Detecting Camouflaged Object in Frequency Domain

Yijie Zhong[1*] Bo Li[2*†] Lv Tang[*†] Senyun Kuang[3] Shuang Wu[2] Shouhong Ding[2]
[1] Tongji University, [2] Youtu Lab, Tencent, [3] Southwest Jiaotong University
dun.haski@gmail.com, libraboli@tencent.com, luckybird1994@gmail.com
syKuang@my.swjtu.edu.cn, calvinwu@tencent.com, ericshding@tencent.com

Abstract

Camouflaged object detection (COD) aims to identify objects that are perfectly embedded in their environment, which has various downstream applications in fields such as medicine, art, and agriculture. However, it is an extremely challenging task to spot camouflaged objects with the perception ability of human eyes. Hence, we claim that the goal of COD task is not just to mimic the human visual ability in a single RGB domain, but to go beyond the human biological vision. We then introduce the frequency domain as an additional clue to better detect camouflaged objects from backgrounds. To well involve the frequency clues into the CNN models, we present a powerful network with two special components. We first design a novel frequency enhancement module (FEM) to dig clues of camouflaged objects in the frequency domain. It contains the offline discrete cosine transform followed by the learnable enhancement. Then we use a feature alignment to fuse the features from RGB domain and frequency domain. Moreover, to further make full use of the frequency information, we propose the high-order relation module (HOR) to handle the rich fusion feature. Comprehensive experiments on three widely-used COD datasets show the proposed method significantly outperforms other state-of-the-art methods by a large margin.

1. Introduction

With the goal of detecting and segmenting the objects that are perfectly embedded in the environment, camouflaged object detection (COD) has become prevalent in the computer vision community [9, 17, 59]. As a preliminary step, COD plays an essential role in various visual systems, such as polyp segmentation [10], lung infection segmentation [11], and recreational art [3].

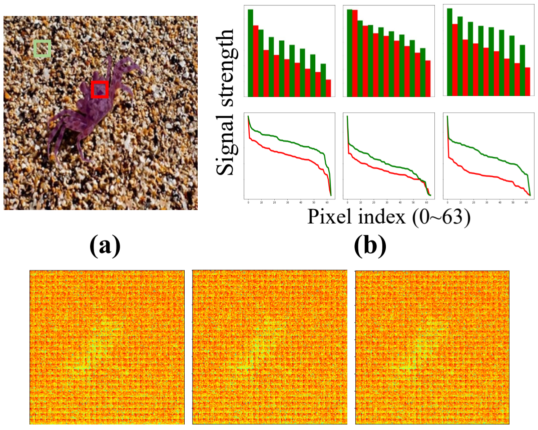
Table 1. The **bolded** numbers represent the best results, and the underline indicates the second best. **We apply the vanilla U-net as the network structure for the U-Net in this table. And it is trained by the commonly used weighted BCE loss and weighed IoU loss.** It shows the competitive performance against the state-of-the-art SINet [9], LSR [33], PFNet [8], and UGTR [58].

Method	SINet	LSR	PFNet	UGTR	**UNet**
COD10K-Test(2026 images) [9]					
$S_\alpha \uparrow$	0.771	0.793	0.800	**0.818**	0.803
$E_\phi \uparrow$	0.806	0.868	**0.877**	0.850	0.873
$F_\beta^w \uparrow$	0.551	0.663	0.660	**0.667**	0.655
$M \downarrow$	0.051	0.041	0.040	**0.035**	0.039
CAMO-Test(250 images) [22]					
$S_\alpha \uparrow$	0.751	**0.793**	0.782	0.785	**0.793**
$E_\phi \uparrow$	0.771	0.826	0.842	**0.859**	0.848
$F_\beta^w \uparrow$	0.606	0.696	0.695	0.686	**0.697**
$M \downarrow$	0.100	0.085	0.085	0.086	**0.081**
CHAMELEON(76 images) [42]					
$S_\alpha \uparrow$	0.869	**0.893**	0.882	0.888	0.883
$E_\phi \uparrow$	0.891	0.928	**0.931**	0.918	0.929
$F_\beta^w \uparrow$	0.740	**0.812**	0.810	0.796	0.806
$M \downarrow$	0.044	0.033	0.033	**0.031**	0.032
Avg. Rank	5	2.6	2.8	2.5	**2.1**

Traditional methods [17, 37, 41] detect camouflaged objects by utilizing handcrafted low-level features, thus these methods often fail in complex scenes. Recently, with the application of deep convolutional neural networks (CNN), the CNN-based methods have pushed the performance of COD to a new level. Some methods [8, 40] make attempts at designing texture enhanced module or adopting attention mechanisms to guide the models to focus on camouflaged regions. Methods try to locate camouflaged objects accurately with the help of extra edge information [59]. In [33], new supervision data is introduced for segmenting the camouflaged objects. Recent works [35] try to treat segmenting camouflaged objects as a two-stage process. Abandoning these sophisticated techniques, we simply use U-Net like networks with Res2Net [13] and ResNet50 [16] backbones,

[*] All authors contributed equally to this paper.

[†] Corresponding authors. This work is supported by the Fundamental Research Funds for the Central Universities.



(a) (b)

(c)

Figure 1. Frequency-aware clues for camouflaged object detection. We apply Discrete Cosine Transform (DCT) in every 8 × 8 patch. (a) the inpur RGB image in which the ground truth region is in dark color; (b) the statistical results of the frequency signal of the selected patch (target object and background); (c) coefficients of the Y, Cb, Cr space after DCT.

to detect camouflaged objects. As can be seen in **Table 1**, compared to existing state-of-the-art (SOTA) methods, only using U-Net network can already achieve competitive performance especially on larger datasets (achieve SOTA performance in 3 metrics), which denotes existing SOTA methods may not well address COD task.

All these SOTA COD methods share one common characteristic: they just reinforce RGB domain information of an image by sophisticated techniques. However, according to the studies of biology and psychology [36], predator frequency-dependent predation makes use of their perceptual filters [21] bound to specific features when separating the target animals from its background. When processing a visual scene, animals have more wavebands than humans, which makes it hard to spot camouflaged objects for human visual system (HVS) [4, 43]. In this research, we claim that the goal of COD task is not just to mimic the human visual ability in single RGB domain, but to go beyond the human biological vision. Hence, for better detecting camouflaged objects from backgrounds, some other clues in image are needed (*e.g.*, clues in frequency domain).

As described in the previous work [51], CNN has the potential to exploit the various frequency image components that are not perceivable to humans. The first problem that this paper addresses is how to involve frequency-aware clues into the CNN models. To learn more statistical information and to enhance clues about camouflaged objects in frequency domain, we design a frequency enhancement module (FEM). It consists of an offline discrete cosine transform and an online learnable enhancement followed by the feature alignment to fuse features from both RGB and frequency domains. Moreover, we propose a novel frequency loss to directly constrain in frequency and guide the network to focus more on the frequency signals. As can be seen in Figure 1(a), "red box" means target object and "green box" denotes background. The target object is cryptic in the background. In RGB domain, the target object is hard to see. However, in frequency domain Figure 1(c), information that can help distinguish target object and background is captured. When there are noise objects in the image, they may be extracted together with the camouflaged objects. In order to distinguish the real camouflaged objects, we propose the high-order relation module (HOR). As the target and noise objects always share similar structural information, a low-order relation is not sufficient for obtaining the discriminative features.

The main contributions are summarized as follows:

- To our best knowledge, we are the first to claim COD task should go beyond RGB domain and introduce frequency clues to better detecting camouflaged objects.
- We present a powerful network for COD task with enhanced frequency clues. And we design a Frequency Enhancement Module (FEM) with a frequency perceptual loss and a high-order relation module (HOR) to better leverage the information in frequency domain for dense prediction task.
- Comprehensive experiments on three widely-used COD datasets (CHAMELEON, CAMO-Test and COD10-Test) show that the proposed method outperforms other state-of-the-art methods by a large margin.

2. Related Work

2.1. Camouflaged object detection

The camouflaged object detection (COD) task [23, 33, 35] has posed new challenges by pushing the boundaries of generic / salient object detection [15, 28, 29] to concealed objects blending in with their surroundings. Fan *et al.* [9] present the SINet to address this challenge by first roughly searching for camouflaged objects and then performing segmentation. Yan *et al.* [57] introduce MirrorNet to use both instance segmentation and adversarial attack for COD. Recently, Zhai *et al.* [59] propose a graph-based model to simultaneously perform camouflaged object detection and the camouflaged object-aware edge extraction by comprehensively reasoning about multi-level relations. [40] considers the subtle texture difference between camouflaged objects and the background. Unlike previous works, our novelty is that we introduce the frequency domain information to boost the performance of the COD task. Using textures, boundaries, etc. as clues may fail to detect camouflaged objects in complex situations. Because these information is the same as these observed by human vision system and can easily be deceived or misled.

2.2. Salient object detection

Salient object detection (SOD) aims to identify the most attention-grabbing objects in an image and then segment their pixel-level silhouettes [19,31,47–49,60,64]. Hundreds of image-based SOD methods have been proposed in the past decades [6,24–27,46]. Early methods are mainly based on the handcrafted low-level features as well as heuristic priors. Recently, deep convolutional neural networks have set new state-of-the-art on salient object detection. Due to the effectiveness of feature enhancement, attention mechanisms [50,54] are applied to saliency detection [2]. In addition, edge/boundary cues are leveraged to refine the saliency map [38,44]. However, applying the SOD approaches for camouflaged object segmentation may not be appropriate as the term "salient" is essentially the opposite of "camouflaged" (*standout vs. immersion*).

2.3. Learning in the frequency domain

Compressed representations in the frequency domain contain rich patterns for image understanding tasks. [14] extracts features from the frequency domain to classify images. [5] proposes a model conversion algorithm to convert the spatial-domain CNN models to the frequency domain. [56] avoids the complex model transition procedure and uses the SE-Block to select the frequency channels. [39] designs a frequency channel attention network. Despite the achievements of previous methods in frequency domain, how to model the interaction relationship between frequency domain and RGB domain for dense prediction is barely explored. Different from the previous works, we design a learnable enhancement module, and align the RGB domain and frequency domain. Thus, our method can better leverage the rich information from different domains.

3. Method

3.1. Network overview

Figure 2 illustrates the proposed network. The RGB input is transformed to the frequency domain and strengthened by the frequency enhancement module (FEM). Then the RGB and frequency input are fed into the network in RGB flow and frequency flow seperately. The feature alignment (FA) is used to fuse these features from RGB and frequency domains. To find more slight differences within the features to distinguish the camouflaged objects, the high-order relation module (HOR) is built in the main network. Let $x^{rgb} \in \mathbb{R}^{H\times W\times 3}$ denote the RGB input, where H, W are the height and width of the image. And the feature maps from the last residual block of each layer of the backbone can be considered as $\{X^1, X^2, X^3, X^4\}$. Then all these feature maps are processed in the skip connection and decoded in a bottom-up manner. Each decode block is consist of two convolution layers followed by BN and ReLU.

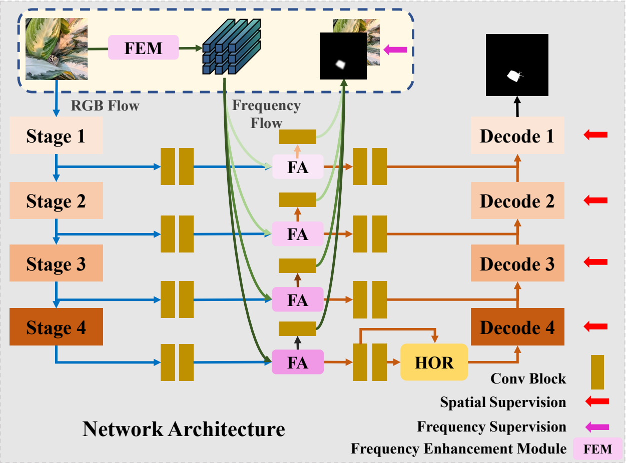


Figure 2. An overview of the proposed network.

3.2. Frequency enhancement module

Offline Discrete Cosine Transform. In this part, the input RGB image is firstly processed by DCT to utilize the frequency information. x^{rgb} is transformed to YCbCr space (denoted by $x^{ycbcr} \in \mathbb{R}^{H\times W\times 3}$). Then, we can obtain $\{p^c_{i,j} | 1 \leq i,j \leq \frac{H}{8}\}$ by dividing x^{ycbcr} into a set of 8×8 patches (taking DCT densely on slide windows of the image is a common operation for frequency processing like JPEG compression). $p^c_{i,j} \in \mathbb{R}^{8\times 8}$ denotes the patch of a certain color channel. Each patch is processed by DCT into frequency spectrum $d^c_{i,j} \in \mathbb{R}^{8\times 8}$, where each value corresponds to the intensity of a certain frequency band. To group all components of the same frequency into one channel, we flatten the frequency spectrum and reshape them to form a new input, following the patch index: $x^{freq}_o = x^{freq}_{i,j} = flatten(d_{i,j})$, where $x^{freq}_o \in \mathbb{R}^{\frac{H}{8}\times\frac{W}{8}\times 192}$, and $d_{i,j} \in \mathbb{R}^{8\times 8\times 3}$ denotes the concatenation of all the $d^c_{i,j}$. In this way, we rearrange the signals which are in zigzag order in one patch and each channel of x^{freq}_o belongs to one band. Thus, the original color input is transformed to the frequency domain.

Online learnable enhancement. Figure 3 depicts the frequency domain transformation process, in which the images are mapped into the frequency domain and enhanced by a learnable module to discover the cues of camouflage objects hidden in frequency space. In practice, there are a variety of camouflaged objects and complicated backgrounds, the fixed offline DCT may not handle this well. We also need an adaptive learning process to adapt to complex scenarios. Since information will be lost during the pre-processing such as JPEG compression. We need to strengthen the fre-

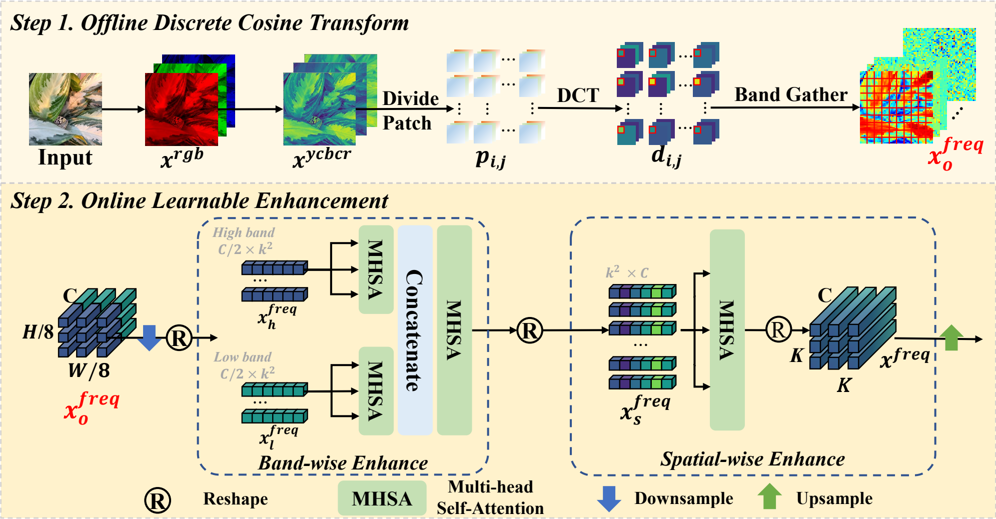


Figure 3. Our proposed FEM contains two steps: an offline DCT process and an online enhancement with neural networks.

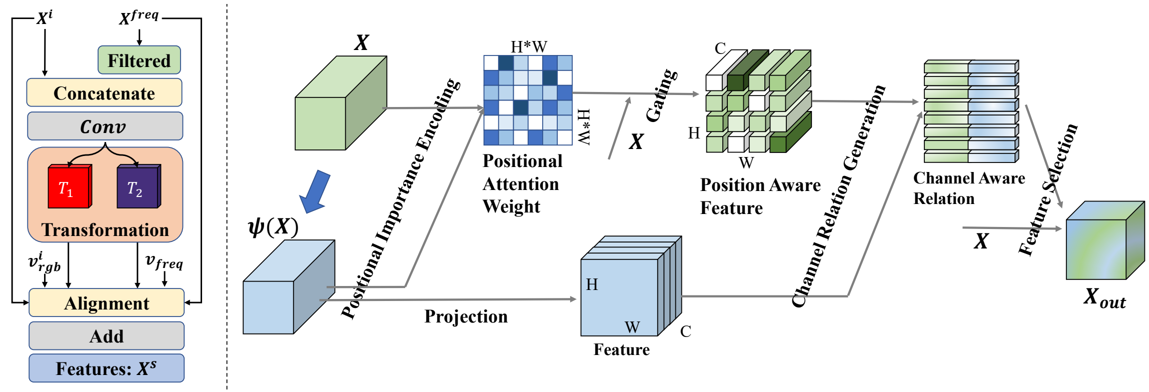
quency signals. Thus, we introduce online learnable enhancement to increase the adaptability of signals.

We build the enhancement module from both within individual patch and between patches. Following the traditional methods [45], we first enhance the coefficients in local frequency bands. We downsample and partition the signals into two parts, the low x_l^{freq} and high signals $x_h^{freq} \in \mathbb{R}^{96\times k^2}$, where k means the size. To boost the signals in the corresponding frequency bands, we feed them into two multi-head self-attention (MHSA) [50] separately and concatenate their output to recover the original shape. Then another MHSA reconciles all the different frequency bands, and the newly formed signal denotes x_f^{freq}. The MHSA is able to capture the rich correlation between each item in the input features. At this point, the different frequency spectrums of the image are fully interacted with. As for DCT, patches are independent of each other, the above procedure only enhances a single patch. To help the network identify the location of the camouflaged object, we need to establish connections between patches. So we first reshape x_f^{freq} to $x_s^{freq} \in \mathbb{R}^{k^2 \times C}$. Then we use MHSA to model the relationships among all the patches. Finally, we can upsample and get the enhanced frequency signals x^{freq}. Both x^{rgb} and x^{freq} are fed into the network. As we apply single layer MHSA in each place and the size of the frequency signals is in a small scale, it will not bring high computational cost.

Feature alignment. We introduce the frequency information to help distinguish the camouflaged objects from the background or interference objects. We should build another module to fuse the features from RGB domain and signal domain well as they are misaligned, as shown in Figure 4(a). The feature alignment is a mutually reinforcing process. The frequency features are discriminative for camouflaged objects. The RGB features have a larger receptive field, and can compensate for the frequency features. Since the previous processing ensures that x^{rgb} and x^{freq} are spatially aligned, we only align the frequency domain with the RGB domain in this part.

As the CNN models are more sensitive to low-frequency channels, we first apply a filter to extract the useful part X^{freq} from x^{freq} for COD. According to the visualization in Figure 1, we can see that the differences at higher frequencies can help to find the camouflage objects. We design a binary base filter f_{base} that covers the high frequency bands, and add three learnable filters $\{f_i\}_{i=1}^3$ for the Y, Cb, Cr color space. The filtering is a dot-product between the frequency response and the combined filters $f_{base} + \sigma(f_i)$, where $\sigma(y) = \frac{1-exp(-y)}{1+exp(-y)}$. For an input frequency domain feature x^{freq}, the network can focus on the most important spectrum automatically by: $X_i^{freq} = x_i^{freq} \odot [f_{base} + \sigma(f_i)]$, where $\odot$ is the element-wise product. Finally, we put them back together: $X^{freq} = Concat([X_1^{freq}, X_2^{freq}, X_3^{freq}])$.

Then, we calculate the transformation for the two signal from the spatial domain and frequency domain. As X^i has different sizes, X^{freq} needs to be scaled to its corresponding size. We concatenate X^i and X^{freq}, then feed it into a *Conv* layer with $4n$ output channels, whose output is T. We take $T^j \in \mathbb{R}^{H\times W\times n} (j = 1, 2, 3, 4)$ out of the third dimension, and reshape them to $HW \times n$. Thus, we obtain the fusion matrix $T_1 \in \mathbb{R}^{HW\times HW}$ for RGB domain, and T_2 for frequency domain by: $T_1 = T^1 (T^2)^T, T_2 = T^3 (T^4)^T$. Secondly, we can align the feature maps. Multiplied with the transformation and a learned vector $v \in \mathbb{R}^{1\times C}$ to adjust the intensity of each channel, the aligned feature of each



(a) Feature Alignment

(b) High Order Relation Module

Figure 4. Illustrations of the feature alignment and high-order relation module. a) Feature Alignment: fusing the features from RGB domain and frequency domain. b) HOR: joint positional and channel-relation, selecting the semantic channels and frequency bands.

domain can be defined as:

$$\begin{aligned} X^i_{rbg2s} &= T_1 X^i \otimes v^i_{rgb}, \\ X_{freq2s} &= T_2 X^{freq} \otimes v_{freq}. \end{aligned} \tag{1}$$

Finally, we can obtain the fused features by adding the two domain features: $X^i_s = X^i_{rbg2s} + X_{freq2s}$. In this way, we can make use of the discriminative frequency information to find the camouflaged objects, while maintaining the CNN clues to ensure the integrity and details of the objects.

Frequency perception loss. To further capture the frequencies that differ from human perception, we introduce a novel loss to constrain the network. Besides calculating losses directly in the RGB domain, we also intend to provide supervision of the network in the frequency domain. On the one hand the commonly used losses may not produce effective guidance for the network in the frequency domain and can lead to the loss of key clues. On the other hand, we assume that the predictions should be correct not only at each pixel location, but also in the coefficients after DCT when they act on the original images.

As DCT is a patch-based operation, we may get coarse predictions here, which mainly focus on the localization of the camouflaged objects. Depart from using a pixel-loss, we compute loss in the frequency domain following DCT, and the network can be guided to mine more information in the frequency domain. Given the input RGB image x, the corresponding ground truth mask M, and the prediction mask Y, we can define the loss as follows:

$$\mathcal{L}_f(Y, M, x) = ||\mathbf{DCT}(x \otimes Y) - \mathbf{DCT}(x \otimes M)||^2_2 / q, \tag{2}$$

where q is the quantization table and $\otimes$ means the element-wise product. Especially, Y and M will first be copied and expanded to the same size as x.

3.3. High order frequency channel selection

With the help of the frequency domain information, we can already improve the performance of network via the invisible clues. However, if we intend to better distinguish camouflaged objects from other non-camouflaged objects, we need to dig deeply into the relations between different pixels in X^i_s. Specifically, the true camouflaged and interfering objects can be separated from the background together with the help of the frequency domain information. However, true camouflaged and interfering objects often share extremely similar structural information and frequency domain clues would hardly distinguish the slight differences. An intuitive method is to introduce attention mechanism(e.g. commonly used Self-Attention Module [52]) to exploring the relationship of different pixels within the feature X^i_s, which may help distinguish the slight differences. However, common used attention mechanism can only capture low-order relation, and it is not enough to spot such subtle differences. Consequently, we propose a high-order relation module (HOR) to address this problem.

Thus, we propose the high-order relation module (HOR) to make the most use of information in the frequency signals, as shown in Figure 4(b). The structural relations are constructed by employing a position-aware gating operation, providing high-order spatial enhancement for further channel interactions and discriminant spectrum selection.

Let $X \in \mathbb{R}^{C \times H \times W}$ denote the input feature, and we first reshape it to $C \times HW$. As frequency responses come from a local region, encoding original features with positional importance is thus necessary to distinguish the camouflaged objects from other objects. The positional attention weights can be represented as:

$$W = \text{softmax}(X^T \psi(X)) \in \mathbb{R}^{HW \times HW}. \tag{3}$$

In addition, different network layers present potential information in different scales, where the latter one has a larger receptive field. Leveraging cross-layer semantics also enhances the representation of multi-scale learning. Here $\psi(X)$ denotes the latter layer than X. Thus W serves as an attention weight to find the RGB and frequency response correlations across different layers. The positional weights then strengthen the original feature and subsequently pass an adaptive gating operation to select the most useful features when occurring different samples:

$$A = \mathcal{G}(W) \cdot (WX^T) + X, \tag{4}$$

where $\mathcal{G}(W) \in \mathbb{R}^{HW}$ denotes the gating weight generated by a FC layer and it can be considered as the function $\mathcal{G} : \mathbb{R}^{HW} \rightarrow \mathbb{R}^1$. The gating operation is generated based on the spatial perceptions to form the positional-aware features.

The Non-local attention is the most relevant to our module. However, it can be described implicitly, using a reweighting mechanism for each channel. This attention mechanism can be regarded as denoising or high-pass filtering operations. PFNet [35] uses two such modules consecutively for channel and spatial. And that makes them independent for each other. Similarly, although the feature A is maintained in its original shape $\mathbb{R}^{H\times W\times C}$, the relationship matrix across different semantic channels and frequency bands is omitted. Thus we propose to generate the rich relation-aware representation subsequently. After obtaining the positional enhanced feature A, the channel-aware relation matrix can be built by similar operations:

$$H = \text{softmax}(A^T\psi(X))) \in \mathbb{R}^{C\times C}, \tag{5}$$

where C denotes the channel dimension of positional-aware features. Each tensor in the channel-aware relation has the same C-dimensions for semantic and frequency mappings which are corresponded to the original feature channels and spectrum. Finally, we apply this relation matrix to X to get the selected information beneficial to camouflage objects: $X_{out} = reshape(HX) \in \mathbb{R}^{H\times W\times C}$. The feature X_{out} is then fed into the decoding process.

3.4. Supervision

As can be seen in Figure 2, let $\{D_1, D_2, D_3, D_4\}$ denote the features extracted from each stage of the decode block. We make four predictions $\{P_i\}_{i=1}^4$ under different resolutions in our network and $\{Y_i\}_{i=1}^4$ from the convolution layer after each FA. Each P_i and Y_i is first rescaled to the input image size. We supervise the network in the frequency domain by the frequency perception loss $\mathcal{L}_f$.

We also provide a supervision in the common RGB domain to ensure details. Following [8], we combine the weighted BCE loss ℓ_{bce} and weighted IoU loss ℓ_{iou} [53] to focus more on the distraction region. The loss function is defined as:

$$\mathcal{L}_i = \mathcal{L}_{bce}(P_i, M) + \mathcal{L}_{iou}(P_i, M) + \mathcal{L}_f(Y_i, M, x^{rgb}), \tag{6}$$

where M means the ground truth label and i denotes the i-th stage of the network. Finally, the overall loss function is:

$$\mathcal{L}_{overall} = \sum_{i=1}^{4} 2^{(1-i)}\mathcal{L}_i. \tag{7}$$

4. Experiment

4.1. Experimental setup

Datasets. We evaluate our method on three benchmark datasets: CHAMELEON [42], CAMO [22], COD10K [9]. CHAMELEON [42] has 76 images. CAMO [22] contains 1,250 camouflaged images covering different categories, which are divided into 1,000 training images and 250 testing images. COD10K [9] is currently the largest benchmark dataset, which includes 3,040 images for training and 2,026 for testing. Our training set is a combination of the train sets from CAMO and COD10K following work [9]. NC4K dataset [33] is also widely used for evaluation in camouflaged object detection.

Evaluation metrics. We use four widely used and standard metrics to evaluate our method: structure-measure (S_α) [7], mean E-measure (E_ϕ) [12], weighted F-measure F_β^w [34], and mean absolute error (M).

Implementation details. We use the PyTorch framework to implement our method. If not specially mentioned, we apply the Res2Net [13] as the backbone. We also train a model with ResNet50 for comparing with other methods which use the same backbone. We use the Adam [20] optimizer with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The weight decay is set to 5e-4 for loss optimization. The learning rate is initialized to 1e-4. It drops to half at 20 epochs and is set to 1e-5 after 40 epochs (overall 100 epochs). During the training stage, the batch size is set to 32. For the data augmentation, we perform simple random cropping and flipping. The image is finally resized to 416×416 followed by a color distortion.

Compared Methods. Here we compare our network with 19 state-of-the-art methods. It contains multiple models for different tasks: object detection method FPN [30]; semantic segmentation method PSPNet [61]; instance segmentation methods Mask RCNN [15], HTC [1], and MSRCNN [18]; medical image segmentation methods UNet++ [65] and PraNet [10]; salient object detection methods PiCANet [32], BASNet [38], CPD [55], PFANet [63], and EGNet [62]; and camouflaged object segmentation methods SINet [9], SINet-V2 [8], LSR [33], PFNet [35], R-MGL [59], JCOD [23] and UGTR [58]. For fair comparison, all the prediction maps of the above methods are either provided by the public websites or produced by retraining the models with open source codes. Besides, all the prediction maps are evaluated with the same code.

Table 2. Comparisons of our proposed method and other 18 state-of-the-art methods in the relevant fields on three benchmark datasets. Larger S_α, E_ϕ, and F_β^w, smaller M correspond to better performance. For the Res2Net backbone, the best results are marked in **bold**. For ResNet50, the best three results are in red, blue, and green fonts.

Methods	COD10K-Test(2026 images)				CAMO-Test (250 images)				CHAMELEON(76 images)			
	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$
FPN	0.697	0.691	0.411	0.075	0.684	0.677	0.483	0.131	0.794	0.783	0.590	0.075
PSPNet	0.678	0.680	0.377	0.080	0.663	0.659	0.455	0.139	0.773	0.758	0.555	0.085
Mask RCNN	0.613	0.748	0.402	0.080	0.574	0.715	0.430	0.151	0.643	0.778	0.518	0.099
UNet++	0.623	0.672	0.350	0.086	0.599	0.653	0.392	0.149	0.695	0.762	0.501	0.094
PiCANet	0.649	0.643	0.322	0.090	0.609	0.584	0.356	0.156	0.769	0.749	0.536	0.085
HTC	0.548	0.520	0.221	0.088	0.476	0.442	0.174	0.172	0.517	0.489	0.204	0.129
MSRCNN	0.641	0.706	0.419	0.073	0.617	0.669	0.454	0.133	0.637	0.686	0.443	0.091
BASNet	0.634	0.678	0.365	0.105	0.618	0.661	0.413	0.159	0.687	0.721	0.474	0.118
CPD	0.747	0.770	0.508	0.059	0.726	0.729	0.550	0.115	0.853	0.866	0.706	0.052
PFANet	0.636	0.618	0.286	0.128	0.659	0.622	0.391	0.172	0.679	0.648	0.378	0.144
EGNet	0.737	0.779	0.509	0.056	0.732	0.768	0.583	0.104	0.848	0.870	0.702	0.050
PraNet	0.789	0.861	0.629	0.045	0.769	0.824	0.663	0.094	0.860	0.907	0.763	0.044
SINet	0.771	0.806	0.551	0.051	0.751	0.771	0.606	0.100	0.869	0.891	0.740	0.044
LSR	0.793	0.868	0.663	0.041	0.793	0.826	0.696	0.085	0.892	0.928	0.812	0.033
PFNet	0.800	0.877	0.660	0.040	0.782	0.842	0.695	0.085	0.882	0.931	0.810	0.033
R-MGL	0.814	0.852	0.666	0.035	0.775	0.812	0.673	0.088	0.892	0.918	0.813	0.030
JCOD	0.809	0.884	0.684	0.035	0.800	0.859	0.728	0.073	0.891	0.943	0.817	0.030
UGTR	0.818	0.850	0.667	0.035	0.785	0.859	0.686	0.086	0.888	0.918	0.796	0.031
Ours-R50	0.833	0.907	0.711	0.033	0.828	0.884	0.747	0.069	0.894	0.950	0.819	0.030
SInet-V2	0.815	0.887	0.680	0.037	0.820	0.882	0.743	0.070	0.888	0.942	0.816	0.030
Ours-R2N	**0.837**	**0.918**	**0.731**	**0.030**	**0.844**	**0.898**	**0.778**	**0.062**	**0.898**	**0.949**	**0.837**	**0.027**

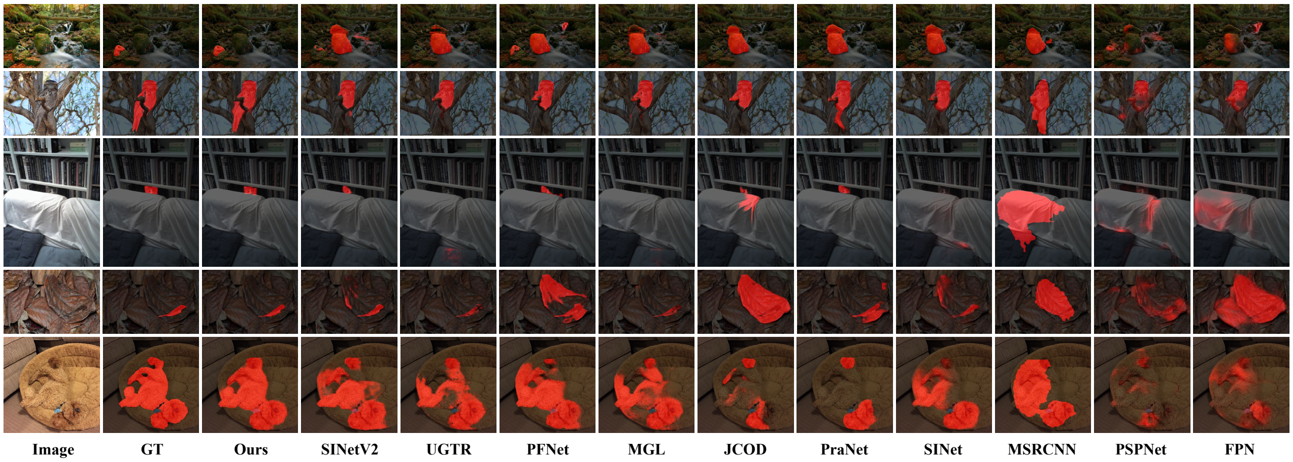
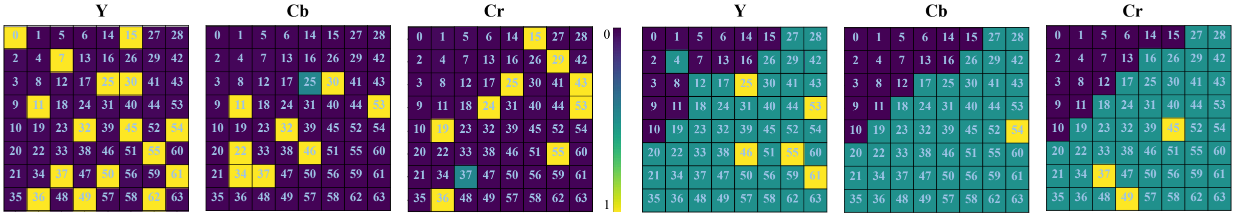


Figure 5. Visual comparisons of camouflaged object detection maps produced by the state-of-the-art methods. Our method can better identify camouflaged objects than all the compared approaches.

4.2. Comparisons with the state-of-the-arts

Quantitative results. For quantitative evaluations, we report four popular metrics in Table 2. The baseline is the vanilla U-Net with weighted BCE loss and weighted IoU loss. Note that no matter which backbone network is applied, our network achieves the competitive performance across these datasets.

Visual comparisons. In Figure 5, we provide challenging examples. Compared with other methods, our method achieves more competitive visual performance mainly in the following aspects. *(a) More accurate camouflaged object localization.* Our method can detect the camouflaged object more completely and accurately. When the camouflaged objects share similar appearance with the background, we can also easily find them with the help of the discriminative frequency information. *(b) Stronger noise object suppression.* Our method can address more complex background interference, such as salient but non-camouflaged regions.



Frequency filter without base filter

Frequency filter with base filter

Figure 6. Heat maps of Y, Cb, and Cr components from the models. The left does not use the base filter f_{base}, and the network is free to learn which bands to focus on. For the right, we give the base filter to make it more concerned with the high frequency (bands greater than 16 followed by [56]).

Table 3. Quantitative results of ablation studies. "Freq. Info." represents *frequency information*. *Off.* and *On.* mean the two steps of the FEM separately.

Models	Freq. Info.	CHAMELEON(76 images)				CAMO-Test(250 images)				COD10K-Test(2026 images)			
		$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$
Baseline	-	0.884	0.931	0.809	0.032	0.820	0.879	0.743	0.072	0.815	0.884	0.680	0.037
+HOR	-	0.888	0.932	0.815	0.031	0.824	0.884	0.750	0.070	0.819	0.889	0.690	0.036
Baseline with frequency	*Off.*	0.886	0.930	0.812	0.031	0.826	0.877	0.744	0.071	0.818	0.890	0.685	0.036
+FEM+$\mathcal{L}_f$	*Off.+On.*	0.887	0.934	0.826	0.030	0.831	0.884	0.756	0.069	0.831	0.902	0.721	0.033
+FEM+$\mathcal{L}_f$+FA	*Off.+On.*	0.891	0.941	0.829	0.029	0.834	0.890	0.768	0.067	0.834	0.903	0.727	0.031
+FEM+$\mathcal{L}_f$+FA+SelfAttention	*Off.+On.*	0.894	0.943	0.833	0.029	0.837	0.893	0.774	0.064	0.836	0.906	0.729	0.031
+FEM+$\mathcal{L}_f$+FA+HOR(Ours)	*Off.+On.*	**0.898**	**0.949**	**0.837**	**0.027**	**0.844**	**0.898**	**0.778**	**0.062**	**0.837**	**0.918**	**0.731**	**0.030**

Some scenes have many distinct objects that are easy to spot, and the target object is hidden in them. We need to suppress the noisy objects that are not part of the camouflaged objects. Only our method can effectively highlight the camouflaged objects and suppress the interference. Moreover, our method shows superiority on account of digging the slight difference between the camouflaged objects and other regions.

4.3. Ablation study

Visualization of the frequency filters. Firstly, we explore which frequency bands are more effective for COD. We train the model without a base filter. Figure 6 left side shows the heat maps of the selection spectrum by the learnable filters. Following [56], we consider the bands greater than 16 as the high spectrum signals. The most frequency bands have a low response and the high spectrum is more important in the heat maps. Secondly, we train the model with the base filter f_{base}. In this way, we explicitly tell the network to focus on higher frequency information. As shown in the right side of Figure 6, the network can further find a smaller number of specific, discriminative frequency bands.

Importance of proposed modules to our network. To study this problem, we removed each module in turn. In Table 3, the comparisons between the results of models already show the effectiveness of our proposed Frequency enhancement module with $\mathcal{L}_f$, feature alignment, and high-order relation generation module. From line.2 and line.7, we could find that the network performs better benefiting from HOR. However, simply applying the attention-like model cannot achieve the full performance without the help of frequency information. From line.3, it can be seen that directly adding the frequency signals to the network without other processing has limited benefit. By comparing the results of models in line.4 and line.5, we observe that fuse the features from two domains can make more use of the frequency domain information. From line.6, it shows that only building the low-order relation by self-attention module is suboptimal compared to our HOR.

5. Conclusion

In this paper, we utilize frequency information of an image to help detect camouflaged objects. By strengthening the coefficients in all the frequency bands with the frequency enhancement module, we can extract the discriminative cues. We further align the spatial-domain (RGB) and the frequency domain to get the fusion features. Besides, by establishing the high order relationships within the intra-image features, we can suppress the background and find the true target objects. Experiments demonstrate that our proposed network achieves better performance than state-of-the-art COD methods on three benchmarks. Comprehensive ablation studies also validate our contributions. This work will benefit researchers exploring the potential of utilizing different frequency clues in various areas of computer vision community.

References

[1] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. Hybrid task cascade for instance segmentation. In *CVPR*, pages 4974–4983, 2019.

[2] Shuhan Chen, Xiuli Tan, Ben Wang, and Xuelong Hu. Reverse attention for salient object detection. In *ECCV*, volume 11213, pages 236–252, 2018.

[3] Hung-Kuo Chu, Wei-Hsin Hsu, Niloy J. Mitra, Daniel Cohen-Or, Tien-Tsin Wong, and Tong-Yee Lee. Camouflage images. *TOG*, 29(4):51:1–51:8, 2010.

[4] I. C. Cuthill. Camouflage. *Journal of Zoology*, 308, 2019.

[5] Max Ehrlich and Larry Davis. Deep residual learning in the JPEG transform domain. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 3483–3492. IEEE, 2019.

[6] Deng-Ping Fan, Ming-Ming Cheng, Jiangjiang Liu, Shanghua Gao, Qibin Hou, and Ali Borji. Salient objects in clutter: Bringing salient object detection to the foreground. In *ECCV*, volume 11219, pages 196–212, 2018.

[7] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *ICCV*, pages 4558–4567(2017), 2017.

[8] Deng-Ping Fan, Ge-Peng Ji, Ming-Ming Cheng, and Ling Shao. Concealed object detection. *CoRR*, abs/2102.10274, 2021.

[9] Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Camouflaged object detection. In *CVPR*, pages 2774–2784, 2020.

[10] Deng-Ping Fan, Ge-Peng Ji, Tao Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao. Pranet: Parallel reverse attention network for polyp segmentation. In *MICCAI*, volume 12266, pages 263–273, 2020.

[11] Deng-Ping Fan, Tao Zhou, Ge-Peng Ji, Yi Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao. Inf-net: Automatic COVID-19 lung infection segmentation from CT images. *TMI*, 39(8):2626–2637, 2020.

[12] Deng-Ping Fan, Ge-Peng Ji, Xuebin Qin, and Ming-Ming Cheng. Cognitive vision inspired object segmentation metric and loss function. *SCIENTIA SINICA Informationis*, 2021.

[13] Shanghua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip H. S. Torr. Res2net: A new multi-scale backbone architecture. *PAMI*, 43:652–662, 2021.

[14] Lionel Gueguen, Alex Sergeev, Ben Kadlec, Rosanne Liu, and Jason Yosinski. Faster neural networks straight from JPEG. In *NIPS*, pages 3937–3948, 2018.

[15] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *ICCV*, pages 2980–2988, 2017.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[17] Jianqin Yin Yanbin Han Wendi Hou and Jinping Li. Detection of the mobile object with camouflage color under dynamic background based on optical flow. *Procedia Engineering*, 15:2201–2205, 2011.

[18] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring R-CNN. In *CVPR*, pages 6409–6418, 2019.

[19] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *TPAMI*, 20:1254–1259, 1998.

[20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[21] Langley and M. Cynthia. Search images: selective attention to specific visual features of prey. *J Exp Psychol Anim Behav Process*, 22(2):152–163, 1996.

[22] Trung-Nghia Le, Tam V. Nguyen, Zhongliang Nie, Minh-Triet Tran, and Akihiro Sugimoto. Anabranch network for camouflaged object segmentation. *CVIU*, 184:45–56, 2019.

[23] Aixuan Li, Jing Zhang, Yunqiu Lv, Bowen Liu, Tong Zhang, and Yuchao Dai. Uncertainty-aware joint salient object and camouflaged object detection. *CoRR*, abs/2104.02628, 2021.

[24] Bo Li, Zhengxing Sun, and Yuqi Guo. Supervae: Superpixelwise variational autoencoder for salient object detection. In *AAAI*, 2019.

[25] Bo Li, Zhengxing Sun, Lv Tang, and Anqi Hu. Two-b-real net: Two-branch network for real-time salient object detection. In *ICASSP*, 2019.

[26] Bo Li, Zhengxing Sun, Lv Tang, Yunhan Sun, and Jinlong Shi. Detecting robust co-saliency with recurrent co-attention neural network. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 818–825, 2019.

[27] Bo Li, Zhengxing Sun, Junfeng Xu, Shuang Wang, and Peiwen Yu. Saliency based multiple object cosegmentation by ensemble MIML learning. *MTAP*, 2020.

[28] Xin Li, Fan Yang, Leiting Chen, and Hongbin Cai. Saliency transfer: An example-based method for salient object detection. In Subbarao Kambhampati, editor, *IJCAI*, pages 3411–3417, 2016.

[29] Xin Li, Fan Yang, Hong Cheng, Junyu Chen, Yuxiao Guo, and Leiting Chen. Multi-scale cascade network for salient object detection. In *ACM MM*, pages 439–447, 2017.

[30] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 936–944, 2017.

[31] Jiangjiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. A simple pooling-based design for real-time salient object detection. In *CVPR*, pages 3917–3926(2019), 2019.

[32] Nian Liu, Junwei Han, and Ming-Hsuan Yang. Picanet: Learning pixel-wise contextual attention for saliency detection. In *CVPR*, pages 3089–3098(2018), 2018.

[33] Yunqiu Lv, Jing Zhang, Yuchao Dai, Aixuan Li, Bowen Liu, Nick Barnes, and Dengping Fan. Simultaneously localize, segment and rank the camouflaged objects. In *CVPR*, pages 11591–11601, 2021.

[34] Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps. In *CVPR*, pages 248–255(2014), 2014.

[35] Haiyang Mei, Ge-Peng Ji, Ziqi Wei, Xin Yang, Xiaopeng Wei, and Deng-Ping Fan. Camouflaged object segmentation with distraction mining. *CoRR*, abs/2104.10475, 2021.

[36] S. Merilaita, N. E. Scott-Samuel, and I. C. Cuthill. How camouflage works. *Philosophical Transactions of the Royal Society of London*, 372(1724):20160341, 2017.

[37] Yuxin Pan, Yiwang Chen, Qiang Fu, Ping Zhang, and Xin Xu. Study on the camouflaged target detection method based on 3d convexity. *MAS*, 5:152–157, 2011.

[38] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jägersand. Basnet: Boundary-aware salient object detection. In *CVPR*, pages 7479–7489, 2019.

[39] Zequn Qin, Pengyi Zhang, Fei Wu, and Xi Li. Fcanet: Frequency channel attention networks. *CoRR*, abs/2012.11879, 2020.

[40] Jingjing Ren, Xiaowei Hu, Lei Zhu, Xuemiao Xu, Yangyang Xu, Weiming Wang, Zijun Deng, and Pheng-Ann Heng. Deep texture-aware features for camouflaged object detection. *CoRR*, abs/2102.02996, 2021.

[41] P. Sengottuvelan, Amitabh Wahi, and A. Shanmugam. Performance of decamouflaging through exploratory image analysis. In *ICETET*, pages 6–10, 2008.

[42] P Skurowski, H Abdulameer, J Blaszczyk, T Depta, A Kornacki, and P Koziel. Animal camouflage analysis: Chameleon database. In *Unpublished Manuscript*, 2018.

[43] M. Stevens and S. Merilaita. Animal camouflage: current issues and new perspectives. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 2009.

[44] Jinming Su, Jia Li, Yu Zhang, Changqun Xia, and Yonghong Tian. Selectivity or invariance: Boundary-aware salient object detection. In *ICCV*, pages 3798–3807, 2019.

[45] Jinshan Tang, Eli Peli, and Scott T. Acton. Image enhancement using a contrast measure in the compressed domain. *SPL*, 10(10):289–292, 2003.

[46] Lv Tang and Bo Li. CLASS: cross-level attention and supervision for salient objects detection. In *ACCV*, 2020.

[47] Lv Tang, Bo Li, Senyun Kuang, Mofei Song, and Shouhong Ding. Re-thinking the relations in co-saliency detection. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2022.

[48] Lv Tang, Bo Li, Yanliang Wu, Bo Xiao, and Shouhong Ding. Fast: Feature aggregation for detecting salient object in real-time. In *ICASSP*, pages 1525–1529. IEEE, 2021.

[49] Lv Tang, Bo Li, Yijie Zhong, Shouhong Ding, and Mofei Song. Disentangled high quality salient object detection. In *ICCV*, pages 3560–3570. IEEE, 2021.

[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.

[51] Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P. Xing. High-frequency component helps explain the generalization of convolutional neural networks. In *CVPR*, pages 8681–8691, 2020.

[52] Xiaolong Wang, Ross B. Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, pages 7794–7803(2018), 2018.

[53] Jun Wei, Shuhui Wang, and Qingming Huang. F3net: Fusion, feedback and focus for salient object detection. *CoRR*, abs/1911.11445, 2019.

[54] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. CBAM: convolutional block attention module. In *ECCV*, volume 11211, pages 3–19, 2018.

[55] Zhe Wu, Li Su, and Qingming Huang. Cascaded partial decoder for fast and accurate salient object detection. In *CVPR*, pages 3907–3916, 2019.

[56] Kai Xu, Minghai Qin, Fei Sun, Yuhao Wang, Yen-Kuang Chen, and Fengbo Ren. Learning in the frequency domain. In *CVPR*, pages 1737–1746, 2020.

[57] Jinnan Yan, Trung-Nghia Le, Khanh-Duy Nguyen, Minh-Triet Tran, Thanh-Toan Do, and Tam V. Nguyen. Mirrornet: Bio-inspired camouflaged object segmentation. *Access*, 9:43290–43300, 2021.

[58] Fan Yang, Qiang Zhai, Xin Li, Rui Huang, Hong Cheng, and Deng-Ping Fan. Uncertainty-guided transformer reasoning for camouflaged object detection. In *IEEE International Conference on Computer Vision(ICCV)*, 2021.

[59] Qiang Zhai, Xin Li, Fan Yang, Chenglizhao Chen, Hong Cheng, and Dengping Fan. Mutual graph learning for camouflaged object detection. In *CVPR*, pages 12997–13007, 2021.

[60] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Xiang Ruan. Amulet: Aggregating multi-level convolutional features for salient object detection. In *ICCV*, pages 202–211, 2017.

[61] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, pages 6230–6239, 2017.

[62] Jiaxing Zhao, Jiangjiang Liu, Deng-Ping Fan, Yang Cao, Jufeng Yang, and Ming-Ming Cheng. Egnet: Edge guidance network for salient object detection. In *ICCV*, pages 8778–8787, 2019.

[63] Ting Zhao and Xiangqian Wu. Pyramid feature attention network for saliency detection. In *CVPR*, pages 3085–3094, 2019.

[64] Yijie Zhong, Bo Li, Lv Tang, Hao Tang, and Shouhong Ding. Highly efficient natural image matting. In *BMVC*, 2021.

[65] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *MICCAI*, volume 11045, pages 3–11, 2018.