cost and Speed. As mentioned above, the total number of training epochs is set to 75, with 60 for initial localization and 15 for further refinement. The entire process takes about 5.8 hours, with 3.3 hours dedicated to training and 2.5 hours to fine-tuning. Our AdaptCOD contains 66.4M parameters and 67.5G Macs, and the inference speed is 30.4 FPS. Fig. 10 presents the comparisons of computational cost and speed between our method and others.

4.2 Ablative Studies

Importance of Each Component. We first analyze the influence of the three components. As shown in Tab. 1, the experimental results indicate that both the

Fig. 6. Visual results of the predictions with different components used in our Adapt-COD. The visualization samples are randomly picked from the NC4K and COD10K datasets.

Setting	Parts		NC4K (4,121)				COD10K-Test (2,026)				CAMO-Test (250)			
	LS	BR	$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	$M \downarrow$	$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	$M \downarrow$	$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	$M \downarrow$
1			0.870	0.920	0.810	0.038	0.843	0.909	0.787	0.029	0.839	0.901	0.791	0.059
2	✓		0.885	0.930	0.839	0.035	0.860	0.924	0.799	0.025	0.866	0.926	0.827	0.048
3		✓	0.892	0.933	0.852	0.032	0.887	0.929	0.815	0.024	0.877	0.927	0.840	0.046
4	✓	✓	**0.906**	**0.942**	**0.860**	**0.029**	**0.892**	**0.938**	**0.819**	**0.021**	**0.886**	**0.932**	**0.844**	**0.043**

Table 1. Ablation study on the proposed three-part framework. All the parts contribute to the overall performance. 'LS': local segmentation part; 'BR': background retrieval part.

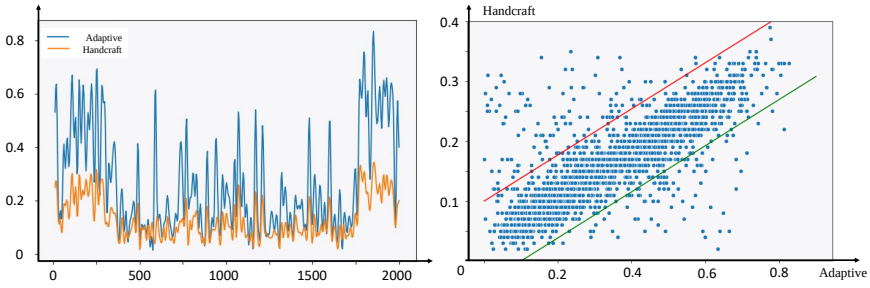
Settings	Expansion Manner	NC4K (4,121)				COD10K-Test (2,026)				CAMO-Test (250)			
		$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	$M \downarrow$	$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	$M \downarrow$	$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	$M \downarrow$
1	Fixed $\alpha = 0$	0.883	0.924	0.825	0.037	0.851	0.897	0.734	0.028	0.862	0.906	0.812	0.052
2	Fixed $\alpha = 1/4$	0.890	0.933	0.845	0.031	0.882	0.930	0.811	0.026	0.871	0.916	0.830	0.048
3	Fixed $\alpha = 1/2$	0.889	0.927	0.828	0.034	0.863	0.896	0.733	0.027	0.870	0.909	0.823	0.051
4	Fixed $\alpha = 1$	0.878	0.925	0.829	0.035	0.850	0.891	0.730	0.029	0.866	0.901	0.816	0.055
5	Context Adaptive	**0.906**	**0.942**	**0.860**	**0.029**	**0.892**	**0.938**	**0.819**	**0.021**	**0.886**	**0.932**	**0.844**	**0.043**

Table 2. Ablation study on the expansion manner for the bounding box of the target. 'Fixed': fixed the expansion rate for the bounding box; 'Adaptive': expand the box via our context adaptive expansion strategy.

segmentation and the retrieval parts are conducive to the performance improvement of AdaptCOD. In particular, equipped with our background retrieval strategy, the segmentation phase improves the S_m and wF metrics by 2.1% and 3.2% on average on all the benchmarks, respectively. Furthermore, the combination of all three components can further improve the performance, which demonstrates the importance of our local segmentation and background retrieval. The corresponding visual results are shown in Fig. 6. As can be observed, the predictions from the localization component can roughly locate the target objects. The utilization of the segmentation component can further improve the segmentation results but fails to process some details near the boundaries of the camouflaged objects. In the retrieval component, the potential background pixels are discriminated from the visual features, yielding more accurate predictions.

Effectiveness of Context Adaptive Expansion. To demonstrate the effectiveness of the proposed expansion strategy, we fine-tune AdaptCOD with different expansion rates for comparison. Given the fixed expansion rate α, the box size can be expanded to $(h + 2\alpha h, w + 2\alpha w)$. The results of AdaptCOD with different expansion strategies on all the benchmarks are shown in Tab. 2. Remarkably, our context adaptive partition strategy achieves the best performance, and the strategy with a fixed expansion rate of $\alpha = 1/4$ is the second best. These results indicate that too much or too little background context can disturb the accuracy of the segmentation predictions. And the rationality of our strategy of expanding the bounding box also gets verified.

More Analysis on the Expansion Strategies. To further confirm the effectiveness of our context adaptive expansion strategy for camouflaged object detection, we compare our adaptive expansion rate with the best expansion rate



(a) expansion rate curves (b) expansion rate scatter

Fig. 7. The curves and scatter map of handcraft expansion rate and our adaptive rate. (a) Comparison of expansion rate curves between ideal value and our adaptive value on the COD10K test set. 'Handcraft': the best expansion rate for the specific image; 'Adaptive': our context adaptive expansion rate r_t. (b) Scatter map with the handcraft rate axis and adaptive expansion rate axis.

Sample	Computation			COD10K (2,026)			NC4K (4,121)		
Strategy	Param (M)	Macs (G)	FPS	$S_m \uparrow$	$wF \uparrow$	M $\downarrow$	$S_m \uparrow$	$wF \uparrow$	M $\downarrow$
attention-based sampler	132.8	67.5	19.8	0.886	0.808	0.024	0.897	0.851	**0.029**
Adaptive partition (Ours)	66.4	67.5	30.4	**0.892**	**0.819**	**0.021**	**0.906**	**0.860**	**0.029**

Table 3. Comparison between our adaptive partition strategy and the attention-based sampler.

on different samples. In particular, we test our AdaptCOD on each image in the COD10K test set with equidistant sampled $\alpha = 0, 0.01, 0.02, \cdots, 0.99, 1.00$, and choose α with the highest wF as its best expansion rate. Fig. 7(a) illustrates the smoothed line graph of our adaptive expansion rate and the ideal expansion rate for the samples. As can be observed, the curves of the best expansion rate and our adaptive rate present a positive correlation. Therefore, our context adaptive expansion strategy can even become an ideal expansion strategy if it is properly scaled. Furthermore, the curves can also explain the reason why the fixed $\alpha = 1/4$ in Tab. 2 can achieve the second best performance is that $1/4$ is closest to the best rate curve. Fig. 7(b) shows the hand-craft and adaptive expansion rates on the y-axis and x-axis, respectively. Particularly, the least-squares method is employed to fit the relationship between the two rates and denote the points with vertical errors less than 0.05 between two diagonal lines. It is clear that over 95% of the points lie between the two lines, which indicates there is a positive correlation between our adaptive expansion rate and the optimal expansion rate. These results further provide insights into the effectiveness of our approach.

Comparison with SegMaR. Here, we compare our adaptive partition sampler with the attention-based sampler used in SegMaR [24]. Firstly, attention-based samplers adopt anisotropy sampling, which may cause image distortion. These samplers are originally designed for image classification, and this task does not harm the category judgment. For COD, a dense prediction task, the predictions

Methods	Single Large (579)			Single Small (1,259)			Multi Objects (188)			Multi Small (119)		
	$S_m\uparrow$	$wF\uparrow$	$M\downarrow$	$S_m\uparrow$	$wF\uparrow$	$M\downarrow$	$S_m\uparrow$	$wF\uparrow$	$M\downarrow$	$S_m\uparrow$	$wF\uparrow$	$M\downarrow$
AdaptCOD w/o BR	0.909	0.891	0.019	0.847	0.749	0.026	0.816	0.723	0.031	0.790	0.653	0.033
AdaptCOD	**0.923**	**0.901**	**0.015**	**0.888**	**0.798**	**0.017**	**0.827**	**0.731**	**0.026**	**0.796**	**0.661**	**0.030**

Table 4. Effectiveness of our background retrieval module for different situations. 'BR' represent the background retrieval module.

		NC4K (4,121)				COD10K-Test (2,026)				CAMO-Test (250)			
Settings	Selection of γ	$S_m\uparrow$	$\alpha E\uparrow$	$wF\uparrow$	$M\downarrow$	$S_m\uparrow$	$\alpha E\uparrow$	$wF\uparrow$	$M\downarrow$	$S_m\uparrow$	$\alpha E\uparrow$	$wF\uparrow$	$M\downarrow$
1	$\gamma = 0$	0.883	0.924	0.825	0.037	0.851	0.897	0.734	0.028	0.862	0.906	0.812	0.052
2	$\gamma = 0.2$	0.892	0.934	0.833	0.034	0.872	0.917	0.799	0.028	0.873	0.916	0.825	0.048
3	$\gamma = 0.3$	0.899	0.937	0.846	0.033	0.880	0.925	0.807	0.024	0.874	0.920	0.831	0.046
4	$\gamma = 0.4$	**0.906**	**0.942**	**0.860**	**0.029**	0.892	0.938	**0.819**	**0.021**	**0.886**	**0.932**	**0.844**	**0.043**
5	$\gamma = 0.5$	0.904	0.941	0.857	0.030	**0.893**	**0.939**	0.818	0.022	0.883	0.930	**0.844**	**0.043**
6	$\gamma = 0.6$	0.900	0.938	0.850	0.031	0.884	0.930	0.810	0.022	0.870	0.919	0.829	0.045
7	$\gamma = 0.8$	0.895	0.933	0.844	0.033	0.878	0.922	0.805	0.025	0.868	0.913	0.821	0.047

Table 5. AdaptCOD with different hyperparameter γ.

of the sampled images require to be restored to match the original images, which may cause hollow results. In addition, such a sampling method can also pose obvious domain gaps between the original and sampled images, and hence the model in each part should be preserved. Our sampler only requires a model and is more suitable than the attention-based sampler for COD. As shown in Tab. 3, our adaptive partition strategy can achieve better performance using half parameters and higher speed.

Discussion on Background Retrieval for Different Situations. We split the COD10K test set into four subsets: SL (single large objects), SS (single small objects), M (multiple objects), and MS (multiple small objects), which respectively contain 579, 1,259, 188, 119 samples. Tab. 4 shows that our background retrieval process consistently enhances performance across all four scenarios, but the effectiveness of our retrieval module in different situations is different. For scenes with a single small object, our retrieval module can bring the largest improvement, while for scenes with multiple small camouflaged targets, the improvement is relatively low.

Selection of the Hyperparameter γ. In Eqn. 4, γ is a fixed scaling parameter controlling the magnitude of the context adaptive expansion. For the camouflaged objects in most scenes, we empirically let the enlarged box approach $2\times$ area to the box enclosing the coarsely predicted targets by setting γ to $\sqrt{2} - 1 \approx 0.4$. We further conduct experiments to validate the rationality of this hyperparameter selection. Here, we investigate how the model performance would change when adjusting the hyperparameter γ. We select 7 different values around 0.4, *i.e.,* $\{0, 0.2, 0.3, 0.4, 0.5, 0.6, 0.8\}$, and show the results in Tab. 5. The model with γ =0.4 or 0.5 achieves the best performance and increasing or decreasing γ brings negative impacts. These results illustrate that an appropriate

Fig. 8. Failure samples. Completely incorrect initial prediction of the target at the localization part is the main reason for the failure cases.

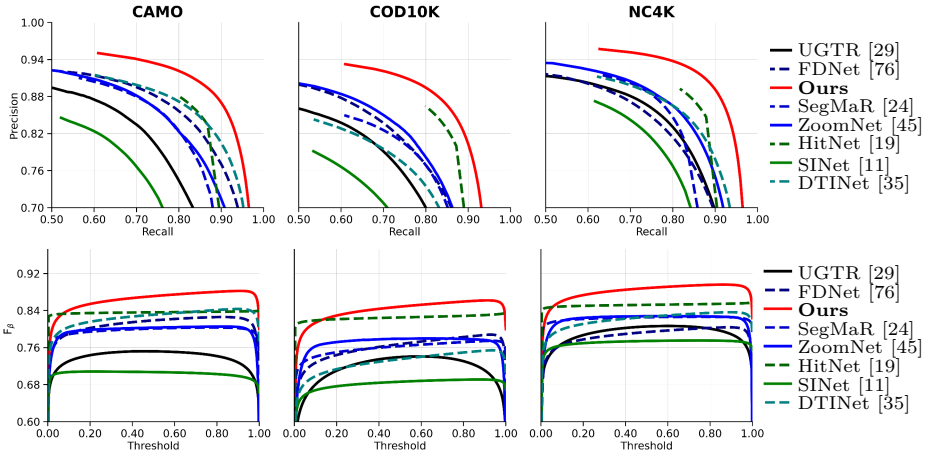


Fig. 9. PR and F_β curves of the proposed AdaptCOD and the recent SOTA algorithms on three COD datasets.

size of background context benefits the model. We recommend the users to set γ to 0.4.

Failure Samples. Our AdaptCOD aims to refine coarse segmentation results through the background retrieval strategy. As shown in Fig. 8, for some scenarios where the initial predictions have wrong regions or miss some targets, our method may also fail. However, we believe our method is still important as the improvement over the recent state-of-the-art methods are still notable as shown in Tab. 6.

4.3 Comparison with Other Methods

Comparison with State-of-the-art Methods. We compare our AdaptCOD with the recent 22 state-of-the-art methods, including OPNet [42], FEDER [16],

Method	NC4K (4,121)				COD10K-Test (2,026)				CAMO-Test (250)			
	$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	M↓	$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	M↓	$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	M↓
PraNet$_{2020}$ [12]	0.822	0.871	0.724	0.059	0.789	0.839	0.629	0.045	0.769	0.833	0.663	0.094
SINet$_{2020}$ [11]	0.808	0.883	0.723	0.058	0.776	0.867	0.631	0.043	0.745	0.825	0.644	0.092
SLSR$_{2021}$ [38]	0.840	0.902	0.766	0.048	0.804	0.882	0.673	0.037	0.787	0.855	0.696	0.080
MGL-R$_{2021}$ [68]	0.833	0.893	0.739	0.053	0.814	0.865	0.666	0.035	0.782	0.847	0.695	0.085
PFNet$_{2021}$ [41]	0.829	0.892	0.745	0.053	0.800	0.868	0.660	0.040	0.782	0.852	0.695	0.085
UJSC$_{2021}$ [29]	0.842	0.907	0.771	0.047	0.809	0.891	0.684	0.035	0.800	0.853	0.728	0.073
C^{2}FNet$_{2021}$ [52]	0.838	0.898	0.762	0.049	0.813	0.886	0.686	0.036	0.796	0.864	0.719	0.080
COS-T$_{2021}$ [57]	0.825	0.881	0.730	0.055	0.790	0.901	0.693	0.035	0.813	0.896	0.776	0.060
UGTR$_{2021}$ [65]	0.839	0.886	0.746	0.052	0.817	0.850	0.666	0.036	0.784	0.859	0.794	0.086
SINetV2$_{2022}$ [10]	0.847	0.898	0.770	0.048	0.815	0.863	0.680	0.037	0.820	0.875	0.743	0.070
DGNet$_{2022}$ [22]	0.857	0.907	0.784	0.042	0.822	0.877	0.693	0.033	0.839	0.901	0.769	0.057
SegMaR$_{2022}$ [24]	0.841	0.905	0.781	0.046	0.833	0.895	0.724	0.033	0.815	0.872	0.742	0.071
ZoomNet$_{2022}$ [45]	0.853	0.907	0.784	0.043	0.838	0.893	0.729	0.029	0.820	0.883	0.752	0.066
FDNet$_{2022}$ [76]	0.834	0.895	0.750	0.052	0.837	0.897	0.731	0.030	0.844	0.903	0.778	0.062
TPRNet$_{2022}$ [72]	0.854	0.903	0.790	0.047	0.829	0.892	0.725	0.034	0.814	0.870	0.781	0.076
DTINet$_{2022}$ [35]	0.863	0.915	0.792	0.041	0.824	0.893	0.695	0.034	0.857	0.912	0.796	0.050
EINet$_{2022}$ [30]	0.880	0.931	0.826	0.035	0.849	0.918	0.746	0.027	0.870	0.925	0.822	0.048
HitNet$_{2022}$ [19]	0.870	0.921	0.825	0.039	0.868	0.932	0.798	0.024	0.844	0.902	0.801	0.057
TransCoop$_{2022}$ [59]	0.883	0.927	0.837	0.033	0.863	0.919	0.779	0.024	0.870	0.923	0.832	0.047
FEDER$_{2023}$ [16]	0.859	0.915	0.832	0.041	0.837	0.913	0.756	0.028	0.827	0.893	0.801	0.064
FSPNet$_{2023}$ [20]	0.879	0.915	0.816	0.035	0.851	0.895	0.735	0.026	0.856	0.899	0.799	0.064
OPNet$_{2023}$ [42]	0.883	0.932	0.838	0.034	0.857	0.919	0.767	0.026	0.858	0.915	0.817	0.050
AdaptCOD (Ours)	**0.906**	**0.942**	**0.860**	**0.029**	**0.892**	**0.938**	**0.819**	**0.021**	**0.886**	**0.932**	**0.844**	**0.043**

Table 6. Comparisons of our AdaptCOD with the recent state-of-the-art methods. As can be seen, our AdaptCOD outperforms previous methods by a large margin. '↑': the higher the better, '↓': the lower the better.

FSPNet [20], ZoomNet [45], FDNet [76], SegMaR [24], DGNet [22], SINetV2 [10], C^{2}FNet [52], TPRNet [72], UJSC [29], PFNet [41], MGL-R [68], SLSR [38], SINet [11], PraNet [12], COS-T [57], HitNet [19], EINet [30], DTINet [35], TransCoop [59] and UGTR [65]. To make the comparison fair, the prediction maps are directly provided by their authors or generated by their well-trained models, and they are evaluated by the same code. As shown in Tab. 6, our AdaptCOD outperforms other methods on all metrics without any extra training data or post-processing tricks. Particularly, it outperforms the second-best method TransCoop [59] by an average of 2.4% in terms of S_m and wF. Compared to the recent amplification-involved methods SegMaR [24] and ZoomNet [45], our method also shows obvious performance improvement on all datasets.

PR and F$_\beta$ curves. We also provide the PR and F$_\beta$ curves of the proposed AdaptCOD and previous cutting-edge methods. As shown in Fig. 9, the solid red lines belonging to our AdaptCOD methods are higher than the ones that belong to other methods by a large margin, which further demonstrates the effectiveness of the adaptive partition strategy and background retrieval module in our AdaptCOD.

Performance-computation Trade-off. Based on the most popular benchmark dataset of COD, *i.e.*, COD10K, we present the relationship between the performance and efficiency of AdaptCOD and other methods. As shown in Fig. 10, our AdaptCOD greatly surpasses previous state-of-the-art methods at similar computation cost.

Qualitative Evaluation. Fig. 11 illustrates the visual comparisons of our AdaptCOD and recent state-of-the-art methods on challenging scenes. The selected samples can be separated into two parts. For the first part where the

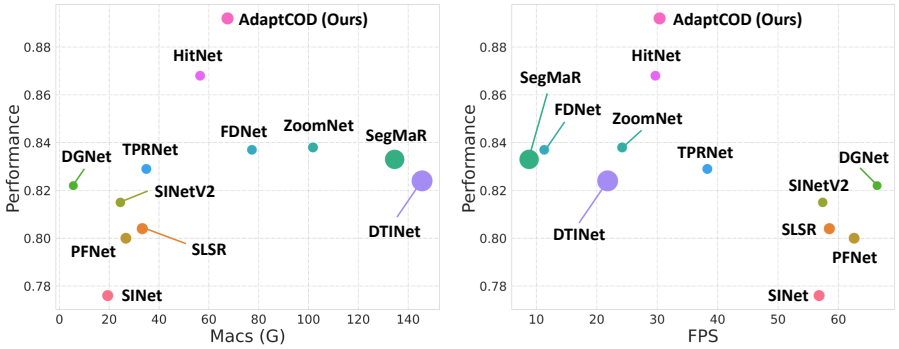


Fig. 10. Relationship between model performance and efficiency. We report the S-measure performance of different methods on COD10k. The larger the colored scatter point size, the heavier the model parameters.

Method	NC4K (4,121)				COD10K-Test (2,026)				CAMO-Test (250)			
	$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	M$\downarrow$	$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	M$\downarrow$	$S_m \uparrow$	$\alpha E \uparrow$	$wF \uparrow$	M$\downarrow$
ZoomNet$_{2022}$ [45]	0.853	0.907	0.784	0.043	0.838	0.893	0.729	0.029	0.820	0.883	0.752	0.066
w/ Seg part (Ours)	0.858	0.912	0.795	0.040	0.845	0.897	0.738	0.028	0.824	0.886	0.755	0.065
w/ AdaptCOD (Ours)	**0.868**	**0.920**	**0.801**	**0.038**	**0.851**	**0.903**	**0.745**	**0.026**	**0.832**	**0.894**	**0.769**	**0.063**
HitNet$_{2022}$ [19]	0.870	0.921	0.825	0.039	0.868	0.932	0.798	0.024	0.844	0.902	0.801	0.057
w/ Seg part (Ours)	0.876	0.925	0.836	0.037	0.873	0.937	0.805	0.023	0.849	0.905	0.811	0.053
w/ AdaptCOD (Ours)	**0.884**	**0.930**	**0.841**	**0.035**	**0.880**	**0.940**	**0.812**	**0.022**	**0.860**	**0.912**	**0.824**	**0.050**

Table 7. Application of our proposed framework AdaptCOD to state-of-the-art COD model ZoomNet [45] and HitNet [19]. 'w/ Seg part': the model equipped with the segmentation part in AdaptCOD. 'w/ AdaptCOD': the model equipped with the segmentation part and background retrieval part in AdaptCOD

images contain small camouflaged objects, *e.g.*, 6^{th}, 10^{th}, and 12^{th} rows, it is challenging to identify them. In terms of the other part, the patterns of the camouflaged objects are extremely similar to their surrounding context, *e.g.*, 1^{st} and 2^{nd} rows, which will inevitably cause confusion. As can be seen, other state-of-the-art methods exhibit obvious missing, misjudgment, and ambiguity in the segmentation results, while our AdaptCOD achieves accurate segmentation thanks to the context adaptive partition strategy and the background retrieval module. Additionally, it is apparent that our predictions have clearer and more complete object regions compared to other methods. These visual results intuitively show the superior performance of the proposed method over other methods.

4.4 Application to Other Models

Considering that the proposed AdaptCOD is a generic scheme for COD research, we also attempt to apply this scheme to other COD models. The only thing we should do is fine-tuning the chosen models from well-trained parameters. In our experiments, we replace the shared model in our framework with the

Fig. 11. Qualitative comparisons on challenging samples in the testing set of COD10K.

Method	CVC300 [2]		CVC-ClinicDB [1]		Kvasir [21]		ColonDB [55]		ETIS [50]	
	DICE↑	mIoU↑	DICE↑	mIoU↑	DICE↑	mIoU↑	DICE↑	mIoU↑	DICE ↑	mIoU↑
CASCADE$_{2023}$ [48]	90.47	83.79	94.34	89.98	92.58	87.76	82.54	**74.53**	80.07	72.58
w / Seg part (Ours)	90.71	83.88	94.79	90.55	92.87	87.80	**82.58**	74.50	82.09	73.55
w / AdaptCOD (Ours)	**91.05**	**83.98**	**95.16**	**90.95**	**92.90**	**87.86**	82.56	74.42	**83.49**	**75.26**

Table 8. Application of our AdaptCOD to state-of-the-art polyp segmentation model CASCADE [48]. 'w / Seg part': CASCADE equipped with the segmentation part in AdaptCOD.

recent state-of-the-art model ZoomNet [45] and HitNet [19]. As shown in Tab. 7, the introduction of AdaptCOD brings significant and consistent performance improvements on all the benchmarks for the basic ZoomNet and HitNet. These results validate the strong generalization of our work to other alternative COD methods.

Our AdaptCOD method is also applicable to polyp segmentation in medical imaging, where early detection of colorectal cancer is challenging due to the low-contrast boundaries between the polyps and their highly similar surroundings during colonoscopy diagnosis. As reported in Tab. 8, the application of AdaptCOD on the polyp segmentation method yields better performance on all datasets. These experimental results further indicate the generalization ability of our AdaptCOD.

5 Conclusions and Future Works

Conclusions. In this paper, we propose a novel learning scheme for camouflaged object detection, termed AdaptCOD, which splits the prediction process into three parts, namely localization, segmentation, and retrieval. We design a context adaptive partition strategy to dynamically select proper context regions around the target objects. A background retrieval module is also developed to further polish the predicted camouflaged objects. Experiments show the effectiveness of the proposed method. We also show that existing COD methods can also benefit from the proposed method, reflecting its generalization ability.

Future Works. Despite the great performance of the proposed method, there are still some promising directions to explore in the future:

1) Localization Enhancement. As our AdaptCOD framework progressively improves the segmentation results, the accurate localization of the camouflaged objects has a large impact on the performance, especially for some extremely complex scenarios.

2) Background Retrieval Manners. To refine the local details, our AdaptCOD retrieves the background regions by measuring the similarities between the background indexing vector and the candidate foreground locations. We believe there would be more appropriate measuring methods that can be used to further improve the segmentation quality near the boundaries of camouflaged objects.

References

1. Bernal, J., Sánchez, F.J., Fernández-Esparrach, G., Gil, D., Rodríguez, C., Vilariño, F.: Wm-dova maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians. CMIG **43**, 99–111 (2015)
2. Bernal, J., Sánchez, J., Vilarino, F.: Towards automatic polyp detection with a polyp appearance model. Pattern Recognition **45**(9), 3166–3182 (2012)
3. Chen, G., Liu, S.J., Sun, Y.J., Ji, G.P., Wu, Y.F., Zhou, T.: Camouflaged object detection via context-aware cross-level fusion. IEEE Transactions on Circuits and Systems for Video Technology **32**(10), 6981–6993 (2022)
4. Cheng, X., Xiong, H., Fan, D.P., Zhong, Y., Harandi, M., Drummond, T., Ge, Z.: Implicit motion handling for video camouflaged object detection. In: CVPR (2022)
5. Chu, X., Tian, Z., Wang, Y., Zhang, B., Ren, H., Wei, X., Xia, H., Shen, C.: Twins: Revisiting the design of spatial attention in vision transformers. In: NeurIPS (2021)
6. De Boer, P.T., Kroese, D.P., Mannor, S., Rubinstein, R.Y.: A tutorial on the cross-entropy method. Ann. Oper. Res. **134**(1), 19–67 (2005)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: NAACL (2018)
8. Fan, D.P., Cheng, M.M., Liu, Y., Li, T., Borji, A.: Structure-measure: A new way to evaluate foreground maps. In: IEEE ICCV (2017)
9. Fan, D.P., Gong, C., Cao, Y., Ren, B., Cheng, M.M., Borji, A.: Enhanced-alignment measure for binary foreground map evaluation. In: IJCAI (2018)
10. Fan, D.P., Ji, G.P., Cheng, M.M., Shao, L.: Concealed object detection. IEEE TPAMI **44**(10), 6024–6042 (2022)
11. Fan, D.P., Ji, G.P., Sun, G., Cheng, M.M., Shen, J., Shao, L.: Camouflaged object detection. In: IEEE CVPR (2020)
12. Fan, D.P., Ji, G.P., Zhou, T., Chen, G., Fu, H., Shen, J., Shao, L.: Pranet: Parallel reverse attention network for polyp segmentation. In: MICCAI (2020)
13. Fan, D.P., Zhang, J., Xu, G., Cheng, M.M., Shao, L.: Salient objects in clutter. IEEE TPAMI **45**(2), 2344–2366 (2023)
14. Pérez-de la Fuente, R., Delclòs, X., Peñalver, E., Speranza, M., Wierzchos, J., Ascaso, C., Engel, M.S.: Early evolution and ecology of camouflage in insects. Proceedings of the National Academy of Sciences **109**(52), 21414–21419 (2012)
15. Han, K., Xiao, A., Wu, E., Guo, J., Xu, C., Wang, Y.: Transformer in transformer. In: NeurIPS (2021)
16. He, C., Li, K., Zhang, Y., Tang, L., Zhang, Y., Guo, Z., Li, X.: Camouflaged object detection with feature decomposition and edge reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22046–22055 (2023)
17. He, R., Dong, Q., Lin, J., Lau, R.W.: Weakly-supervised camouflaged object detection with scribble annotations. In: AAAI (2023)
18. Hou, Q., Cheng, M.M., Hu, X., Borji, A., Tu, Z., Torr, P.: Deeply supervised salient object detection with short connections. IEEE TPAMI **41**(4), 815–828 (2019)
19. Hu, X., Wang, S., Qin, X., Dai, H., Ren, W., Luo, D., Tai, Y., Shao, L.: High-resolution iterative feedback network for camouflaged object detection. In: AAAI (2023)
20. Huang, Z., Dai, H., Xiang, T.Z., Wang, S., Chen, H.X., Qin, J., Xiong, H.: Feature shrinkage pyramid for camouflaged object detection with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5557–5566 (2023)

21. Jha, D., Smedsrud, P.H., Riegler, M.A., Halvorsen, P., Lange, T.d., Johansen, D., Johansen, H.D.: Kvasir-seg: A segmented polyp dataset. In: MMM (2020)
22. Ji, G.P., Fan, D.P., Chou, Y.C., Dai, D., Liniger, A., Van Gool, L.: Deep gradient learning for efficient camouflaged object detection. MIR (2022), doi: https://doi.org/10.1007/s11633-022-1365-9
23. Ji, G.P., Zhu, L., Zhuge, M., Fu, K.: Fast camouflaged object detection via edge-based reversible re-calibration network. PR **123**, 108414 (2022)
24. Jia, Q., Yao, S., Liu, Y., Fan, X., Liu, R., Luo, Z.: Segment, magnify and reiterate: Detecting camouflaged objects the hard way. In: IEEE CVPR (2022)
25. Lamdouar, H., Yang, C., Xie, W., Zisserman, A.: Betrayed by motion: Camouflaged object discovery via motion segmentation. In: ACCV (2020)
26. Le, T.N., Cao, Y., Nguyen, T.C., Le, M.Q., Nguyen, K.D., Do, T.T., Tran, M.T., Nguyen, T.V.: Camouflaged instance segmentation in-the-wild: Dataset, method, and benchmark suite. IEEE TIP **31**, 287–300 (2022)
27. Le, T.N., Nguyen, T.V., Nie, Z., Tran, M.T., Sugimoto, A.: Anabranch network for camouflaged object segmentation. CVIU **184**, 45–56 (2019)
28. Le, X., Mei, J., Zhang, H., Zhou, B., Xi, J.: A learning-based approach for surface defect detection using small image datasets. Neurocomputing (2020)
29. Li, A., Zhang, J., Lv, Y., Liu, B., Zhang, T., Dai, Y.: Uncertainty-aware joint salient object and camouflaged object detection. In: IEEE CVPR (2021)
30. Li, C., Jiao, G.: Einet: camouflaged object detection with pyramid vision transformer. JEI **31**(5), 053002 (2022)
31. Li, Z.Y., Gao, S., Cheng, M.M.: Exploring feature self-relation for self-supervised transformer. arXiv preprint arXiv:2206.05184 (2022)
32. Lin, J., Tan, X., Xu, K., Ma, L., Lau, R.W.: Frequency-aware camouflaged object detection. ACM TMCCA **19**(2), 1–16 (2023)
33. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: IEEE CVPR (2017)
34. Liu, Y., Cheng, M.M., Fan, D.P., Zhang, L., Bian, J.W., Tao, D.: Semantic edge detection with diverse deep supervision. International Journal of Computer Vision **130**(1), 179–198 (2022)
35. Liu, Z., Zhang, Z., Wu, W.: Boosting camouflaged object detection with dual-task interactive transformer. ICPR (2022)
36. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. ICLR (2017)
37. Lv, Y., Zhang, J., Dai, Y., Li, A., Barnes, N., Fan, D.P.: Towards deeper understanding of camouflaged object detection. IEEE TCSVT (2023)
38. Lv, Y., Zhang, J., Dai, Y., Li, A., Liu, B., Barnes, N., Fan, D.P.: Simultaneously localize, segment and rank the camouflaged objects. In: IEEE CVPR (2021)
39. Margolin, R., Zelnik-Manor, L., Tal, A.: How to evaluate foreground maps? In: IEEE CVPR (2014)
40. Máttyus, G., Luo, W., Urtasun, R.: Deeproadmapper: Extracting road topology from aerial images. In: IEEE ICCV (2017)
41. Mei, H., Ji, G.P., Wei, Z., Yang, X., Wei, X., Fan, D.P.: Camouflaged object segmentation with distraction mining. In: IEEE CVPR (2021)
42. Mei, H., Xu, K., Zhou, Y., Wang, Y., Piao, H., Wei, X., Yang, X.: Camouflaged object segmentation with omni perception. International Journal of Computer Vision pp. 1–16 (2023)
43. Mei, H., Yang, X., Zhou, Y., Ji, G.P., Wei, X., Fan, D.P.: Distraction-aware camouflaged object segmentation. SCIS (2023)

44. Mondal, A., Ghosh, S., Ghosh, A.: Partially camouflaged object tracking using modified probabilistic neural network and fuzzy energy based active contour. International Journal of Computer Vision **122**, 116–148 (2017)
45. Pang, Y., Zhao, X., Xiang, T.Z., Zhang, L., Lu, H.: Zoom in and out: A mixed-scale triplet network for camouflaged object detection. In: IEEE CVPR (2022)
46. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. In: NeurIPS (2019)
47. Perazzi, F., Krähenbühl, P., Pritch, Y., Hornung, A.: Saliency filters: Contrast based filtering for salient region detection. In: IEEE CVPR (2012)
48. Rahman, M.M., Marculescu, R.: Medical image segmentation via cascaded attention decoding. In: IEEE WACV (2023)
49. Ren, J., Hu, X., Zhu, L., Xu, X., Xu, Y., Wang, W., Deng, Z., Heng, P.A.: Deep texture-aware features for camouflaged object detection. IEEE TCSVT **33**(3), 1157–1167 (2023)
50. Silva, J., Histace, A., Romain, O., Dray, X., Granado, B.: Toward embedded detection of polyps in wce images for early diagnosis of colorectal cancer. International journal of computer assisted radiology and surgery **9**, 283–293 (2014)
51. Srinivas, A., Lin, T.Y., Parmar, N., Shlens, J., Abbeel, P., Vaswani, A.: Bottleneck transformers for visual recognition. In: IEEE CVPR (2021)
52. Sun, Y., Chen, G., Zhou, T., Zhang, Y., Liu, N.: Context-aware cross-level fusion network for camouflaged object detection. In: IJCAI (2021)
53. Sun, Y., Wang, S., Chen, C., Xiang, T.Z.: Boundary-guided camouflaged object detection. In: IJCAI (2022)
54. Tabernik, D., Šela, S., Skvarč, J., Skočaj, D.: Segmentation-based deep-learning approach for surface-defect detection. J. Intell. Manuf. **31**(3), 759–776 (2020)
55. Tajbakhsh, N., Gurudu, S.R., Liang, J.: Automated polyp detection in colonoscopy videos using shape and context information. IEEE TMI **35**(2), 630–644 (2015)
56. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017)
57. Wang, H., Wang, X., Sun, F., Song, Y.: Camouflaged object segmentation with transformer. In: ICCSIP (2021)
58. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pvt v2: Improved baselines with pyramid vision transformer. CVMJ **8**(3), 415–424 (2022)
59. Wu, F., Li, X., Zhang, Y., Hu, K.: Transcoop: Cooperation of transformers and cnns for camouflaged object segmentation. In: IEEE ICME (2022)
60. Wu, H., Xiao, B., Codella, N., Liu, M., Dai, X., Yuan, L., Zhang, L.: Cvt: Introducing convolutions to vision transformers. In: IEEE ICCV (2021)
61. Wu, M., Zhang, X., Sun, X., Zhou, Y., Chen, C., Gu, J., Sun, X., Ji, R.: Difnet: Boosting visual information flow for image captioning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18020–18029 (2022)
62. Xie, S., Tu, Z.: Holistically-nested edge detection. In: IEEE ICCV (2015)
63. Xu, B., Liang, H., Liang, R., Chen, P.: Locate globally, segment locally: A progressive architecture with knowledge review network for salient object detection. In: AAAI (2021)
64. Xu, W., Xu, Y., Chang, T., Tu, Z.: Co-scale conv-attentional image transformers. In: IEEE ICCV (2021)

65. Yang, F., Zhai, Q., Li, X., Huang, R., Luo, A., Cheng, H., Fan, D.P.: Uncertainty-guided transformer reasoning for camouflaged object detection. In: IEEE ICCV (2021)
66. Yang, J., Li, C., Zhang, P., Dai, X., Xiao, B., Yuan, L., Gao, J.: Focal attention for long-range interactions in vision transformers. In: NeurIPS (2021)
67. Yin, B., Zhang, X., Hou, Q., Sun, B.Y., Fan, D.P., Van Gool, L.: Camoformer: Masked separable attention for camouflaged object detection. arXiv preprint arXiv:2212.06570 (2022)
68. Zhai, Q., Li, X., Yang, F., Chen, C., Cheng, H., Fan, D.P.: Mutual graph learning for camouflaged object detection. In: CVPR (2021)
69. Zhai, Q., Li, X., Yang, F., Jiao, Z., Luo, P., Cheng, H., Liu, Z.: Mgl: Mutual graph learning for camouflaged object detection. IEEE TIP **32**, 1897–1910 (2023)
70. Zhai, W., Cao, Y., Xie, H., Zha, Z.J.: Deep texton-coherence network for camouflaged object detection. IEEE TMM (2022)
71. Zhang, M., Xu, S., Piao, Y., Shi, D., Lin, S., Lu, H.: Preynet: Preying on camouflaged objects. In: ACM MM (2022)
72. Zhang, Q., Ge, Y., Zhang, C., Bi, H.: Tprnet: camouflaged object detection via transformer-induced progressive refinement network. TVCJ pp. 1–15 (2022)
73. Zhang, X., Sun, X., Luo, Y., Ji, J., Zhou, Y., Wu, Y., Huang, F., Ji, R.: Rstnet: Captioning with adaptive attention on visual and non-visual words. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 15465–15474 (2021)
74. Zheng, D., Zheng, X., Yang, L.T., Gao, Y., Zhu, C., Ruan, Y.: Mffn: Multi-view feature fusion network for camouflaged object detection. In: WACV (2023)
75. Zheng, H., Fu, J., Zha, Z.J., Luo, J.: Looking for the devil in the details: Learning trilinear attention sampling network for fine-grained image recognition. In: IEEE CVPR (2019)
76. Zhong, Y., Li, B., Tang, L., Kuang, S., Wu, S., Ding, S.: Detecting camouflaged object in frequency domain. In: IEEE CVPR (2022)
77. Zhou, Y., Li, Z., Guo, C.L., Bai, S., Cheng, M.M., Hou, Q.: Srformer: Permuted self-attention for single image super-resolution. In: ICCV (2023)
78. Zhu, H., Li, P., Xie, H., Yan, X., Liang, D., Chen, D., Wei, M., Qin, J.: I can find you! boundary-guided separated attention network for camouflaged object detection. In: AAAI (2022)