Frequency Learning Network with Dual-Guidance Calibration for Camouflaged Object Detection

Yilin Zhao, Qing Zhang*, and Yuetong Li

Shanghai Institute of Technology, Shanghai, China

Abstract. Camouflaged object detection (COD), which aims to accurately identify objects that visually blend into surroundings, has attracted increasing interest recently. Existing models usually seek a breakthrough in the RGB domain. However, it is difficult to distinguish the target objects that are visually consistent to the backgrounds in some challenging scenarios. Considering that the frequency components can more effectively capture the details and structures of the image, we rethink the COD task from the perspective of the frequency domain. To this end, we propose a frequency learning network to mine boundary and position cues for prediction. Specifically, we design the frequency feature aggregation module to merge cross-level frequency features, which are then grouped to generate details and position cues by the frequency feature learning module. Subsequently, we propose the frequency-assisted object-boundary calibration module and the dual-guidance feature reasoning module to progressively optimize the dual-guidance cues to help calibrate the camouflaged object feature for high-quality prediction. Quantitative and qualitative experimental results demonstrate that our network outperforms the state-of-the-art COD methods.

Keywords: Camouflaged object detection · Spatial frequency · Guidance cue

1 Introduction

In nature, most wild animals visually blend in with their surroundings to avoid detection by predators, since their colors, textures and patterns are highly similar to the background, making them difficult for the predators to distinguish. Camouflaged Object Detection (COD) is a task that aims to detect and segment objects that are perfectly hidden in their surroundings. It has attracted increasing research interest and has been widely used in many real-world applications, ranging from pest monitoring [?] in agriculture to polyp segmentation [?, ?] and lung infection segmentation [?, ?] in medicine.

Different from other dense pixel prediction tasks (*e.g.*, semantic segmentation, salient object detection), COD is more challenging due to the low contrast

[*] Qing Zhang (zhangqing0329@gmail.com) is the corresponding author. This work is supported by Natural Science Foundation of Shanghai under Grant Nos (21ZR1462600). Code is available at `https://github.com/LinAyq/DCNet`.

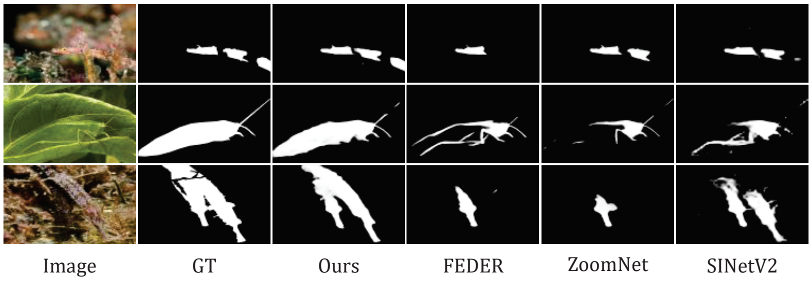

Fig. 1: Visual comparisons of the prediction results generated by the representative models (*i.e.*, ZoomNet [?], SINetV2 [?], and FEDER [?]) in three challenging scenarios including occlusion, indistinguishable boundaries and multiple objects.

between the camouflaged objects and the background, resulting in two issues: the incomplete interior regions and edge disruption. The former is caused by the difficulty in coarsely locating the positions of the potential camouflaged objects; and the latter is attribute to ambiguous object boundaries.

To tackle these challenges, some bio-inspired COD methods [?,?,?,?] are proposed to mimic the human visual perception mechanism or predator behaviors for segmenting the camouflaged objects. Some other methods make efforts to introduce additional guidance cues, such as boundary detection [?,?,?], texture prediction [?] or both [?], to help the network to distinguish the camouflaged objects from the similar backgrounds by performing the multi-task learning. Although these deep learning-based camouflaged object detection methods have achieved excellent performance, it is still quite difficult to accurately segment the entire camouflaged objects while achieving explicit boundaries and effectively suppress the distraction from the confusing background in a simple but effective way, especially in some challenging scenarios such as occlusion, highly uncertain or fuzzy object boundaries, and multiple camouflaged objects. As shown in Fig.1, the existing representative methods can be misled by the visually similar background, leading to low-quality prediction results.

In general, the visual low contrast between the target object and the background makes the identification process in the RGB domain very difficult. Biological studies [?] have shown that the human vision system utilizes different neural pathways to respond to different frequency stimuli, allowing for the processing of specific information. In the image frequency domain analysis, the high-frequency component describes the fine details, while the low-frequency component provides the global structure. Thus we rethink COD in terms of frequency domain to fully exploit and integrate the spatial frequency information to localize the concealed target objects and identify the uncertain boundaries.

To this end, we propose a frequency feature learning network, which solves the COD task exclusively from the perspective of spatial frequency, uses different frequency components to extract details and positions of camouflaged objects, and then calibrates and exploits them to guide the learning of the spatial frequency information. Specifically, in the frequency-aware feature learning stage, the multi-level RGB features from the Transformer encoder are first converted into the spatial frequency features. Then, a frequency feature aggregation (FFA) is designed to merge the cross-level frequency information to obtain enhanced frequency features at different levels, which are divided into two groups according to their intrinsic properties. Subsequently, a frequency feature learning (FFL) module is proposed to excavate the relationships of the intra-group frequency features. In this way, we obtain a boundary cue that provides the detail of the target object as well as a position cue that gives the coarse localization. In the dual-guidance feature calibration stage. The frequency-assisted object-boundary calibration (FOC) module is explored to gradually correct the errors of the dual cues to improve their representation abilities under the help of the spatial frequency features, which is different from the previous works [?,?] that generate the supplementary information at the early stage. Simultaneously, the dual-guidance feature reasoning (DFR) module is proposed to achieve the camouflage object inference by integrating the object-boundary cues and the high-level inferred results, thereby generating the final prediction results.

In summary, the main contributions of this paper are as follows:

- We propose a two-stage network to uniformly identify the camouflaged object from the visually similar background in the spatial frequency domain. The comprehensive experiments demonstrate that our network is competitive against the representative models on the public benchmarks.
- A frequency-aware feature learning stage is designed to integrate the cross-level spatial frequency features and fully exploit the advantages of different frequencies at different levels, thereby obtaining the coarse detail and position cues of the camouflaged objects.
- A dual-guidance feature calibration stage is proposed to progressively calibrate the camouflaged object feature for prediction with the assistance of the boundary and position cues, which are simultaneously optimized to identify the indistinguishable target objects and the subtle details.

2 Related Work

2.1 Camouflaged Object Detection

Recently, compared to the handcrafted models, deep learning-based COD approaches [?, ?, ?, ?] have made great progress. Inspired by the hunting process of predators, Mei *et al.* [?] develop a distraction mining strategy for distraction discovery and removal. To capture the subtle discriminative features, He *et al.* [?] propose to learn an auxiliary edge reconstruction task to promote the precise segmentation. In [?], a boundary-guided network is designed to enforce the network to focus on the object structure and details. Jia *et al.* [?] propose an iterative refinement framework which integrates Segment, Magnify and Reiterate in a multi-stage detection fashion.

Due to the outstanding performance of Transformers originating from the natural language processing, Transformers are introduced to COD [?] for modeling the long-range dependencies. Lyu *et al.* [?] adopt two independent branches that simultaneously perform uncertainty reasoning and edge inference. Huang *et al.* [?] use ViT as the encoder and hierarchically decode locality-enhanced neighboring transformer features through progressive shrinking. Xing *et al.* [?] perform cross-level and adjacent-level feature fusion to obtain rich searched features and fine segmentation. Yang *et al.* [?] designed a hierarchical location guidance module to to locate the potential objects. Although these methods have significantly improved segmentation performance, there still remains several challenging issues regarding uniformly detected interior regions and explicit boundaries and details in complex scenarios.

2.2 Frequency-aware Feature Learning

The human visual spatial frequency model shows that an image can be decomposed into high- and low-spatial frequency parts. Introducing frequency cues will help the model to better identify camouflaged objects from the background. The pioneering work in [?] is to explore learning in the frequency domain for object detection, where the Discrete Cosine Transform (DCT) coefficients are fed into a convolutional neural network (CNN) model for inference. Gueguen et al. [?] trains CNNs directly on DCT coefficients in JPEG codecs to accelerate and improve network performance. Zhong *et al.* [?] introduce frequency cues in COD task and design a frequency enhancement module for dense prediction. In [?], a two-stage framework is proposed to distinguish camouflaged objects by learning high-frequency and low-frequency features, and to obtain the final refined detection results through a progressive refinement mechanism.

In contrast, we attempt to detect and segment the camouflaged objects from the perspective of the spatial frequency domain instead of partially exploiting the frequency information.

3 Methodology

3.1 Overview

In this paper, we propose a frequency learning network based on dual-guidance calibration, called DCNet, to achieve the segmentation results with explicit boundaries from the similar background and interference objects. As shown in Fig. 2, our network consists of a frequency-aware feature learning stage and a dual-guidance feature calibration stage.

Given an RGB image $I \in \mathbb{R}^{H \times W \times 3}$, we adopt Swin Transformer as our backbone to extract multi-level features, denoted as $X_i (i = 1, 2, ..., 4)$. In the frequency-aware feature learning stage, we first transform the RGB features at different levels into the spatial frequency features through the octave convolution [?]. Each feature has its specific characteristic, and the adjacent-level features have similar properties. Inspired by this observation, we first design the

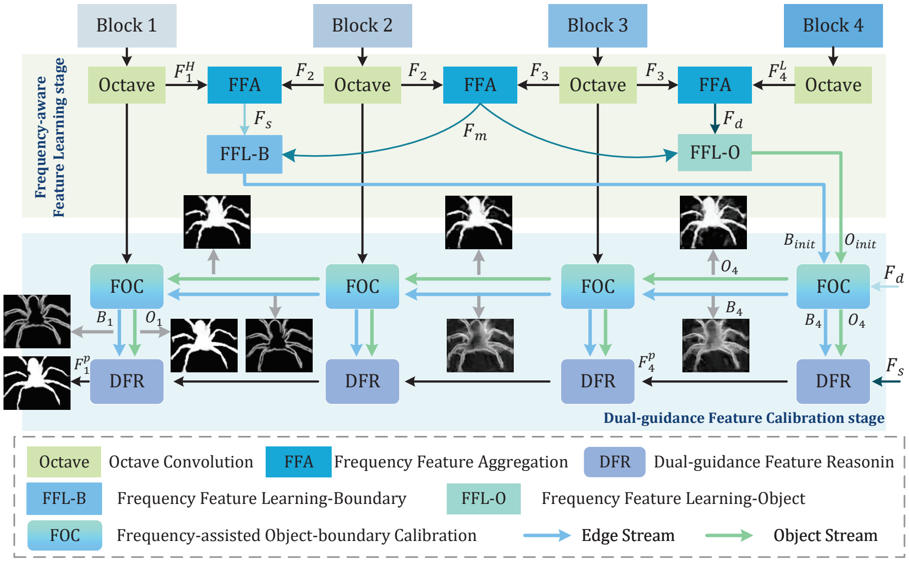
Block 1
Block 2
Block 3
Block 4
Frequency-aware Feature Learning stage
Octave
FFA
F_1^H
F_2
F_3
F_4^L
F_s
F_m
F_d
FFL-B
FFL-O
B_{init}
O_{init}
FOC
DFR
B_1
O_1
B_4
O_4
F_1^p
F_4^p
Dual-guidance Feature Calibration stage
Octave Octave Convolution
FFA Frequency Feature Aggregation
DFR Dual-guidance Feature Reasonin
FFL-B Frequency Feature Learning-Boundary
FFL-O Frequency Feature Learning-Object
FOC Frequency-assisted Object-boundary Calibration
Edge Stream
Object Stream

Fig. 2: The framework of our proposed network DCNet, which consists of a frequency-aware feature learning stage and a dual-guidance feature calibration stage.

frequency feature aggregation (FFA) module to capture the complementary information between the adjacent-level spatial frequency features to enhance the representation ability of frequency features, thus obtaining the enhanced shallow-level, mid-level, and deep-level frequency features F_s, F_m, and F_d, respectively. According to their different intrinsic characteristics, we then design a frequency feature learning module for coarse detail extraction, namely FFL-B, to obtain a boundary feature B_{init}, and use a frequency feature learning module for coarse localization extraction, namely FFL-O, to obtain an object feature O_{init}. Subsequently, in the dual-guidance feature refinement stage, with the guidance of the boundary feature B_i and the object feature O_i, the camouflaged object feature F_i^p are progressively optimized by the frequency-assisted object-boundary calibration (FOC) module and the dual-guidance feature reasoning (DFR) module. Finally, the final prediction result is obtained by our DFR module.

3.2 Spatial Frequency-aware Feature Learning

Different from previous COD models [?,?,?,?] which enhance the feature representation ability in the RGB domain, we perform our network in the frequency perspective. The plug-and-play octave convolution [?] is employed to process low-frequency and high-frequency features separately. In this paper, we integrate them to produce a merged spatial frequency feature.

Specifically, let $X_i = (X_i^H, X_i^L)$ and $F_i = (F_i^H, F_i^L)$ be the input and output tensors, where X_i^H and F_i^H are the high-frequency components, and X_i^L and

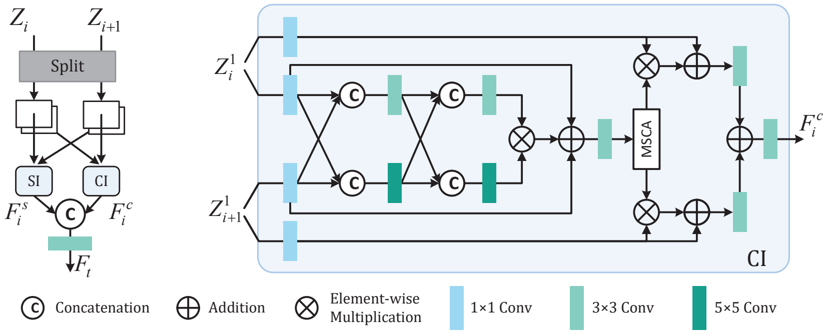

Fig. 3: Illustration of our frequency feature aggregation (FFA) module.

F_i^L are the low-frequency components in the spatial domain. The process of separating RGB features into high-frequency and low-frequency components in octave convolution can be formulated as follows:

$$F_i^H = f(X_i^H; W^{H\to H}) + Up(f(X_i^L; W^{L\to H}), 2) \tag{1}$$

$$F_i^L = f(X_i^L; W^{L\to L}) + f(pool(X_i^H, 2); W^{H\to L}) \tag{2}$$

where $f(X;W)$ denotes a convolution with parameters W, $pool(X,k)$ is an average pooling operation with kernel size $k\times k$ and stride k, $Up(X,k)$ is an upsampling operation by a factor of k via nearest interpolation.

Then we fuse the high-frequency component F_i^H and the low-frequency component F_i^L as a complete frequency representation of spatial domain:

$$F_i = R(F_i^H) + R(F_i^L) \tag{3}$$

where $R(\cdot)$ denotes that the feature resolution is set to a fixed size. In this way, the RGB information from the backbone is converted to spatial frequency domain features $F_i (i = 1, 2, 3, 4)$.

Frequency Feature Aggregation (FFA) Module. To enhance the limited feature representation ability of a single level, we design the FFA module to integrate the cross-level frequency features to obtain a more powerful frequency feature representation. Considering that the adjacent-level features have relatively smaller semantic gap, we implement our FFA module by mining the adjacent-level contextual relationship.

The details of our FFA module is shown in Fig. 3. Given the current-level feature Z_i and the adjacent deep-level feature Z_{i+1} as inputs, our FFA module first equally splits the inputs into two features along the channel dimension, denoted as $\{Z_i^1, Z_i^2\}$ and $\{Z_{i+1}^1, Z_{i+1}^2\}$, respectively. Then, they are regrouped into two feature groups, denoted as $\{Z_i^1, Z_{i+1}^1\}$ and $\{Z_i^2, Z_{i+1}^2\}$, which are fed into the channel-aware integration (CI) block and the spatial-aware integration

(SI) block, respectively, producing complementary features F_i^c and F_i^s. It can be computed as:

$$F_i^c = CI(Z_i^1, Z_{i+1}^1) \tag{4}$$

$$F_i^s = SI(Z_i^2, Z_{i+1}^2) \tag{5}$$

Then they are combined to produce the spatial domain feature F_t as follows:

$$F_t = \mathcal{C}_3([F_i^c, F_i^s]) \tag{6}$$

where $[\cdot]$ is the concatenation operation, $\mathcal{C}_k(\cdot)$ denotes a $k \times k$ convolutional layer, and t is s, m, and d when i is 1, 2, and 3, respectively.

In our CI block shown in Fig. 3, we first perform a 1×1 convolution on each of the two input features, respectively. Then we employ two convolution branches with different kernels for the interaction of spatial features, thus fully perceiving complementary information. The kernel sizes are set to 3 and 5 in our experiments. And then the fused features are fed into an MSCA block [?] to reduce the semantic gap by combining local and global contexts. Finally, the refined input features are merged by an addition operation and a 3×3 convolutional layer. The structure of our SI block is similar to that of the CI block. The difference is that in our SI block, we replace the concatenation operation with the addition operation in two convolution branches.

Note that, since the high-frequency component describes the fine-grained details and the low-frequency component depicts the global structures and layouts, we employ the high-frequency component F_1^H in F_1 and the low-frequency component F_4^L in F_4, instead of the complete frequency features F_1 and F_4, to generate enhanced frequency features F_s and F_d at the shallow and the deep levels, respectively, thereby emphasizing detail and semantic learning. In this way, the different characteristics of the features at different levels are fully exploited.

Frequency Feature Learning (FFL) Module. According to the characteristics of features at different levels, we divide all the enhanced frequency features $\{F_s, F_m, F_d\}$ into two groups, namely the detail group $\{F_s, F_m\}$ and the semantics group $\{F_m, F_d\}$, to take full advantage of different frequency features. Furthermore, we perform different implementations when our FFL module is applied to different frequency feature groups, *i.e.*, FFL-B for mapping the detail group to the boundary cue and FFL-O for injecting the semantics group to the localization cue, respectively, to explore the differences of the inter-group frequency features in a more effective way.

Specifically, as shown in Fig. 4, for the FFL-O module, we first apply a multi-scale receptive field block [?] to each of the input frequency features $\{F_m, F_d\}$ to enlarge the receptive field and capture more contextual information, producing two enhanced features F_m^r and F_d^r, and then they are interacted and merged to generate the object feature O_{init} that provides the positions of the potential camouflaged objects. This process can be formulated as:

$$O_{init} = \mathcal{C}_3(Avg(Up(F_d^r, 2) \otimes F_m^r) + \mathcal{C}_3[Up(F_d^r, 2), F_m^r]) \tag{7}$$

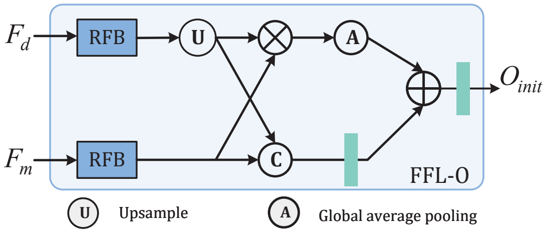

Fig. 4: Illustration of the frequency feature learning module for object (FFL-O).

where $\otimes$ denotes the element-wise multiplication, and $Avg(\cdot)$ is the global average pooling operation which emphasizes the important information from the perspective of the channel dimension.

Meanwhile, the spatial frequency features $\{F_s, F_m\}$ are sent to the FFL-B module to produce boundary information. The detailed illustration of the FFL-B module is similar to that of the FFL-O module. The difference lies in the pooling operation, *i.e.*, we use the global max pooling in our FFL-B module. Global max pooling is employed to emphasize the detected regions with drastic changes, thereby capturing the high-frequency boundary feature representation. It can be expressed as:

$$B_{init} = \mathcal{C}_3(Max(Up(F_m^r, 2) \otimes F_s^r) + \mathcal{C}_3[Up(F_m^r, 2), F_s^r]) \tag{8}$$

where $Max(\cdot)$ denotes the global max pooling operation.

3.3 Dual-guidance Feature Calibration

Our dual-guidance feature calibration stage aims to introduce the coarse boundary feature B_{init}, the coarse object feature O_{init} to help progressively distinguish the camouflaged objects and produce results with explicit boundaries in a dual-path manner, as shown in Fig. 2. It consists of two key components: the frequency-assisted object-boundary calibration (FOC) module for optimizing the object feature and the boundary feature simultaneously, and the dual-guidance feature reasoning (DFR) module for progressively refining the camouflaged object feature under the help of the optimized boundary and object features.

Frequency-assisted Object-boundary Calibration (FOC) Module. We utilize the frequency features to provide additional information to compensate for deficiencies and correct errors in our FOC module, progressively optimizing the object features O_i and the boundary features B_i. Fig. 5 depicts its workflow. The object feature O_{i+1} and the boundary feature B_{i+1} generated from the adjacent higher level, as well as the side-output frequency feature F_i are taken as inputs. We first utilize the frequency feature F_i to guide the learning of the boundary feature and the object feature to focus on the important context and

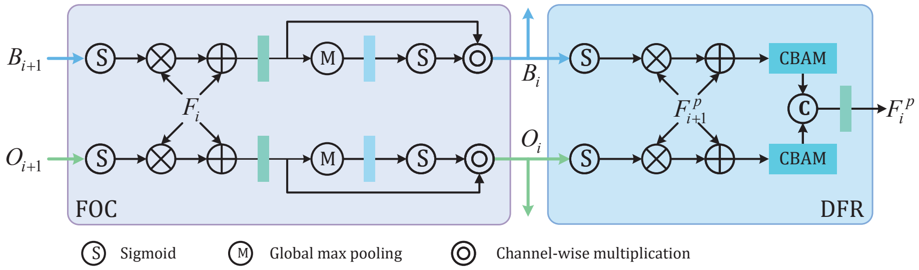

Fig. 5: Illustration of the frequency-assisted object-boundary calibration (FOC) module and the dual-guidance feature reasoning (DFR) module.

to suppress the noise, respectively. This allows the network to take advantage of the frequency feature F_i to provide useful information to enrich the object and boundary features. This process can be formulated as:

$$B_i^a = \mathcal{C}_3(\sigma(B_{i+1}) \otimes F_i + F_i) \tag{9}$$

$$O_i^a = \mathcal{C}_3(\sigma(O_{i+1}) \otimes F_i + F_i) \tag{10}$$

where $\sigma(\cdot)$ is the Sigmoid activation function. A gate function is then used to further refine the boundary and object features by emphasizing the important feature channels, making the network aware of where to learn:

$$B_i = \sigma(\mathcal{C}_1(Max(B_i^a))) \circledcirc B_i^a \tag{11}$$

$$O_i = \sigma(\mathcal{C}_1(Max(O_i^a))) \circledcirc O_i^a \tag{12}$$

where $\circledcirc$ is the channel-wise multiplication. Note that, for the FOC module appending at the top level of our network, the features B_{init} and O_{init} generated by the FFL-B and FFL-O modules are taken as inputs to provide the coarse boundary and localization information, and the deep-level frequency feature F_d is also taken as input to further emphasize the importance of semantics in positioning the camouflaged objects.

Dual-guidance Feature Reasoning (DFR) Module. Since there are semantic gaps between the boundary feature B_i, the object feature O_i and the camouflaged object feature F_{i+1}^p from the adjacent deeper level, we design the DFR module to integrate these features to obtain the current-level camouflaged object feature F_i^p, instead of simply aggregating them by addition, concatenation or multiplication as some previous models [?, ?, ?] do in the decoder, as shown in Fig. 5. Specifically, the object feature O_i and the boundary feature B_i are introduced to excavating the localization and detail cues of the camouflaged object feature F_{i+1}^p. And the attention unit [?] is employed to refine features from the perspective of the spatial and channel dimensions. Thus, we obtain the camouflaged object feature F_i^p at the current level as follows:

$$F_i^p = \mathcal{C}_3([A(\sigma(B_i) \otimes F_{i+1}^p + F_{i+1}^p), A(\sigma(O_i) \otimes F_{i+1}^p + F_{i+1}^p)]) \tag{13}$$

where $A(\cdot)$ is the attention unit [?]. Note that we use the shallow-level frequency feature F_s as input to provide detail information for the DFR module that is appended at the top level of our network.

3.4 Loss Function

In the proposed network, we implement supervisions on the predicted camouflaged object prediction maps P_i generated from the features F_i^p, the object maps S_i generated from the features O_i and the boundary maps E_i generated from the features B_i.

Following previous work [?], we employ a hybrid loss function $\mathcal{L}_{hybrid}$ composed of a weighted BCE loss ℓ_{bce}^{ω} [?] and a weighted IoU loss ℓ_{iou}^{ω} [?] to measure the differences between the predictions P_i and S_i and the ground truth. For the boundary supervision, we use the dice loss $\mathcal{L}_{dice}$ [?]. Therefore, the overall loss of our DCNet can be expressed as:

$$\mathcal{L}_{overall} = \sum_{i=1}^{4} (\mathcal{L}_{hybrid}(P_i, G) + \mathcal{L}_{hybrid}(S_i, G) + \mathcal{L}_{dice}(E_i, G_e)) \tag{14}$$

where G_e is the boundary ground truth, G is the object ground truth.

4 Experiments

4.1 Experimental Settings

Datasets and Evaluation Metrics. To verify the effectiveness of our proposed model, we evaluate our network on three COD public benchmark datasets, including CAMO [?], COD10K [?] and NC4K [?]. CAMO consists of 1000 training images and 250 images for testing. COD10K contains a total of 5066 images, where 3040 images are used for training and 2026 images for evaluation. NC4K is another large-scale COD dataset consisting of 4121 images for testing. Following the previous works [?, ?, ?], we use 3040 images from COD10K and 1000 images from CAMO as the training set.

Four widely used metrics are employed to evaluate the performance, including mean absolute error ($\mathcal{M}$) [?], weighted F-measure (F_β^ω) [?], S-measure (S_m) [?] and E-measure (E_ϕ^m) [?].

Implementation Details. Our proposed network is implemented by PyTorch with an NVIDIA GTX 3090 GPU (24GB memory). We adopt the Swin Transformer as backbone network, and the learning rate is initialized to 5e-5, divided

Table 1: Quantitative comparison with the SOTA methods on three benchmark datasets. Notes ↑ / ↓ denote the larger/smaller is better, respectively. "–" is not available. The top two models are **bolded** and underlined for highlighting, respectively.

Method	**Publication**	**CAMO**				**COD10K**				**NC4K**			
		$S_m \uparrow$	$F_\beta^\omega \uparrow$	$\mathcal{M} \downarrow$	$E_\phi^m \uparrow$	$S_m \uparrow$	$F_\beta^\omega \uparrow$	$\mathcal{M} \downarrow$	$E_\phi^m \uparrow$	$S_m \uparrow$	$F_\beta^\omega \uparrow$	$\mathcal{M} \downarrow$	$E_\phi^m \uparrow$
SINet	20CVPR	0.752	0.606	0.100	0.771	0.771	0.551	0.051	0.807	0.808	0.723	0.058	0.871
PFNet	21CVPR	0.782	0.695	0.085	0.841	0.800	0.660	0.040	0.877	0.829	0.745	0.053	0.887
MGL-R	21CVPR	0.776	0.673	0.088	0.812	0.814	0.666	0.035	0.851	0.833	0.739	0.053	0.867
SLSR	21CVPR	0.787	0.696	0.080	0.838	0.804	0.673	0.037	0.880	0.840	0.766	0.048	0.895
UGTR	21ICCV	0.785	0.686	0.086	0.823	0.818	0.667	0.035	0.853	0.839	0.747	0.052	0.874
BSANet	22AAAI	0.794	0.717	0.079	0.851	0.818	0.699	0.034	0.891	0.842	0.771	0.048	0.897
BGNet	22IJCAI	0.812	0.749	0.073	0.870	0.831	0.722	0.033	0.901	0.851	0.788	0.044	0.907
SINetV2	22TPAMI	0.820	0.743	0.071	0.882	0.815	0.680	0.037	0.887	0.847	0.770	0.048	0.903
FDNet	22CVPR	0.844	0.778	0.062	0.898	0.837	0.731	0.030	0.918	-	-	-	-
SegMAR	22CVPR	0.816	0.753	0.071	0.874	0.833	0.724	0.034	0.899	0.841	0.781	0.046	0.896
ZoomNet	22CVPR	0.820	0.752	0.066	0.877	0.838	0.729	0.029	0.888	0.853	0.784	0.043	0.896
FPNet	23ACMMM	0.851	0.802	0.056	0.905	0.850	0.755	0.028	0.912	-	-	-	-
UEDG	23TMM	0.863	0.817	0.048	0.922	0.858	0.766	0.025	0.924	0.879	0.830	0.035	0.929
FSPNet	23CVPR	0.856	0.799	0.050	0.899	0.851	0.735	0.026	0.895	0.879	0.816	0.035	0.915
FEDER	23CVPR	0.802	0.738	0.071	0.867	0.822	0.716	0.032	0.900	0.847	0.789	0.044	0.907
DINet	24TMM	0.821	0.748	0.068	0.873	0.832	0.724	0.031	0.903	0.856	0.790	0.043	0.909
RISNet	24CVPR	0.870	0.827	0.050	0.922	**0.873**	**0.799**	0.025	**0.931**	**0.882**	**0.834**	0.037	0.925
DCNet	Ours	**0.873**	**0.834**	**0.042**	**0.928**	0.860	0.772	**0.024**	**0.931**	**0.882**	**0.834**	**0.033**	**0.934**

by 10 every 50 epochs. During training, all input images are resized to 384×384 and augmented by random horizontal flipping, cropping, rotation, and color enhancement before being fed into our network. The batch size is set to 16 and the total number of training epochs is set to 100.

4.2 Comparison with State-of-the-art Methods

We compare our DCNet with several representative methods, including SINet [?], PFNet [?], MGL_R [?], SLSR [?], UGTR [?], BSANet [?], BGNet [?], SINetV2 [?], FDNet [?], SegMAR [?], ZoomNet [?], FPNet [?], UEDG [?], FSPNet [?], FEDER [?], DINet [?] and RISNet [?]. For a fair comparison, all the predicted maps are provided by the authors or reproduced by the public released codes.

Table. 1 presents the quantitative results of our proposed method against other competitors. We can observe that our method consistently outperforms competitors on all the benchmark datasets. Specifically, our DCNet ranks first on CAMO and NC4K, and performs slightly worse than the recently published SOTA model RISNet [?] on COD10K, indicating the superiority of our proposed network.

Fig. 6 shows the visual comparisons of our DCNet and some other state-of-the-art methods. From the results we can clearly see that our prediction results precisely locate the camouflaged objects and identify the blurred object boundaries in the challenging scenarios, including small objects (rows 1, 2), medium objects (rows 3, 4), large objects (rows 5, 6), occlusions (row 3) and multiple objects (row 2). For example, in the first row, other methods incorrectly detect the interference as the target object. In contrast, our DCNet can capture the inconspicuous objects. For the large objects, the prediction maps of most methods are structurally incomplete (row 6) or have fuzzy edges (row 5). Due to the

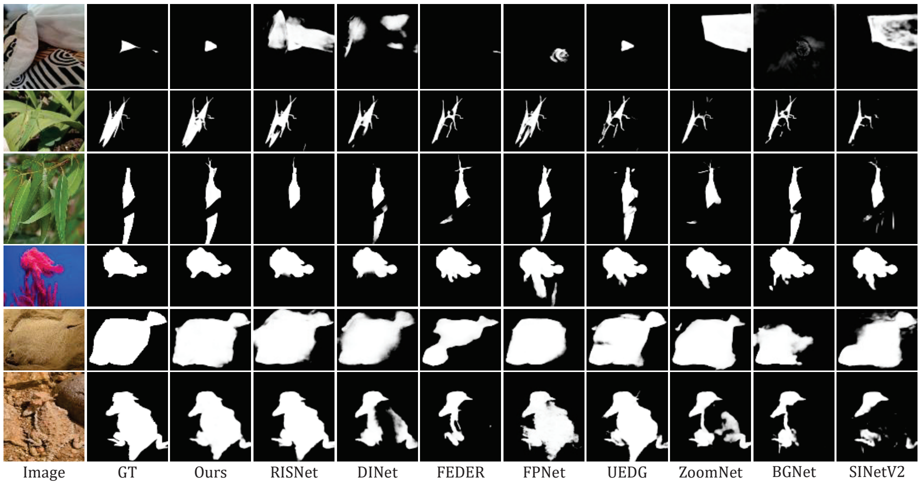

Fig. 6: Visual comparisons of several representative COD methods and our proposed DCNet.

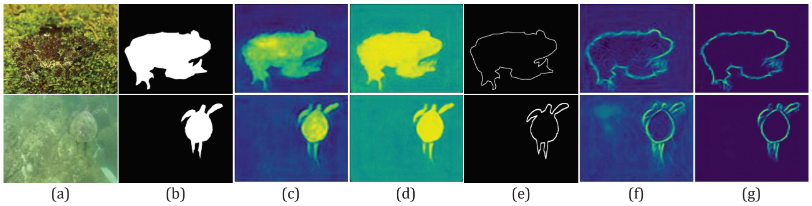

Fig. 7: Visualization of feature maps for FOC. (a) Image, (b) GT of object, (c) O_4, (d) O_1, (e) GT of edge, (f) B_4, (g) B_1. Please zoom in for more details.

integration of edge and structure representations in the frequency domain, our prediction maps have clearer edges and a more complete internal structure.

4.3 Ablation Studies

Effectiveness of Key Components. To verify the effectiveness of the proposed key components, we conduct a series of ablation experiments on large-scale COD10K and NC4K datasets, as shown in Table. 2. We first completely replace all the key components with a concatenation operation and a convolutional layer to construct the baseline, denoted as "B". Then, we gradually add different components to the baseline network, including the frequency feature conversion (*i.e.*, OCT), the FFA module, the FFL module, the FOC module and the DFR module. From Table. 2, we can see that the performance gradually increases, indicating that each proposed component play positive role in our network to achieve accurate segmentation results.

Table 2: Ablation analyses of each component.

No.	Method						COD10K				NC4K			
	B	OCT	FFA	FFL	FOC	DFR	$S_m \uparrow$	$F_\beta^\omega \uparrow$	$\mathcal{M} \downarrow$	$E_\phi^m \uparrow$	$S_m \uparrow$	$F_\beta^\omega \uparrow$	$\mathcal{M} \downarrow$	$E_\phi^m \uparrow$
①	✓						0.836	0.710	0.032	0.904	0.867	0.803	0.039	0.915
②	✓	✓					0.842	0.728	0.030	0.914	0.872	0.811	0.038	0.926
③	✓	✓	✓				0.851	0.738	0.028	0.918	0.875	0.817	0.037	0.926
④	✓	✓	✓	✓			0.851	0.741	0.027	0.919	0.875	0.820	0.036	0.928
⑤	✓	✓	✓	✓	✓		0.853	0.757	0.026	0.925	**0.882**	0.825	0.034	0.931
Ours	✓	✓	✓	✓	✓	✓	**0.860**	**0.772**	**0.024**	**0.931**	**0.882**	**0.834**	**0.033**	**0.934**

Table 3: Ablation analyses of the configurations of our propsoed modules.

Method	COD10K				NC4K			
	$S_m \uparrow$	$F_\beta^\omega \uparrow$	$\mathcal{M} \downarrow$	$E_\phi^m \uparrow$	$S_m \uparrow$	$F_\beta^\omega \uparrow$	$\mathcal{M} \downarrow$	$E_\phi^m \uparrow$
w/o SFL	0.849	0.740	0.028	0.918	0.873	0.817	0.037	0.926
All_B	0.848	0.747	0.027	0.919	0.874	0.818	0.037	0.925
All_O	0.850	0.755	0.026	0.923	0.879	0.828	0.035	0.931
O+B	0.848	0.748	0.027	0.918	0.877	0.821	0.036	0.927
w/o BS	0.846	0.736	0.029	0.911	0.877	0.816	0.036	0.926
w/o OS	0.850	0.745	0.028	0.914	0.878	0.821	0.036	0.926
All_F_s	0.858	0.764	0.025	0.927	0.879	0.830	0.034	0.929
All_F_d	0.859	0.767	0.025	0.927	**0.882**	0.831	0.034	0.932
Ours	**0.860**	**0.772**	**0.024**	**0.931**	**0.882**	**0.834**	**0.033**	**0.934**

Effectiveness of Frequency-aware Feature Learning Different from the previous frequency-based networks for camouflaged object detection, we derive our DCNet entirely from a spatial frequency perspective. To verify the effectiveness of spatial frequency learning, we construct the network "w/o SFL" by removing the octave units that convert the RGB features into the spatial frequency domain features $F_i (i = 1, 2, 3, 4)$. The experimental results are shown in Table. 3. Compared to our network, we observe a significant decrease of the network "w/o SFL" in performance, verifying that the frequency features are suitable for the COD task due to their powerful representation ability of details and structure. The visual comparison as shown in Fig. 8. It can be seen that our spatial frequency-based network uniformly detects the complete target object and suppresses the irrelevant background noise in the complex scenes where the camouflaged objects are visually blended into their surroundings.

In our network, we perform different implementations of the FFL module for the boundary cue and the object cue. To confirm its rationality, we construct three variants, including using FFL-B for dual guidance cues (*i.e.*, All_B), using FFL-O for dual guidance cues (*i.e.*, All_O), and swapping the position of the FFL-B and FFL-O modules (*i.e.*, O+B). As shown in Table. 3, we can see that the performance of the three variants decreases, indicating that our different implementations for the boundary and position cues make full use of the different frequency features.

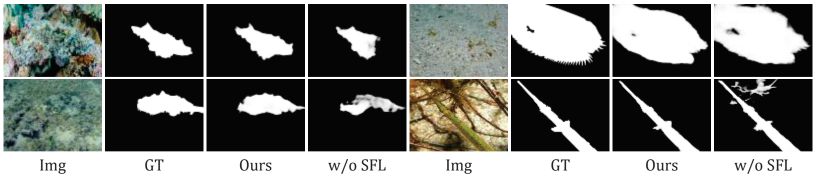

Fig. 8: Visual comparisons of the network "w/o SFL" and our model.

Effectiveness of Dual-guidance Feature Calibration Our FOC module is used to optimize the boundary and object with the assistance of the side-output spatial frequency feature. To show the effect of the FOC more intuitively, we exhibit the visualized features B_1, O_1, B_4 and O_4 in Fig. 7. By comparing O_4 and O_1, we can observe that the interior regions of the target object in features O_1 are highlighted more completely, and by comparing B_4 and B_1, it can be seen that the boundary feature B_1 has more consistent boundaries, confirming that the dual guidance cues are progressively refined by our FOC module.

In addition, we also construct two networks by removing the boundary stream (*i.e.*, w/o BS) and the object stream (*i.e.*, w/o OS), respectively, as shown in Table. 3. Comparing "w/o BS" and "w/o OS" with our network, it can be seen that the networks using merely one guidance cue perform worse than our network, indicating that it is important for the COD task to capture the clear boundaries and complete interiors.

We also provide two variants, including merely sending the feature F_s to FOC and DFR (*i.e.*, All_F_s) and using the feature F_d for FOC and DFR (*i.e.*, All_F_d). As shown in Table. 3, the experimental results indicate that exploiting the advantages of different frequency features contributes to performance improvement.

5 Conclusion

In this paper, we rethink the COD task entirely from the perspective of spatial frequency information to propose a frequency feature network, termed DCNet, for accurate segmentation results. Instead of partially using the frequency information, we fully derive our network using the spatial frequency information. Specifically, our network mainly consists of two parts: a frequency-aware feature learning stage and a dual-guidance feature calibration stage. We first integrate the cross-level frequency features and then group them to explore their intra-group relationships, thus obtaining the boundary and position cues of the camouflaged objects. Subsequently, unlike the previous works that directly generate the guidance cue at the early stage, we gradually refine the guidance cues, which are used to calibrate and enrich the camouflaged object feature for high-quality prediction. The experimental results show that our network exhibits a competitive performance against the state-of-the-art COD methods.