Guided multi-scale refinement network for camouflaged object detection

Xiuqi Xu1 · Shuhan Chen1 · Xiao Lv2 · Jian Wang1 · Xuelong Hu1

Received: 15 July 2020 / Revised: 9 March 2021 / Accepted: 24 May 2022 / Published online: 30 July 2022
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract
The purpose of camouflaged object detection (COD) is to identify the hidden camouflaged object in an input image. Compared with other binary segmentation tasks like salient object detection, COD needs to deal with more complex scenes, such as low contrast, similar foreground and background. In this work, we proposed a novel guided multi-scale refinement network for COD. Specifically, we first design a global perception module for coarse localization by stacking multi-scale residual block on the top of the backbone in a recurrent manner. Then, we propose the guided multi-scale refinement module to refine such initial prediction progressively, which is combined with multi-level side-output features in a prediction-to-feature fusion strategy. By plugging into side-output features for multi-scale guidance, the missing object parts and false detection can be well remedied. Experimental results show that our proposed network can more accurately locate the camouflaged object and salient object with sharpened details than existing state-of-the-art approaches. In addition, our model is also very efficient and compact, which enables potential real-world applications.

Keywords Camouflaged object detection · Multi-scale global perception · Guided multi-scale refinement

Shuhan Chen
shchen@yzu.edu.cn

Xiuqi Xu
frequency.xu@foxmail.com

Xiao Lv
lvxiao87@126.com

Jian Wang
Haixiaoqu@163.com

Xuelong Hu
xlu@yzu.edu.cn

1 School of Information Engineering, Yangzhou University, Yangzhou, China

2 Chongqing Special Equipment Inspection and Research Institute, Chongqing, China
1 Introduction

Camouflaged object detection (COD) aims at finding and segmenting the camouflaged object from the background. It is a newly-developing task and also very challenging. As shown in Fig. 1, can you find the concealed object in the first image? Apparently, it is very difficult even for our human being. Therefore, although both as the binary segmentation task, COD is more challenging than salient object detection (SOD), as seen in Fig. 1.

The creatures in nature have their own unique survival skills and camouflage is one of them. The long process of evolution has earned many creatures the nickname of “The master of camouflage”. Creatures use their own structures and physiological characteristics to reduce the risk of being recognized by their predators. A variety of biological camouflage is accompanied by natural selection and survival of the fittest of natural evolution, with the history of life as long as the development process. Meanwhile, camouflage in biology also plays an important role in human scientific research and is widely used in military, agricultural and medical fields. In the field of medical imaging (e.g., lung infection segmentation [49]), in the outbreak of COVID-19 worldwide in February 2020, low-contrast chest imaging caused a lot of trouble to the diagnosis of the disease. This can lead to doctors making wrong diagnoses, delaying the disease and leading to serious consequences. Therefore, camouflage object detection is not only an important contribution to scientific research but also of great significance to human society.

Both as a binary segmentation task, SOD [11, 13, 16, 24, 25, 50, 53, 59, 61, 63] have developed rapidly, but the field of COD has not received much attention. Generally, camouflage object usually has indistinguishable boundary and low contrast with the background when compared with SOD task, which makes it difficult for SOD methods to accurately locate the camouflaged objects. In this paper, we propose Guided multi-scale Refinement Network (GRN) which incorporates multi-scale residual block into both localization and refinement process to capture useful semantic information as much as possible. Specifically, it starts with a multi-scale Global Perception (GP) module to find the coarse location of the camouflaged object. In the refinement step, although many multi-scale feature fusion methods have been proposed in existing SOD models, they are still facing the problem that high-level features are gradually diluted in the fusion process. To address it, we further propose Guided multi-scale Refinement (GR) module to merge multi-level features with top-down guidance to alleviate the dilution problem.

In summary, the contributions of this paper can be summarized as follows:

1) We first propose multi-scale GP module to capture richer semantic information for the precise initial location of camouflaged object.

2) We further propose GR module to refine the initial prediction in a multi-scale manner with top-down guidance. With its help, the missing object parts and details can be well remedied.

Fig. 1 Visual comparison between COD (left) and SOD (right). As can be seen, it is very difficult to find the camouflaged object in the first image even for our human being. Thus, COD is much more challenging than SOD. From left to right are input images and corresponding ground truth.
3) Extensive experiments on three COD datasets and four SOD datasets show the state-of-the-art performance of the proposed model. In addition, our model is also very efficient and compact.

2 Related work

2.1 Salient object detection

Multi-scale fusion In the past years, many efforts about various network architectures have been made in SOD. Among them, multi-scale feature fusion is a common way to combine the complementary cues between high-level features in deep layers and low-level features in shallow layers. Hou et al. [17] introduced short connections to the skip-layer structures within the HED [52] architecture. Liu et al. [28] leveraged global and local pixel-wise contextual attention network to capture global and local context information. Deng et al. [5] learned the residual between the intermediate saliency prediction and the ground truth by alternatively leveraging the low-level integrated features and the high-level integrated features of a fully convolutional network (FCN) [31]. Wu et al. [51] proposed a novel cascaded partial decoder framework which directly utilizes generated saliency map to refine the features of backbone network. This strategy efficiently suppresses distractors in the features and improves their representation ability significantly. Wei et al. [48] aggregated multilevel features selectively and adaptively selected complementary components from embedded features before fusion, effectively avoiding the introduction of too much redundant information and the destruction of original features. Pang et al. [37] proposed an aggregate interaction module to integrate the features from adjacent levels, using small up-/down-sampling rates and introducing less noise.

Edge-aware Recently, some edge-aware methods are proposed to further improve the visual quality of saliency maps near the object boundaries. Zhao et al. [64] made use of the rich edge information and location information, locating the significant targets more accurately by fusing features. Qin et al. [39] proposed a predict-refine architecture which is composed of a densely supervised Encoder-Decoder network and a residual refinement module. They define the hybrid loss to guide the network to learn the transformation between the input image and the ground truth in a three-level hierarchy. Su et al. [20] proposed a boundary-aware network. In this network, the feature selectivity at boundaries is enhanced by incorporating a boundary localization stream, while the feature invariance at interiors is guaranteed with a complex interior perception stream. Wu et al. [50] proposed a Stacked Cross Refinement Network which aims to simultaneously refine multi-level features of salient object detection and edge detection by stacking cross refinement unit.

However, most of the above methods either having high-complexity or facing the feature dilution problem. In this paper, we propose a simple GR to let the deeper-layer prediction guide the shallower layer features extraction for more effective and efficient fusion, where the precise location of deep prediction and the rich spatial details from shallow features will be comprehensively integrated.

2.2 Camouflaged object detection

Compared to the SOD task, COD has not been fully explored due to its challenging. Conventional methods [36, 41, 44] typically use hand-crafted features (e.g., optical flow, color,
texture) for camouflaged region detection. In the deep learning era, CNN-based methods have emerged as promising alternatives for COD. In [66], Zheng et al. constructed a large camouflaged people dataset while they built a dense deconvolution network for people detection. Similar to [66], Le et al. [21] provided a new generic camouflaged objects dataset as well as a two-branch network named Anabranck Network for this task. More recently, Fan et al. [10] proposed a large-scale COD dataset called COD10K and introduced a two-stage network called SINet, which consists of the search module and the identification module.

To further promote this and improve the performance, in this paper, we propose a novel framework called GRN for COD. Specifically, we innovatively integrated multi-scale residual block both into global perception and residual refinement to capture richer semantic information for more accurate detection.

3 Proposed framework

We first introduce our motivation and the whole architecture in Section 3.1 which is based on the multi-scale residual block. Then, we integrate it into the U-net architecture for global perception module and residual refinement module which are described in Sections 3.2 and 3.3 respectively. Finally, we present the loss function in Section 3.4.

3.1 Overview

As mentioned above, different from the salient object that usually has significant difference with background, the camouflaged object can well conceal itself into the background thus is very difficult to be discovered. In the human visual system, various population receptive fields helps us to find objects with different scales as studied in neuroscience experiments. In computer vision tasks, the receptive field is also very important to enrich context information, thus, various effective structures have been proposed to enlarge the receptive field of the network for better performance, such as ASPP (Atrous spatial pyramid pooling) [4], DenseASPP [55], RFNet (An end-to-end image matching network based on receptive field) [45], etc.

Recently, Cheng et al. [12] proposed a multi-scale residual based Res2Net for image classification, which achieved better performance compared to the original ResNet [15]. Specifically, it divides the input feature evenly into \( p \) feature subsets, which have the same spatial size. One split subset is first fed into a convolutional layer for feature learning, which is further added with its next subset and then fed into another convolutional layer. This process repeated several times until all the split subsets are processed. Finally, all these output subsets after feature learning are concatenated together and sent to another \( 1 \times 1 \) convolution for residual learning. Res2Net expands the receptive field by replacing standard residual block with the above one. Benefit from this structure, these splits were processed in a multi-scale fashion, which was conducive to the extraction of both global and local information. Let \( f_k^i \) and \( \hat{f}_k^i \) be the input and output feature in split group \( k \), “Conv” denotes \( 3 \times 3 \) convolutional operation, the above process can be defined as:

\[
\hat{f}_k = \begin{cases} 
\text{Conv}_k(f_k), & k = 1; \\
\text{Conv}_k(\hat{f}_{k-1} + f_k), & 1 < k < p; \\
f_k, & k = p.
\end{cases}
\]
Fig. 2 The whole architecture of the proposed network. The gray blocks mark the trunk of backbone network. GP takes the deepest feature as input to generate the coarse localization of the camouflaged object. The initial prediction is further refined progressively by the proposed GR module with the supervision on each side-output. The convolutional layer is adopted for channel reduction.

Inspired by this successful structure, we introduce it into our framework to enrich semantic information for COD. Our network is implemented in a two-stage strategy based on the following observation: in human visual system, we will first search the whole space when we try to find a target, if we find it we will focus on it to complete its details. Following it, our network is composed of two parts: multi-scale Global Perception (GP) module and Guided multi-scale Refinement (GR) module, which are designed for coarse initial localization and residual refinement respectively. Starting with GP to locate camouflaged object roughly, we apply our proposed GR to refine it progressively with multi-scale top-down guidance. The whole architecture is illustrated in Fig. 2. The details of GP and GR are described in the following subsections.

3.2 Multi-scale global perception

We notice that there are some outstanding structures proposed to capture multi-scale contexts, such as PPM (Pyramid Pooling Module) [29], ASPP [4]. Different from these parallel concatenation based methods, we build a multi-scale global perception module by simply

Fig. 3 Multi-scale residual block proposed in Res2Net [12]. Here, $g$ is set to 4 for illustration.
stacking the above multi-scale residual block (shown in Fig. 3) on the top of the last layer of the backbone network to increase the receptive field. We stack it in a recurrent manner to reduce the number of model parameters while keeping the ability of the module to capture high-level semantic information. The detailed comparison of the recurrent block numbers is shown in the ablation experiment section and Table 3. The output of the GP gives the coarse localization of the camouflaged object, which needs to be further refined by the proposed GR module as described in the next subsection.

### 3.3 Guided multi-scale refinement module

After obtained the coarse prediction, we need to remedy the missing object parts and also fix the false detection by integrating multi-level convolutional features. As we know, different layers of deep CNNs learn different scale features and shallow layers capture low-level structure information while deep layers capture high-level semantic information. Typically, there are three ways to fuse their complementary information for better feature representation: feature to feature fusion, prediction to prediction, and prediction to feature fusion. The first one needs to continuously upsample features in deep layers for aggregation, such as FPN [26], which is not efficient enough, especially in the case of many channels and large feature size. The second one is efficient but incompetent to preserve some important cues especially in challenging cases, such as residual learning in R3Net [5]. In general, there are two basic requirements for seamless aggregation: high-level semantic information should be fully preserved, while noisy distractions in shallow layers should be filtered out as much as possible. Following these two criteria, we design an efficient guided refinement module in a prediction to feature fusion manner.

As seen in Fig. 4, the input side-output feature with channel $C$ is first equally split into $g$ groups, each of which consists of $\frac{C}{g}$-channel feature maps. Then, the deeper side-output prediction is used as a guidance feature map to be concatenated with the $\frac{C}{g}$-channel feature maps within each group to achieve $(\frac{C}{g} + 1)$-channel feature maps. A convolution with kernel $3 \times 3$ is performed on each concatenated feature map except for the last one. Let $x^i$ and $\hat{x}^i$ be the input and output feature in side-output stage $i$, the concatenated feature in group $k$ is added with the output of “Conv$_{k-1}$” and then fed into “Conv$_k$”. As suggested in Res2Net [12], $g$ is set to 4 in this paper. Thus, the output feature of group $k$ can be formulated as:

$$\hat{x}^i_k = \begin{cases} 
\text{Conv}_k(\text{Cat}(x^i_k, y^{i+1})), & k = 1; \\
\text{Conv}_k(\hat{x}^i_{k-1} + \text{Cat}(x^i_k, y^{i+1})), & 1 < k < 4; \\
\text{Cat}(x^i_k, y^{i+1}), & k = 4,
\end{cases} \quad (2)$$

![Fig. 4 Structure of the GR module. The purple block is the input which is split equally for subsequent refinement. Each split group will be concatenated with the upsampled prediction from the deep layer, the following steps are the same as the multi-scale residual block](image)
where “Cat” denotes concatenation operation. Then, all the $\hat{x}_k^i$ are concatenated together to obtain a $(C + g)$-channel feature which is fed into several convolutional layers for guided refinement to produce a residual to be added with the input prediction as refined output in side-output stage $i$. The above operation can be expressed as the following formula:

$$y^i = y^{i+1} + \text{Convs}(\hat{x}^i).$$ (3)

It is worthy to point out that the proposed guided multi-scale refinement method is different from $R^3$Net [5], where the deep prediction is directly concatenated with the convolutional feature for residual refinement, which can be classified into single-scale guidance strategy. Furthermore, all the side-output features are first grouped into high-level and low-level features respectively for recurrent refinement in $R^3$Net. While in our model, the multi-level features are fed into our GR module progressively for residual refinement. It doesn’t need upsample features with multiple channels for concatenation, thus can be more efficient.

### 3.4 Loss function

We define our training loss as the summation over all side-outputs:

$$\mathcal{L} = \sum_{(q=1)}^{Q} \alpha_q l^{(q)},$$ (4)

where $l^{(q)}$ is the loss of the q-th side output, $Q$ denotes the total number of the side-outputs and $\alpha_q$ is the weight of each loss.

To obtain high quality regional segmentation and clear boundaries, we propose to define it as a hybrid loss:

$$l^{(q)} = l^{(q)}_{bce} + l^{(q)}_{iou},$$ (5)

where $l^{(q)}_{bce}$ and $l^{(q)}_{iou}$ denote BCE loss [1] and IoU loss [34], respectively.

BCE [1] loss is the most widely used loss in binary classification and segmentation. It is defined as:

$$l_{bce} = - \sum_{(r,c)} [G_{(r,c)} \log(S_{(r,c)}) + (1 - G_{(r,c)}) \log(1 - S_{(r,c)})],$$ (6)

where $G_{(r,c)} \in \{0, 1\}$ is the ground truth label of the pixel $(r, c)$ and $S_{(r,c)}$ is the predicted probability of being salient object.

IoU [34] is originally proposed for measuring the similarity of two sets [19] and then used as a standard evaluation measure for object detection and segmentation. Recently, it has been used as the training loss for SOD [34, 40]. It is defined as bellow:

$$l_{iou} = 1 - \frac{\sum_{r=1}^{H} \sum_{c=1}^{W} S_{(r,c)} G_{(r,c)}}{\sum_{r=1}^{H} \sum_{c=1}^{W} \left[ S_{(r,c)} + G_{(r,c)} - S_{(r,c)}G_{(r,c)} \right]}.$$ (7)
4 Experiments

4.1 Dataset

COD datasets We perform extensive experiments on the following three widely used camouflaged object detection datasets: CHAMELEON [43] contains 76 images that were collected from the Internet via the Google search engine using “camouflaged animal” as a keyword, with manually annotated object-level ground-truths (GT). Another contemporary dataset is CAMO [22], which has 2,500 images (2,000 for training, 500 for testing) covering eight categories. COD10K [10] is a more challenging, higher quality, and densely annotated dataset. COD10K contains 10,000 images (5,066 camouflaged, 3,000 background, 1,934 non-camouflaged), divided into 10 super-classes, and 78 sub-classes (69 camouflaged, nine non-camouflaged) which are collected from multiple photography websites. To provide a large amount of training data for deep learning models, COD10K is split into 6,000 images for training and 4,000 for testing, randomly selected from each sub-class.

SOD datasets To evaluate the learning ability of our model, we also conduct experiments on four widely-adopted salient object datasets: ECSSD [54], DUTS-OMRON [35], HKU-IS [25] and DUTS [47]. Among the four datasets, DUTS is a large scale dataset containing 10,553 training images (denoted as DUTS-TR) and 5,019 test images (denoted as DUTS-TE). More details can be found in recently released standard benchmarks [6].

4.2 Evaluation metrics

Four metrics are utilized to evaluate our model and existing state-of-the-art approaches. MAE (Mean absolute error) is a metric that evaluates the average difference between the prediction map and ground truth. Let \( P \) and \( G \) denote the saliency map and ground truth which is normalized to \([0,1]\). We compute the MAE by
\[
\text{MAE} = \frac{1}{WH} \sum_{x=1}^{W} \sum_{y=1}^{H} |P(x, y) - G(x, y)|
\]
where \( W \) and \( H \) are the width and height of images, respectively. F-measure is a harmonic mean of precision and recall, formulated as
\[
F_\beta = \frac{(1+\beta^2) \cdot \text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision} + \text{Recall}}
\]
where \( \beta^2 \) is set to 0.3 to emphasize the precision over recall, as suggested in the previous works [3, 61, 64]. Weight F-measure [33] has offered an intuitive generalization of the \( F_\beta \), and is defined as
\[
F^w_\beta = \frac{(1+\beta^2) \cdot \text{Precision}^w \cdot \text{Recall}^w}{\beta^2 \cdot \text{Precision}^w + \text{Recall}^w}
\]
and it identified other causes of inaccurate evaluation. S-measure (structural similarity measure) [7] focuses on evaluating the structural similarity, which is much closer to human visual perception. It can be computed as
\[
S_\alpha = \alpha s_o + (1 - \alpha)s_r
\]
where \( s_o \) and \( s_r \) denote the region-aware and object-aware structural similarity and \( \alpha \) is set as 0.5. E-measure (enhanced-alignment measure) [8] combines local pixel values with the image-level mean value in one term, jointly capture image-level statistic and local pixel matching information, can be computed as
\[
E_\xi = \frac{1}{WH} \sum_{x=1}^{W} \sum_{y=1}^{H} \theta(\xiFM), \text{ where } \theta(\xiFM) \text{ is the enhanced alignment matrix.}
\]

4.3 Implementation details

All the experiments are implemented in PyTorch and on a PC with an NVIDIA TITAN Xp GPU and Intel i7-6700K CPU 4.0 GHz processor. Pre-trained ResNet-50 [15] on ImageNet is used as backbone network. We apply data augmentation like random flipping and multi-scale input images to alleviate over-fitting risk. All the images are resized into 352×352
Table 1  Quantitative results on different COD datasets in four metrics: max E-measure (higher better), S-measure (higher better), F-measure (higher better) and MAE (lower better)

| Baseline models | CHAMELEON [43] | CAMO [22] | COD10K [10] |
|-----------------|-----------------|-----------|-------------|
| No. | Model | Year | Pub. | $E_{ϕ}$ ↑ | $S_{α}$ ↑ | $F_{β}^{σ}$ ↑ | $M$ ↓ | $E_{ϕ}$ ↑ | $S_{α}$ ↑ | $F_{β}^{σ}$ ↑ | $M$ ↓ | $E_{ϕ}$ ↑ | $S_{α}$ ↑ | $F_{β}^{σ}$ ↑ | $M$ ↓ |
| 1 | FPN [26] | 2017 | CVPR | 0.783 | 0.794 | 0.590 | 0.075 | 0.677 | 0.684 | 0.483 | 0.131 | 0.691 | 0.697 | 0.411 | 0.075 |
| 2 | MaskRCNN [14] | 2017 | CVPR | 0.778 | 0.643 | 0.518 | 0.099 | 0.715 | 0.574 | 0.430 | 0.151 | 0.748 | 0.613 | 0.402 | 0.080 |
| 3 | PSPNet [62] | 2017 | CVPR | 0.758 | 0.773 | 0.555 | 0.085 | 0.659 | 0.663 | 0.455 | 0.139 | 0.680 | 0.678 | 0.377 | 0.080 |
| 4 | UNet++ [67] | 2018 | DLMIA | 0.762 | 0.695 | 0.501 | 0.094 | 0.653 | 0.599 | 0.392 | 0.149 | 0.672 | 0.623 | 0.350 | 0.086 |
| 5 | PiCANet [28] | 2018 | CVPR | 0.749 | 0.769 | 0.536 | 0.085 | 0.584 | 0.609 | 0.356 | 0.156 | 0.643 | 0.649 | 0.322 | 0.090 |
| 6 | MSRCNN [18] | 2019 | CVPR | 0.686 | 0.637 | 0.443 | 0.091 | 0.669 | 0.617 | 0.454 | 0.133 | 0.706 | 0.641 | 0.419 | 0.073 |
| 7 | PoolNet [29] | 2019 | CVPR | 0.779 | 0.776 | 0.555 | 0.081 | 0.698 | 0.702 | 0.494 | 0.129 | 0.713 | 0.705 | 0.416 | 0.074 |
| 8 | BASNet [39] | 2019 | CVPR | 0.721 | 0.687 | 0.474 | 0.118 | 0.661 | 0.618 | 0.413 | 0.159 | 0.678 | 0.634 | 0.365 | 0.105 |
| 9 | PFANet [65] | 2019 | CVPR | 0.648 | 0.679 | 0.378 | 0.144 | 0.622 | 0.659 | 0.391 | 0.172 | 0.618 | 0.636 | 0.286 | 0.128 |
| 10 | CPD [51] | 2019 | CVPR | 0.866 | 0.853 | 0.706 | 0.052 | 0.729 | 0.726 | 0.550 | 0.115 | 0.770 | 0.747 | 0.508 | 0.059 |
| 11 | HTC [2] | 2019 | CVPR | 0.489 | 0.517 | 0.204 | 0.129 | 0.442 | 0.476 | 0.174 | 0.172 | 0.520 | 0.548 | 0.221 | 0.088 |
| 12 | EGNet [64] | 2019 | ICCV | 0.870 | 0.848 | 0.702 | 0.050 | 0.768 | 0.732 | 0.583 | 0.104 | 0.779 | 0.737 | 0.509 | 0.056 |
| 13 | SCRN [50] | 2019 | ICCV | 0.884 | 0.871 | 0.739 | 0.042 | 0.783 | 0.772 | 0.634 | 0.091 | 0.814 | 0.790 | 0.583 | 0.046 |
| 14 | F3Net [48] | 2020 | AAAI | 0.903 | 0.856 | 0.753 | 0.046 | 0.824 | 0.773 | 0.656 | 0.092 | 0.833 | 0.787 | 0.622 | 0.045 |
| 15 | SINet [10] | 2020 | CVPR | 0.891 | 0.869 | 0.740 | 0.044 | 0.771 | 0.751 | 0.606 | 0.100 | 0.806 | 0.771 | 0.551 | 0.051 |
| CR15 | Ours | – | – | **0.934** | **0.862** | **0.816** | **0.036** | **0.841** | **0.766** | **0.737** | **0.090** | **0.873** | **0.798** | **0.691** | **0.039** |

The training set is COD10K, which is the same as SINet [10]. The best two results are highlighted in red and blue respectively. ↑ and ↓ indicate higher and lower is better performance.
| No. | model      | ECSSD [54] | HKU-IS [25] | DUT-OMRON [35] | DUT-TE [47] |
|-----|------------|------------|--------------|----------------|-------------|
|     |            | $E\phi$   | $S\alpha$    | $F_w^\beta$   | $M$         | $E\phi$ | $S\alpha$ | $F_w^\beta$ | $M$      | $E\phi$ | $S\alpha$ | $F_w^\beta$ | $M$      |
| 1   | BAN [39]   | 0.928      | 0.924        | 0.923          | 0.035        | 0.950   | 0.913     | 0.900        | 0.032     | 0.892   | 0.879     | 0.815        | 0.040     |
| 2   | CPD [51]   | 0.925      | 0.918        | 0.917          | 0.037        | 0.944   | 0.905     | 0.891        | 0.034     | 0.887   | 0.869     | 0.805        | 0.043     |
| 3   | EGNet [64] | 0.927      | 0.925        | 0.920          | 0.037        | 0.948   | 0.916     | 0.898        | 0.032     | 0.891   | 0.887     | 0.815        | 0.039     |
| 4   | SCRN [10]  | 0.926      | 0.927        | 0.918          | 0.037        | 0.949   | 0.916     | 0.896        | 0.034     | 0.888   | 0.885     | 0.809        | 0.040     |
| 5   | F3Net [48] | 0.927      | 0.924        | 0.925          | 0.033        | 0.953   | 0.917     | 0.910        | 0.028     | 0.902   | 0.888     | 0.840        | 0.035     |
| 6   | MINet [37] | 0.927      | 0.925        | 0.924          | 0.033        | 0.953   | 0.919     | 0.909        | 0.029     | 0.898   | 0.884     | 0.828        | 0.037     |
| 7   | ITSD [68]  | 0.927      | 0.925        | 0.895          | 0.034        | 0.952   | 0.917     | 0.899        | 0.031     | 0.895   | 0.885     | 0.804        | 0.041     |
| 8   | Ours       | 0.924      | 0.927        | 0.926          | 0.033        | 0.953   | 0.920     | 0.912        | 0.028     | 0.899   | 0.886     | 0.836        | 0.038     |

The best two results are highlighted in red and blue respectively. The training set is DUTS-TR [47] for all the methods.
both for training and testing. The training set is COD10K, which is the same as SINet [10]. The hyper-parameters are set as follows: initial learning rate is set to 0.0005 which is decreased by 10 after 25 epochs, the batch size is set to 10 and the maximum epoch is 30.

4.4 Comparisons with SOTAs

In this section, we first compare the proposed model with state-of-the-arts methods on COD and SOD datasets. For a fair comparison, we use either the implementations with the pre-trained parameters or the detection results reported by the authors.

Quantitative evaluation on COD For COD task, our GRN is compared with 14 state-of-the-art methods [2, 14, 18, 28, 29, 39, 46, 48, 50, 51, 55, 62, 64, 65, 67]. Table 1 shows the detailed experimental results. As can be seen that our proposed network significantly outperforms other state-of-the-art methods on the COD dataset, in specific, we achieved

---

**Table 1**

| Im | GT | Ours | SINet | F³ Net | SCRN | CPD | EGNet |
|---|---|---|---|---|---|---|---|
| ![Image](image1.png) | ![Ground Truth](gt1.png) | ![Ours](ours1.png) | ![SINet](sinet1.png) | ![F³ Net](f3net1.png) | ![SCRN](scrn1.png) | ![CPD](cpd1.png) | ![EGNet](egnet1.png) |

Fig. 5 Visual comparison on COD task in some challenging cases, including objects mimicking camouflage, multiple and small targets, low contrast backgrounds, and occlusion objects. From left to right: input image, ground truth (GT), our method, SINet [10], F³ Net [48], SCRN [50], CPD [51], EGNet [64]
4% and 6.9% improvement over the second best model in terms of $E_\psi$ and $F_\beta_w$. For smaller datasets CHAMELEON and CAMO, we also obtained the best evaluation scores except for S-measure. Although additional edge supervisions are applied in SCRN, we are still comparable with it in S-measure.

**Quantitative evaluation on SOD** We also compare the performance of GRN with several of the other seven recent state-of-the-art methods [37, 39, 48, 50, 51, 64, 68] on the SOD datasets. Detailed experimental results are presented in Table 2. From those results, we can observe that the proposed GRN performs favorably with them especially on the challenging dataset DUT-OMRON.

**Visual comparison** Figure 5 shows a visual comparison of the results of our method with those of the other five recent state-of-the-art methods [2, 10, 37, 39, 48, 51, 64, 68] on the COD datasets. We choose images in different challenging scenes for comparisons, such as object background interference and low contrast background (1st and 2nd row), object mimics camouflage (3rd, 4th and 5th row), minimal objects (6th, 7th and 8th row) in the scene and occluded objects (9th and 10th row). According to Fig. 5, we can observe that our GRN performs well in all these challenging scenarios. In order to test the universality of the proposed network, we also selected existing SOD networks and compared them with ours on the SOD datasets. According to Fig. 6, we can see that our network also shows excellent detection performance in complex scenarios such as small targets and multiple targets.

**Attributes-based performance on SOC** Attribute is annotated in the challenging SOC [6] dataset that reflects common attributes in real-world scenarios. These annotations are helpful to study the performance of salient object detection model. The structural similarity scores of our proposed model and 10 common algorithms are presented in Table 3. As can be seen from the above table, our model got the highest score in six of the nine attributes. The rest of the attributes were also highly rated or ranked at the top. These results show that the proposed model is applicable to most existing challenge scenarios, especially for the attributes of AC (Appearance Change), OC (Occlusion), SC (Shape Complexity) and SO (Small Object), which are consistent with the model characteristics described in the previous chapters. The network is designed for the purpose of detecting camouflaged objects, so even though it is not so outstanding in BO (Big Object), MB (Motion Blur) and

![Fig. 6 Visual comparison on SOD task. From left to right: input image, ground truth (GT), our method, ITSD [68], MINet [37], F3Net [48], SCRN [50], EGNNet [64], BAN [39], CPD [51]](image-url)
The structural similarity score between our method and the other six excellent methods is presented in the table below. The best two results are highlighted in red and blue respectively.

other attributes, it has achieved a relatively leading result. Furthermore, our model has faster testing times and smaller model sizes, which will be described below.

**Model complexity** The experiment proved that our proposed GRN is not only superior to most of the previous networks in performance, but also with advantages in model complexity. We compare our network with several classic networks, the details of which are shown in Table 4. We report the image size of each network, model size, selected backbone network, and FPS for the final test. Obviously, our model runs much faster than all the competing methods with a much more compact model size. Such high efficiency and compactness enable it a better choice for real-world applications.

### 4.5 Ablation analysis

In this section, we make an in-depth analysis of the proposed GRN with different design options.

**Number of residual blocks** In this variant, we adjusted the recurrent number of residual convolution operations in the GP module to investigate its impact on detection performance.

---

**Table 3** Attributes-based performance on the challenging SOC dataset [6]

| Attr | DSS [17] | CPD [51] | R3Net [5] | PICA [28] | EGNet [64] | SCRN [50] | Ours |
|------|----------|----------|-----------|-----------|------------|-----------|------|
| AC   | 0.753    | 0.799    | 0.713     | 0.792     | 0.806      | 0.809     | **0.819** |
| BO   | 0.561    | 0.647    | 0.437     | **0.729** | 0.528      | 0.689     | 0.643 |
| CL   | 0.722    | 0.773    | 0.659     | 0.787     | 0.757      | **0.795** | 0.789 |
| HO   | 0.767    | 0.803    | 0.740     | 0.809     | 0.802      | 0.823     | **0.830** |
| MB   | 0.719    | 0.754    | 0.657     | 0.775     | 0.762      | **0.792** | 0.787 |
| OC   | 0.718    | 0.750    | 0.653     | 0.765     | 0.754      | 0.775     | **0.789** |
| OV   | 0.700    | 0.748    | 0.624     | **0.781** | 0.752      | 0.774     | **0.789** |
| SC   | 0.761    | 0.793    | 0.716     | 0.784     | 0.793      | **0.809** | **0.813** |
| SO   | 0.713    | 0.745    | 0.682     | 0.748     | 0.749      | 0.767     | **0.786** |

---

**Table 4** Model complexity comparison with representative existing methods

| Models | Parameter | FPS ↑ | Image Size | Model Size ↓ | Backbone |
|--------|-----------|-------|------------|--------------|----------|
| 2019   | EGNet     | 10    | 400×300    | 447.1M       | ResNet-50 |
| 2019   | CPD       | 47    | 352×352    | 192.0M       | ResNet-50 |
| 2019   | SCRN      | 47    | 352×352    | 101.4M       | ResNet-50 |
| 2020   | SINet     | 40    | 352×352    | 196.5M       | ResNet-50 |
| **2020** | **Ours** | **65** | **352×352** | **97.3M** | **ResNet-50** |

The best is highlighted in bold
As can be seen from Table 5, the performance is gradually improved with the increase of the number of residual block until \( n = 5 \), in which the best performance is obtained. This shows that the multi-scale global perception module successfully enlarged the receptive field for better location of camouflaged objects. Behind of this, the experiment also shows that when the number of superimpositions of the residual block exceeds five times, the performance begins to gradually decrease, and the complexity and computational cost of the model are significantly increased. After considering computational complexity and performance, we choose \( n = 5 \) as our final setting.

**Guidance strategy** We also conduct experiment to assess the effectiveness of the proposed multi-scale guidance strategy by comparing to single-scale guidance, which directly concatenates the deep prediction with side-output features, such as R3Net [5]. From the results in Table 5, we can conclude that our multi-scale guidance strategy has a significant improvement over the single-scale manner, especially in E-measure and F-measure promotion. This indicates that more local information and spatial structure information are extracted from the network with the help of the multi-scale guidance, making the output of the network closer to the ground truth.

**5 Conclusion**

Camouflaged object detection is a very challenging object detection task as camouflaged objects are usually concealed in the background. In this paper, we presents an effective two-stage network for COD with multi-scale top-down guidance. By stacking several multi-scale residual blocks on the top of the backbone network in a recurrent manner, we can find the...
localization of the camouflaged object more accurately. Furthermore, such coarse prediction is refined progressively by fusing with side-output features in a multi-scale guided manner. As a result, the missing object parts and false detection can be well remedied. Experimental results on three COD datasets and four SOD datasets with existing state-of-the-art methods demonstrated its effectiveness. In addition, the proposed network is simpler and more efficient than the previous ones, which makes it a better choice for real-world applications.

Acknowledgements This work was supported by the Natural Science Foundation of China (No. 61802336), Jiangsu Province 7th Projects for Summit Talents in Six Main Industries, Electronic Information Industry (DZXK-149, No.110) and Yangzhou University “Qinglan Project”.

References

1. Boer P-TD, Dirk PK, Mannor S, Rubinstein RY (2005) A tutorial on the cross-entropymethod. Ann Oper Res 134(1):19–67
2. Chen K, Pang J, Wang J, Xiong Y, Li X, Sun S, Feng W, Liu Z, Shi J, Ouyang W (2019) Hybrid task cascade for instance segmentation. In: International conference on computer vision and pattern recognition, pp 4974–4983
3. Chen S, Tan X, Wang B, Hu X (2018) Reverse attention for salient object detection. In: European conference on computer vision, pp 234–250
4. Chen L-C, Zhu Y, Papandreou G, Schroff F, Adam H (2018) Encoder-Decoder with atrous separable convolution for semantic image segmentation. In: European conference on computer vision
5. Deng Z, Hu X, Zhu L, Xu X, Qin J, Han G, Heng P (2018) R3net: recurrent residual refinement network for saliency detection. In: Proceedings of the twenty-seventh international joint conference on artificial intelligence, pp 684–690
6. Fan D, Cheng M, Liu J, Gao S, Hou Q, Borji A (2018) Salient objects in clutter: bringing salient object detection to the foreground. In: European conference on computer vision, pp 186–202
7. Fan D, Cheng M, Liu Y, Li T, Borji A (2017) Structure-measure: a new way to evaluate foreground maps. In: IEEE international conference on computer vision, pp 4548–4557
8. Fan D, Gong C, Cao Y, Bo R, Cheng M, Borji A (2018) Enhanced-alignment measure for binary foreground map evaluation. In: International joint conference on artificial intelligence, pp 698–704
9. Fan D, Wang W, Cheng M, Shen J (2019) Shifting more attention to video salient object detection. In: International conference on computer vision and pattern recognition
10. Fan D, Ji G, Sun G, Cheng M, Shen J, Shao L (2020) Camouflaged object detection. In: International conference on computer vision and pattern recognition, pp 2777–2787
11. Fan D, Li T, Ji G, Zhang D, Cheng M, Fu H, Shen J (2021) Re-thinking co-salient object detection. In: IEEE transactions on pattern analysis and machine intelligence
12. Gao S, Cheng M, Zhao K, Zhang X, Yang M, Philip T (2021) Res2net: a new multi-scale backbone architecture. IEEE Trans Pattern Anal Mach Intell 43(2):652–662
13. Goferman S, Zelnik-Manor L, Tal A (2011) Context-aware saliency detection. In: IEEE transactions on pattern analysis and machine intelligence, pp 1915–1926
14. He K, Gkioxari G, Dollf’P, Girshick R (2017) Maskrcnn. In: IEEE international conference on computer vision, pp 2961–2969
15. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: International conference on computer vision and pattern recognition, pp 770–778
16. Hou Q, Cheng M, Hu X, Borji A, Tu Z, Torr PH (2019) Deeply supervised salient object detection with short connections. In: IEEE transactions on pattern analysis and machine intelligence
17. Hou Q, Cheng M, Hu X, Borji A, Tu Z, Torr PHS (2019) Deeply supervised salient object detection with short connections. IEEE Trans Pattern Anal Mach Intell 41(4):815–828
18. Huang Z, Huang L, Gong Y, Huang C, Wang X (2018) Mask scoring r-cnn. In: International conference on computer vision and pattern recognition, pp 6409–6418
19. Jaccard P (1912) The distribution of the flora in the alpine zone. In: New phytologist 11(2), 37–50, 961–967
20. Kim Y, Kim T, Kang B, Kim J, Kim D (2018) BAN: focusing On boundary context for object detection. In: Asian conference on computer vision, pp 555–570
21. Le T-N, Nguyen TV, Nie Z, Tran M-T, Sugimoto A (2019) Anabranch network for camouflaged object segmentation. Comput Vis Image Understand 184:45–56
22. Le T-N, Nguyen TV, Nie Z, Tran M-T, Sugimoto A (2019) Anabranch network for camouflaged object segmentation. In: Computer vision and image understanding, pp 45–56
23. Li G, Yu Y (2016) Deep contrast learning for salient object detection. In: International Conference on computer vision and pattern recognition, pp 478–487
24. Li J, Xia C, Song Y, Fang S, Chen X (2015) A data-driven metric for comprehensive evaluation of saliency models. In: International conference on computer vision and pattern recognition, pp 190–198
25. Li G, Yu Y (2015) Visual saliency based on multiscale deep features. In: International Conference on computer vision and pattern recognition, pp 5455–5463
26. Lin T, Dollar P, Girshick R, He K, Hariharan B, Belongie S (2017) Feature pyramid networks for object detection. In: International Conference on computer vision and pattern recognition, pp 2117–2125
27. Liu N, Han J (2016) Dhsnet, Deep hierarchical saliency network for salient object detection. In: International conference on computer vision and pattern recognition, pp 678–686
28. Liu N, Han J, Yang M (2018) PicANet: learning pixel-wise contextual attention for saliency detection. In: International Conference on computer vision and pattern recognition, pp 3089–3098
29. Liu J, Hou Q, Cheng M, Feng J, Jiang J (2019) A simple pooling-based design for real-time salient object detection. In: International conference on computer vision and pattern recognition
30. Liu Q, Hong X, Zou B, Chen J, Chen Z, Zhao G (2017) Hierarchical contour closure-based holistic salient object detection. IEEE Trans Image Process 26(9):4537–4552
31. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: International Conference on computer vision and pattern recognition, pp 3431–3440
32. Lu X, Wang W, Ma C, Shen J, Shao L, Porikli F (2019) See more, know more: unsupervised video object segmentation with co-attention siamese networks. In: International Conference on computer vision and pattern recognition, pp 3623–3632
33. Margolin R, Zelnik-Manor L, Tal A (2014) How to evaluate foreground maps. In: International conference on computer vision and pattern recognition, pp 248–255
34. Mattyus G, Luo W, Urtasun R (2017) Deep-roadmapper: extracting road topology from aerial images. IEEE Int Conf Comput Vis 10:1109
35. Movahedi V, Elder JH (2010) Design and perceptual validation of performance measures for salient object segmentation. In: International conference on computer vision and pattern recognition, pp 49–56
36. Pan Y, Chen Y, Fu Q, Zhang P, Xu X (2011) Study on the camouflaged target detection method based on 3D convexity. Modern Appl Sci 5(4):152
37. Pang Y, Zhao X, Zhang L, Lu H (2020) Multi-scale interactive network for salient object detection. In: International Conference on computer vision and pattern recognition, pp 9413–9422
38. Qian X, Wang H, Zhao Y, Hou X, Hong R, Wang M, Tang Y (2016) Image location inference by multisaliency enhancement. IEEE Trans Multimedia 19(4):813–821
39. Qin X, Zhang Z, Huang C, Gao C, Dehghan M, Jagersand M (2019) BASNet: boundary-aware salient object detection. In: International Conference on computer vision and pattern recognition, pp 7479–7489
40. Rahman MA, Wang Y (2016) Optimizing intersection-over-union in deep neural networks for image segmentation. In: International Symposium on visual computing, pp 234–244
41. Sengottuvelan P, Wahi A, Shanmugam A (2008) Performance of decamouflaging through exploratory image analysis. In: International conference on emerging trends in engineering and technology, pp 6–10
42. Shi B, Bai X, Belongie S (2017) Detecting oriented text in natural images by linking segments. In: International Conference on computer vision and pattern recognition, pp 2550–2558
43. Skurowski P, Abdulameer H, Blaszczyk J, Depta T, Kornacki A, Koziel P (2018) Animal camouflage analysis: CHAMELEON database Unpublished Manuscript
44. Song L, Geng W (2010) A new camouflage texture evaluation method based on wssim and nature image features. In: 2010 International conference on multimedia technology, pp 1–4
45. Sun L, Yang K, Hu X, Hu W, Wang K (2020) Real-time fusion network for rgb-d semantic segmentation incorporating unexpected obstacle detection for road-driving images. In: IEEE robotics and automation letters
46. Tang Y, Wu X, Bu W (2016) Deeply-supervised recurrent convolutional neural network for the salient object. In: Proceedings of the 2016 ACM on multimedia conference, pp 397–401
47. Wang L, Lu H, Wang Y, Feng M, Wang D, Yin B, Ruan X (2017) Learning to detect salient objects with image-level supervision. In: International conference on computer vision and pattern recognition, pp 136–145
48. Wei J, Wang S, Huang Q (2020) F3Net: fusion, feedback and focus for salient object detection. In: AAAI conference on artificial intelligence
49. Wu Y, Gao S, Mei J, Xu J, Fan D, Zhao C, Cheng M (2020) JCS: an explainable covid-19 diagnosis system by joint classification and segmentation, vol 2004, p 07054
50. Wu Z, Su L, Huang Q (2019) Stacked cross refinement network for edge-aware salient object detection. In: IEEE international conference on computer vision, pp 7264–7273
51. Wu Z, Su L, Huang Q (2019) Cascaded partial decoder for fast and accurate salient object detection. In: International conference on computer vision and pattern recognition, pp 3907–3916
52. Xie S, Tu Z (2015) Holistically-nested edge detection. In: International conference on computer vision and pattern recognition, pp 1395–1403
53. Xu Y, Xu D, Hong X, Ouyang W, Ji R, Xu M, Zhao G (2019) Structured modeling of joint deep feature and prediction refinement for salient object detection. In: IEEE international conference on computer vision, pp 3789–3798
54. Yan Q, Xu L, Shi J, Jia J (2017) Hierarchical saliency detection. In: International conference on computer vision and pattern recognition, pp 1155–1162
55. Yang M, Yu K, Zhang C, Li Z, Yang K (2018) Dense ASPP for semantic segmentation in street scenes. In: International conference on computer vision and pattern recognition, pp 3684–3692
56. Zhang X, Wang T, Qi J, Huchuan L, Wang G (2018) Progressive attention guided recurrent network for salient object detection. In: International conference on computer vision and pattern recognition, pp 714–722
57. Zhang P, Wang D, Lu H, Wang H, Ruan X (2017) Amulet: aggregating multi-level convolutional features for salient object detection. In: IEEE international conference on computer vision, pp 202–211
58. Zhang L, Dai J, Lu H, He Y, Wang G (2018) A bi-directional message passing model for salient object detection. In: International conference on computer vision and pattern recognition, pp 1741–1750
59. Zheng Y, Zhang X, Wang F, Cao T, Sun M, Wang X (2018) Detection of People with camouflage pattern via dense deconvolution network. IEEE Signal Process Lett 26(1):29–33
60. Zhou H, Xie X, Lai J, Chen Z, Yang L (2020) Interactive two-stream decoder for accurate and fast saliency detection. In: International conference on computer vision and pattern recognition, pp 9141–9150
61. Zhu Z, Xu M, Bai S, Huang T, Bai X (2019) Asymmetric non-local neural networks for semantic segmentation. In: IEEE international conference on computer vision, pp 593–602