CamoFocus: Enhancing Camouflage Object Detection with Split-Feature Focal Modulation and Context Refinement

Abbas Khan[1], Mustaqeem Khan[1], Wail Gueaieb[1,2], Abdulmotaleb El Saddik[1,2], Giulia De Masi[3], Fakhri Karray[1,4]

[1]MBZUAI, UAE, Email: [Abbas.Khan, Mustaqeem.Khan]@mbzuai.ac.ae
[2]University of Ottawa, Canada, Email: [wgueaieb, elsaddik]@uottawa.ca
[3]Technology Innovation Institute, UAE, Email: giulia.demasi@tii.ae
[4]University of Waterloo, Canada, Email: karray@uwaterloo.ca

Abstract

Camouflage Object Detection (COD) involves the challenge of isolating a target object from a visually similar background, presenting a formidable challenge for learning algorithms. Drawing inspiration from state-of-the-art (SOTA) Focal Modulation Networks, our objective is to proficiently modulate the foreground and background components, thereby capturing the distinct features of each. We introduce a Feature Split and Modulation (FSM) module to attain this goal. This module efficiently separates the object from the background by utilizing foreground and background modulators guided by a supervisory mask. For enhanced feature refinement, we propose a Context Refinement Module (CRM), which considers features acquired from FSM across various spatial scales, leading to comprehensive enrichment and highly accurate prediction maps. Through extensive experimentation, we showcase the superiority of CamoFocus over recent SOTA COD methods. Our evaluations encompass diverse benchmark datasets, including CAMO, COD10K, CHAMELEON, and NC4K. The findings underscore the potential and significance of the proposed CamoFocus model and establish its efficacy in addressing the critical challenges of camouflage object detection.

1. Introduction

The objects characterizing substantial visual similarity to the background, diminutive sizes, and obscure textures are generally known as camouflaged objects. These objects usually pose significant challenges to the Computer Vision (CV) algorithm or sometimes even humans, thus making Camouflaged Object Detection (COD) extremely challenging. COD has become an important area of research in CV and Machine Learning due to its numerous applications in aquaculture [6], wildlife conservation [23], search, and rescue operations. Besides, it also plays an important role in the medical field for polyp segmentation and other similar tasks [20].

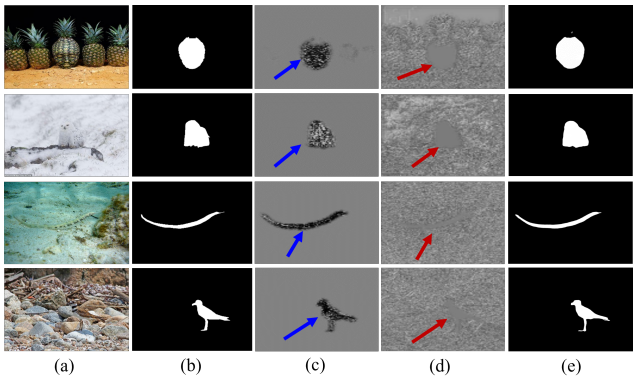

Figure 1. Our proposed FSM module modulates background and foreground features. In (c), the foreground modulator highlights object-related activations, while (d) emphasizes background features, suppressing object-related ones (indicated by the red arrow). (e) displays the refined result from our CamoFocus model much closer to ground truth (b), illustrating the effectiveness of the CRM.

The lack of discernible differences in color, texture, and lower object-background contrast makes COD highly challenging. In addition, smaller objects with a higher resemblance to their surroundings even more complicate the detection process. Therefore, numerous task-specific methods [25,30], auxiliary information-based techniques [35,40] and bio-inspired approaches [18, 34] are used to solve this task. Recent advancements have seen non-bio-inspired techniques exhibit remarkable performance, particularly Vision-Transformed-Based methods like CamoFormer [33] and DTINet [16]. However, their reliance on resource-intensive attention mechanisms has increased computa-

tional complexity. Diverging from this, our proposed CamoFocus takes a fresh direction inspired by cutting-edge Focal Modulation Networks [32]. Compared to 18 other, state-of-the-art (SOTA) methods, CamoFocus achieves superior performance while utilizing comparatively fewer parameters and requiring lower computational resources. To enhance Camouflage Object Detection (COD), our proposed CamoFocus technique introduces two key components. Firstly, we present the Feature Split and Modulation (FSM) module, which splits backbone features, enabling a better understanding of object-background relationships. This is achieved through the foreground and background modulators, which respond differently to input features shaped by the interplay of the mask and backbone features. This approach helps recognize background and foreground features more effectively, as depicted in Fig. 1. Unlike previous methods, which used attention-based modules for comprehending object-background relationships in camouflaged scenarios, our FSM module takes a more focused approach. It treats background and foreground components independently, overcoming limitations of existing techniques and improving performance in challenging situations. After learning the features, they are merged to create more understandable feature maps for subsequent processing by the Context Refinement Module. **2)** Context Refinement Module (CRM): While FSM gains significant object-background comprehension, we employ CRM for further enhancement. This module facilitates cross-scale semantic understanding of features. CRM accepts two different scale inputs, employing bilinear upsampling for channel-wise concatenation. Following this, the concatenated feature map traverses a sequence of convolutions, succeeded by a global skip connection. To maintain lower computational overhead and merely understand the cross-scale modulated feature maps, CRM is utilized in a streamlined manner, relying solely on various convolutional layers with diverse receptive fields and filter sizes. Despite its simplistic architecture, empirical verification underscores CRM's efficacy within the proposed network.

To summarize, the contribution of the proposed CamoFocus to COD are:

- We introduce a novel Feature Split and Modulation (FSM) module that aims to excavate a target object from the surrounding environment utilizing Foreground and Background Modulators. Guided by a supervisory mask, our FSM effectively segregates foreground and background elements, enabling precise object discernment within complex visual contexts.

- To further refine the modulated multi-scale feature representation, we employ a simple yet practical Context Refinement Module (CRM) that enhances the feature representation by cross-scale interaction of different feature maps acquired from the preceding FSM.

- We achieve SOTA results on the testing sets of four benchmark datasets of COD across all evaluation metrics. Extensive experimentation and ablation studies indicate the effectiveness of the proposed technique.

2. Previous Work

Camouflage Objects are intentionally or unintentionally concealed in the surrounding environment and circumvent easy detection. Unlike generic and salient objects, which humans and CV algorithms easily notice, camouflage objects require significant human perception and sophisticated algorithms for their identification [5]. In traditional techniques, hand-crafted features are mainly used to excavate the camouflaged object from the background [8]. These techniques perform satisfactorily in rudimentary tasks but exhibit substantial performance degradation when deployed in intricate scenarios. To circumvent this challenge, data-driven approaches emerged to be more effective in the COD. Subsequently, the advent of large-scale COD-related datasets [5, 12, 17] and tremendous advancements in the CV techniques have enabled substantial progress in the COD.

2.1. Camouflage Object Detection

Over the recent years, various sophisticated techniques have been proposed to tackle the highly challenging COD. Le et al. [12] proposed an auxiliary classification-based technique to improve the performance of COD. Another important work proposed by Fan et al. [6] firstly locates the concealed objects and then performs segmentation. Other than these, some studies have attempted to mimic bio-inspired mechanisms for COD. A study by Mei et al. [18] uses attention mechanisms to initially locate the object and then effectively suppress distractors. A comparatively recent technique by Fan et al. [5] refined coarse maps, and Zhang et al. [37] utilized sensory and cognitive modules. These methods excel at detecting larger objects. To address varied object sizes, Pang et al. [21] use zooming in and out to mimic human vision, and Jia et al. [10] adopt progressive refinement.

2.2. Vision-Transformer-Based Techniques

Various attention mechanisms and transformer-based networks are highly utilized to improve the performance of COD. Sun et al. [25] proposed attention-induced cross-level fusion and dual-branch global context to improve the feature representation. Similarly, Yang et al. [31] leveraged vision transformers coupled with uncertainty quantification and presented a joint framework by combining probabilistic and deterministic techniques. The work proposed by Liu et al. [16] utilized twin transformers for separate background and foreground identification coupled with the

negative mining strategy to improve the COD performance. Another technique proposed by Zhang et al. [37] utilizes a progressive refinement technique and enables information exchange among different image regions in complex objects. Besides these, Zhai et al. [35] proposed exploiting the Graph Neural Network to induce object boundary details into the learning process. The method initially locates the object and refines it using the object boundary-related cues.

3. Proposed Method

3.1. Motivation

Our work aims to improve COD using efficient techniques, drawing inspiration from Focal Modulation as a promising alternative to the conventional attention mechanisms used in this domain. In our framework, we use a more targeted approach for effective foreground-background modulation with FSM module. To further enhance the cross-scale contextual understanding between the features, we use another module to effectively combine features and aggregate the context of the camouflaged object. These modules enable our proposed CamoFocus to achieve higher performance with relatively fewer parameters, promising efficiency and effectiveness in COD.

3.2. Overall Architecture

Our proposed CamoFocus, depicted in Fig. 2, comprises three core components: Backbone, FSM, and CRM. Given the input image $I_o \in \mathbb{R}^{H\times W\times 3}$, the backbone extracts distinct low and high-level features from the given image input image at five stages. The first stage features x_o having a spatial size of $\frac{H}{4} \times \frac{W}{4}$ contain rudimentary information and hence are not further utilized, whereas features from remaining stages, x_1, x_2, x_3, x_4, having low to high-level features with spatial dimensions of $\frac{H}{4}\times\frac{W}{4}$, $\frac{H}{8}\times\frac{W}{8}$, $\frac{H}{16}\times\frac{W}{16}$, $\frac{H}{32} \times \frac{W}{32}$ respectively, undergo channel-wise reduction followed by L_2 normalization and ReLU [2] activation. The mask m is extracted via the Mask function from the concatenated features of x_2 and x_3. Afterward, we utilize four FSM modules to process the combination of extracted features and mask features as illustrated in the mf_i of Fig. 2. Each FSM module yields the processed feature maps in the form of x_1', x_2', x_3', and x_4', which are subsequently fed to the CRM for further refinement. Finally, the cumulative loss is computed across all stages, considering the outputs of the CRMs denoted as P_1, P_2, and P_3, in addition to the mask output m and the ground truth S_o.

3.3. Mask Generation

In our method, masks are obtained by integrating two distinct features, x_2 and x_3, chosen for their information-rich content and reasonable spatial resolution. After spatial equalization and concatenation of x_2 and x_3, two successive convolutional blocks, each consisting of 3×3 convolutions, L_2 normalization and ReLU [2] are applied. The initial block's channel count equals the sum of channels from x_2 and x_3, while the subsequent block's channel count aligns with the output of the first block. The resultant mask m is obtained using Sigmoid activation. This mask m subsequently interacts with features from different stages of backbone x_n, via the mfi function in the FSM module.

3.4. Feature Split and Modulation

In order to achieve a better understanding of the features and effectively segregate the object from intricate backgrounds, we build Feature Split and Modulation. Based on Focal Modulation [32], we use two identical modulators to map foregrounds and backgrounds separately. As demonstrated in Fig. 3, features from the backbone x_n and mask m and $1-m$ undergoes element-wise multiplication in mfi, thereby resulting in x_f and x_b for foreground and background, respectively. After element-wise multiplication, the foreground and background features are projected using two separate linear layers with Eq. 1 and Eq. 2.

$$Z_f^0 = f(X_f) \in \mathbb{R}^{H\times W\times C} \tag{1}$$

$$Z_b^0 = f(X_b) \in \mathbb{R}^{H\times W\times C} \tag{2}$$

$f(X_f)$ and $f(X_b)$ are the background and foreground projection layers. We pass the projected features Z_f^0 and Z_b^0 through a series L of depth-wise convolutions to obtain a distinct understanding of the context for both foreground and background objects via Foreground Modulator (FM) and Background Modulator (BM) as demonstrated in Fig 3. Each block consists of a stack of depthwise convolution layers $l \in 1, ..., L$. Contrary to the original architecture, ReLU [2] activates both blocks since it performs better during the empirical process. In (FM), we use two levels of depth having the initial kernel size of 7 and then increasing by a focal factor of 2. Similarly, in the (BM) block, we use identical kernel size as in (FM).

$$Z_f^l = f_{af}^l((Z_f^{l-1}) = \text{ReLU}(\text{DWConv}(Z_f^{l-1}))) \in \mathbb{R}^{H\times W\times C} \tag{3}$$

$$Z_b^l = f_{ab}^l((Z_b^{l-1}) = \text{ReLU}(\text{DWConv}(Z_b^{l-1}))) \in \mathbb{R}^{H\times W\times C} \tag{4}$$

f_{af}^l and f_{ab}^l in Eq. 3 and Eq. 4 are used as a contextualization function to obtain the contextually aware feature maps using the (FM) and (BM), respectively. The kernel size k for both (FM) and (BM) is initialized with 7 in the first layer with an increase of 2 in the subsequent layers, and the final receptive field using the mechanism

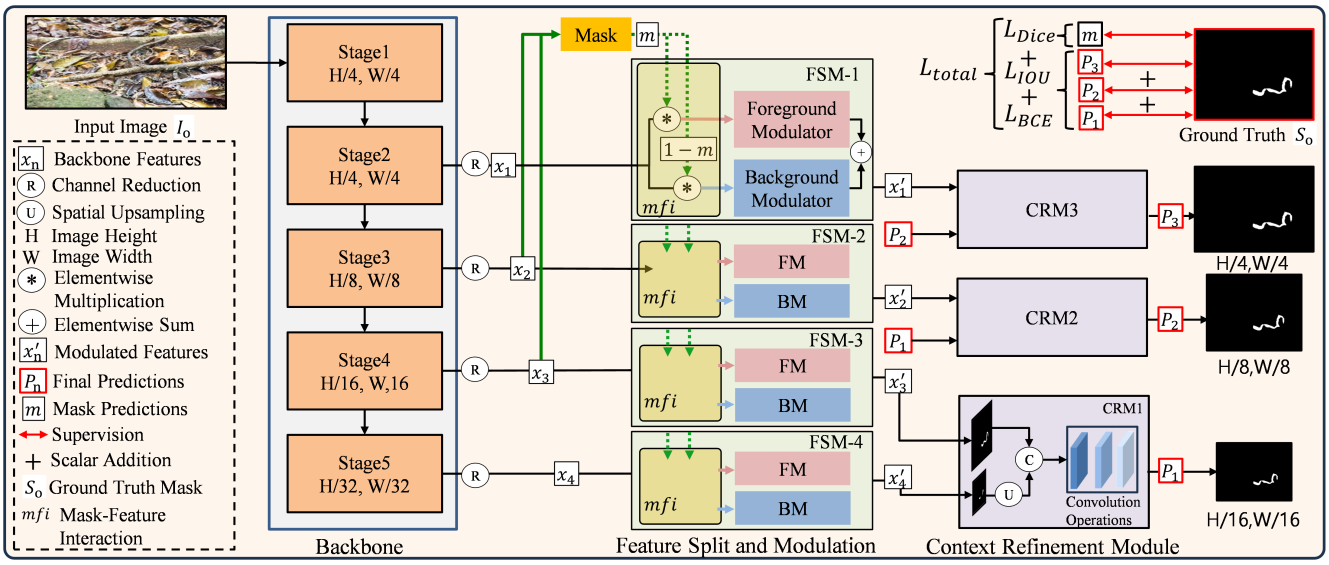

Figure 2. The proposed framework involves a multi-stage process for COD. The input image is first passed through a Backbone network, which extracts x_1, x_2, x_3, and x_4 features. Subsequently, all features x_n are passed through (R) for normalization and dimensionality reduction. The next stage involves the Feature Split and Modulation module, which receives the masks m and features x_n and splits them to obtain a better understanding of the underlying features of the object and background via foreground modulator (FM) and background modulator (BM). The output of FSM module x'_n is subsequently fed to the Context Refinement Module, which outputs P_1, P_2, and P_3, representing predictions at different scales. Finally, the total loss is computed between the ground truth S_o and the mask m and a combination of P_1, P_2, and P_3.

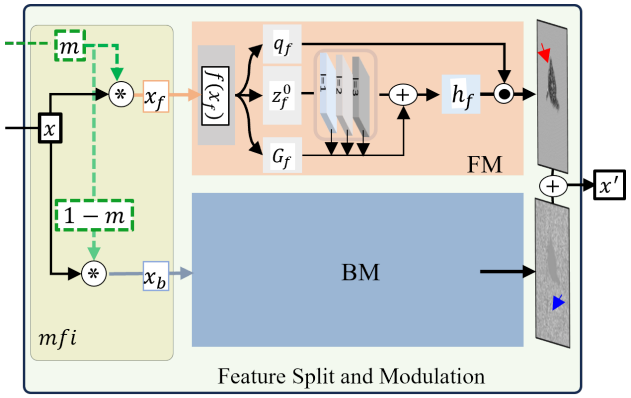

Figure 3. Illustrates the overall mechanism of the Feature Split and Modulation Module. As visualized in the figure, (mfi), (FM), and (BM) are the key components of the FSM module. As indicated by the red and blue arrows, the output of both of these modulators separately attends to the foreground and background of the given feature map.

aligned with the [32] is obtained as $r = 1 + \sum_{i=1}^{k}(k_i - 1)$ and hence the final output obtained by the block is $Z_{fl=1}^{l\,L+1}$ for foreground and $Z_{bl=1}^{l\,L+1}$ for background respectively. Both modulators use gated aggregation G to allow specific and contextually-aware features to the subsequent layers. We have utilized the gating mechanism at each focal level, which helps the network in hierarchical context aggregation. For the (BM), we obtain the gating as $G_b = f_g, b(X_c) \in \mathbb{R}^{H\times W\times(L+1)}$, whereas for the (FM), we obtain gating as $G_f = f_g, f(X_c) \in \mathbb{R}^{H\times W\times(L+1)}$. Afterward, the dot product is performed in both (FM) and (BM) between the feature maps and their respective gates as given by Eq. 5 and Eq. 6.

$$Z_f^{out} = (\sum_{l=1}^{L+1} G^l \odot Z_f^l) \tag{5}$$

$$Z_b^{out} = (\sum_{l=1}^{L+1} G^l \odot Z_b^l) \tag{6}$$

After contextual aggregating the inputs at each focal level l, we obtain the global output map for (FM) and (BM) via Eq. 7 and Eq. 8, respectively.

$$y_{fi} = q_f(x_i) \odot \left(\sum_{\ell=1}^{L} g_{\ell i,f} \cdot z_{\ell i,f}\right) \tag{7}$$

$$y_{bi} = q_b(x_i) \odot \left(\sum_{\ell=1}^{L} g_{\ell i,b} \cdot z_{\ell i,b}\right) \tag{8}$$

Finally, Eq. 9 helps to acquire a combined map of x'_n of (FM) and (BM).

$$x'_n = y_{fi} + y_{bi} \tag{9}$$

After obtaining the modulated feature map x'_n, we pass the feature map to the subsequent block (CRM) for further refinement.

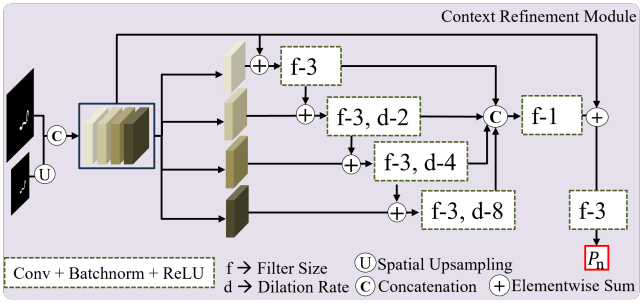

Figure 4. Context Refinement Module: takes two input feature maps and performs a cross-scale operation to further enhance the contextual semantic understanding of the proposed technique.

3.5. Context Refinement Module

For further refining and extracting more cross-scale semantics from the modulated features yielded by FSM, we employ three CRMs in our technique. Each CRM is essentially a combination of several convolution layers, L_2 normalization and ReLU [2] activation. Specifically, each CRM has six convolution layers followed by ReLU and L_2 normalization. As illustrated in Fig 4., each convolution layer has a specific kernel size (f) and dilation rate (d) to specifically attend to the different levels of features. CRM takes two inputs with different spatial sizes and processes them cross-scale. For instance, the first CRM operates on x'_n and $x_n + 1'$ by applying bilinear operation on $x_n + 1'$ to match the spatial dimension of both inputs. Afterward, we apply a concatenation operation on the channel dimension of these inputs. As demonstrated in Fig. 4, the concatenated feature map is passed through a 1x1 convolution. These features are then channel-wise split into four chunks, as demonstrated in Fig 4. Skip connection is hugely employed to obtain a solid semantic relationship between different spatial-level features. Each CRM block results in a prediction map (P_n), which is supervised vis-a-vis ground truth (S_o) at each spatial level as visualized in the Fig. 2.

3.6. Loss Function

We employ a combination of loss functions to supervise our model: Weighted Intersection Over Union L_w^{IOU} and weighted Binary Cross-entropy L_w^{BCE}. IOU in image segmentation is generally responsible for maintaining the structure of the predicted map following the ground truth. Adding a weighting factor to the IOU loss enhances its performance by assigning different weights to the class regions based on their importance in the task. On the other hand, Binary Cross-entropy is the pixel-wise classification, and L_w^{BCE} similarly performs better on complex samples. Combining these losses enables the network to focus more on complex samples, which are expected to rise in the COD. The L_w^{BCE} and L_w^{IOU} losses are computed on the predictions of three camouflaged object masks $(P_i, i \in 1, 2, 3)$ obtained from the CRM module. Similarly, to supervise the mask (m), we use Dice Loss L^{Dice} to improve the mask quality. Since we are using three CRMs therefore, the total loss is computed as the sum of the L_w^{BCE} and L_w^{IOU} losses across the three camouflaged object masks: By incorporating these supervisory signals at different stages into the training process, our proposed model can effectively learn to segment the hidden object in complex scenes. Eq. 10 provides the total loss L_{total} for the network's supervision.

$$L_{total} = \sum_{i=1}^{3} (L_w^{BCE}(P_i, S_o) + L_w^{IOU}(P_i, S_o)) + L^{Dice}(m, S_o)) \tag{10}$$

4. Experimental Results

4.1. Training Settings and Reproducibility

For CamoFocus' implementation, we use Pytorch DL library [22] and employ PVTv2 [28] pre-trained on the ImageNet database [11] . Additionally, we also investigate our technique with other backbones such as Res2Net [7], and EfficientNet-B1 [26] to ensure fair comparison with SOTA techniques. Other than the backbone, the rest of the model is randomly initialized. We resize all images to 416x416 and use Adam optimizer initialized with a learning rate of 1×10^{-4}. We train each model in an end-to-end manner for 90 epochs. Similarly, we maintain a consistent batch size of 24 throughout the experiments. To avoid overfitting and achieve better training performance, we use the learning rate scheduler "poly" [19], gradually decreasing the learning over time and enhancing the models' better convergence. We conducted the experimentation using a dual NVIDIA A100 GPU with a 40G capacity. Depending on the selection of hyperparameters, the complete training time of a single model is recorded to be in the range of 2 to 3 hours.

4.2. Datasets

We conduct experiments on four COD benchmark datasets, CAMO [12], COD10K [5], NC4K [17], and CHAMELEON [24]. For training, we follow the same dataset segregation protocols used in the prior works [18, 21] to ensure an unbiased comparison. Specifically, we train the network on 1000 images from CAMO and 3040 images from COD10K. Similarly, we evaluate the proposed method on the testing sets of all four COD benchmark datasets con-

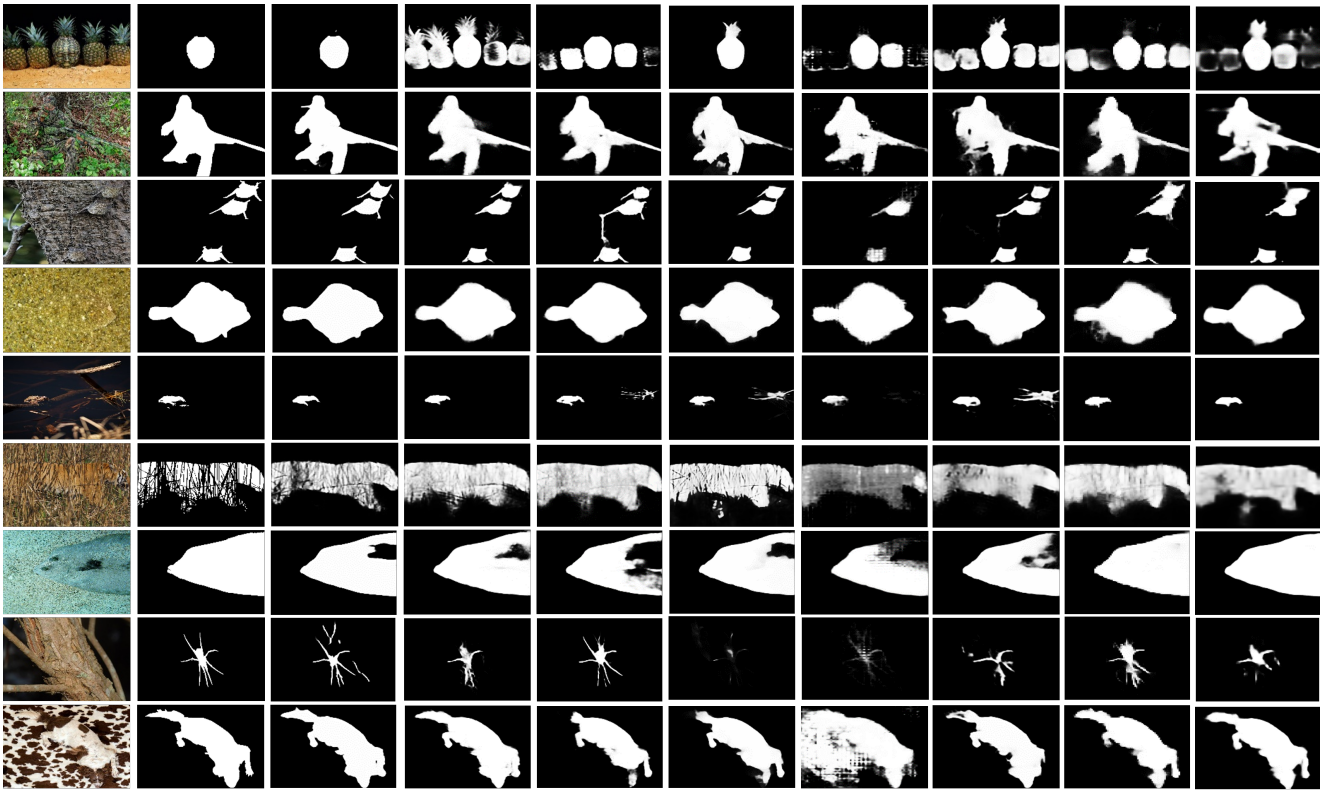

Figure 5. The proposed CamoFocus achieves visually better performance than all most of the recent and SOTA methods, including the recent (d) CamoFormer [33], (e) SegMar [10], (f) ZoomNet [21], and others. The result of CamoFocus on the most challenging image in the first row highlights the performance of the technique.

taining various challenging images as demonstrated in the figures.

4.3. Evaluation Metrics

The COD's widely recognized and most commonly used evaluation metrics are Structure-Measure [3] (S_m), Mean Absolute Error (M), weighted F-measure [1] F_β^ω, and Adaptive E-measure [4] αE. To explain these metrics briefly, (S_m) is used to measure the similarity between the structural content of two images, and (M) is used to determine the absolute pixel difference between the ground truth and the predicted image. Similarly, F_β^ω is another important metric that measures the harmonic mean between the precision and recall of the binary classification. The weight factor in this metric is used to address the class imbalance problem in the dataset. Finally, αE considers the structure and texture of two images to calculate their difference.

4.4. Comparison with SOTA Techniques

In order to showcase the superiority of the proposed method, we compare it with 18 SOTA COD methods, i.e., SINet [6], MGL-R [35], C2FNet [25], PFNet [18], PreyNet [36], UGTR [30], BSANet [40], UJSC [13], VST [15], COS-T [27], DGNet [9], SegMar [10], ZoomNet [21], MFFN [38], DTINet [16], DGNet [9], PopNet [29] and CamoFormer [33]. Compared with these methods, we achieve superior performance in terms of qualitative and quantitative analysis.

4.5. Qualitative Results

The qualitative analysis of the proposed technique compared to SOTA methods is demonstrated in Fig. 5. The images demonstrated in Fig. 5, spanning various scales, are selected from all four testing sets. The findings reveal that the proposed method surpasses existing ones in terms of producing superior object structure and more fined details across all scales of objects. The qualitative results demonstrate the effectiveness of the proposed technique over the SOTA alternatives. Similarly, as demonstrated in Fig. 6, the proposed technique works much better for the most challenging examples as well.

Table 1. The table illustrates the comparative analysis of the results obtained on test sets of four widely used benchmark datasets in COD. S_m, αE, F_β^ω,F_β and M denote Structure Measure, Adaptive E-measure, Weighted F-measure, F-measure, and Mean Absolute Error respectively. The higher values of the proposed technique (highlighted in bold) on all datasets across all evaluation metrics indicate the effectiveness of the proposed method in the COD. Similarly -R indicates ResNet50, -R2 indicates Res2Net50, -E4 EfficientNetB4, -E1 EfficientNetB1, -P4 indicates PVTv4 and -P2 indicated PVTv2

Method	NC4K (4121 images)					COD10K-Test (2026 images)					CHAMELEON (76 images)					CAMO-Test (250 images)				
	$S_m\uparrow$	$\alpha E\uparrow$	$F_\beta^\omega\uparrow$	$F_\beta\uparrow$	$M\downarrow$	$S_m\uparrow$	$\alpha E\uparrow$	$F_\beta^\omega\uparrow$	$F_\beta\uparrow$	$M\downarrow$	$S_m\uparrow$	$\alpha E\uparrow$	$F_\beta^\omega\uparrow$	$F_\beta\uparrow$	$M\downarrow$	$S_m\uparrow$	$\alpha E\uparrow$	$F_\beta^\omega\uparrow$	$F_\beta\uparrow$	$M\downarrow$
Convolution Based Methods																				
SINet-R [6]	0.808	0.883	0.723	0.769	0.058	0.776	0.867	0.631	–	0.043	0.872	0.938	0.806	–	0.034	0.745	0.825	0.644	–	0.092
MGL-R [35]	0.833	0.867	0.740	0.782	0.052	0.814	0.865	0.666	–	0.035	0.893	0.923	0.812	–	0.031	0.775	0.848	0.673	–	0.088
C2FNet-R2 [25]	0.838	0.901	0.762	0.795	0.049	0.813	0.886	0.686	–	0.036	0.888	0.932	0.828	–	0.032	0.796	0.864	0.719	–	0.080
UGTR-R [30]	0.839	0.889	0.747	0.787	0.052	0.818	0.850	0.667	–	0.035	0.888	0.921	0.794	–	0.031	0.784	0.859	0.794	–	0.086
PFNet-R [18]	0.829	0.894	0.745	0.784	0.053	0.800	0.868	0.660	–	0.040	0.882	0.942	0.810	–	0.033	0.782	0.852	0.695	–	0.085
PreyNet-R [36]	–	–	–	–	–	0.813	0.894	0.697	–	0.034	0.902	0.951	0.856	0.866	0.027	0.790	0.854	0.708	0.763	0.077
BSANet-R2 [40]	–	–	–	–	–	0.818	0.894	0.699	–	0.034	0.895	0.946	0.841	–	0.027	0.769	0.851	0.717	–	0.079
ZoomNet-R [21]	0.853	0.907	0.784	0.818	0.043	0.838	0.893	0.729	–	0.029	0.902	0.952	0.845	–	0.023	0.820	0.883	0.752	–	0.066
FDNet-R2 [39]	0.834	0.895	0.750	–	0.052	0.837	0.897	0.731	–	0.030	0.894	0.948	0.819	–	0.030	0.844	0.903	0.778	–	0.062
OCENet-R [14]	0.857	0.899	–	0.817	0.044	0.832	0.890	–	0.745	0.032	0.901	0.940	–	0.843	0.028	0.802	0.866	–	0.767	0.075
SegMar-R [10]	0.841	0.905	0.781	–	0.046	0.833	0.895	0.724	–	0.033	0.897	0.950	0.835	–	0.027	0.815	0.872	0.742	–	0.071
MFFN-R2 [38]	0.856	0.915	0.791	0.827	0.042	0.846	0.917	0.745	–	0.028	0.905	0.963	0.852	–	0.021	–	–	–	–	–
PopNet [29]	0.852	0.908	0.852	–	0.043	0.851	0.910	0.757	–	0.028	0.910	0.962	0.893	–	0.022	0.808	0.871	0.744	–	0.077
CamoFormer-R [33]	0.857	0.915	0.793	–	0.024	0.838	0.898	0.730	–	0.029	0.900	0.949	0.843	–	0.024	0.817	0.884	0.756	–	0.066
DGNet-E4 [9]	0.857	0.910	0.784	–	0.042	0.822	0.879	0.693	–	0.033	0.890	0.934	0.816	–	0.029	0.839	0.901	0.769	–	0.057
CamoFocus-R	**0.847**	**0.910**	**0.788**	**0.812**	**0.043**	**0.825**	**0.903**	**0.719**	**0.749**	**0.033**	**0.898**	**0.953**	**0.849**	**0.859**	**0.027**	**0.812**	**0.873**	**0.752**	**0.794**	**0.071**
CamoFocus-E1	**0.855**	**0.912**	**0.790**	**0.820**	**0.042**	**0.830**	**0.899**	**0.719**	**0.735**	**0.030**	**0.901**	**0.940**	**0.846**	**0.837**	**0.024**	**0.830**	**0.893**	**0.770**	**0.806**	**0.062**
Transformer Based Methods																				
VST-T [15]	0.830	0.887	0.740	–	0.053	0.810	0.866	0.680	–	0.035	0.888	0.936	0.820	–	0.033	0.805	0.863	0.780	–	0.069
COS-T [27]	0.825	0.881	0.730	–	0.055	0.790	0.901	0.693	–	0.035	0.885	0.948	0.854	–	0.025	0.813	0.896	0.776	–	0.060
DTINet-T [16]	0.863	0.915	0.792	–	0.041	0.824	0.893	0.695	–	0.034	0.883	0.928	0.813	–	0.033	0.857	0.912	0.796	–	0.050
CamoFormer-P4 [33]	0.892	0.941	0.847	–	0.030	0.869	0.931	0.786	–	0.023	0.910	0.970	0.865	–	0.022	0.872	0.931	0.831	–	0.046
CamoFocus-P2	**0.889**	**0.936**	**0.853**	**0.870**	**0.030**	**0.873**	**0.935**	**0.802**	**0.818**	**0.021**	**0.912**	**0.957**	**0.876**	**0.884**	**0.023**	**0.873**	**0.926**	**0.842**	**0.861**	**0.043**

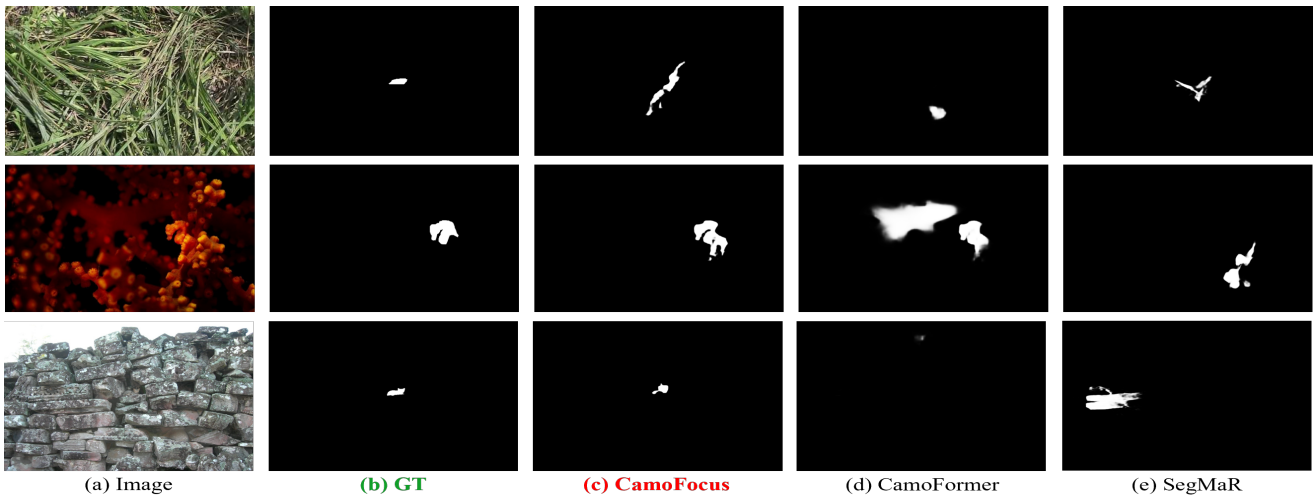

Figure 6. Visualisation of the performance comparison of the CamoFocus on most challenging small objects with other baseline techniques, i.e. (b)ZoomNet [21] and (c)SegMaR [10] on instances of tiny objects. The first two rows are from the NC4K dataset, and the last row represents an instance of the COD10K dataset.

Table 2. Ablation Study and the impact of FSM and CRM modules on the performance across four benchmark datasets.

Method	COD10K-Test (2026 images)				CAMO-Test (250 images)				CHAMELEON (76 images)				NC4K (4121 images)			
	S_m	αE	F_β^ω	M	S_m	αE	F_β^ω	M	S_m	αE	F_β^ω	M	S_m	αE	F_β^ω	M
Baseline (B)	0.831	0.882	0.750	0.057	0.829	0.811	0.693	0.081	0.081	0.893	0.795	0.055	0.742	0.822	0.773	0.061
B + FSM	0.868	0.900	0.790	0.023	0.870	0.920	0.829	0.045	0.905	0.948	0.869	0.024	0.882	0.931	0.849	0.312
B+FSM+CRM	**0.873**	**0.935**	**0.802**	**0.021**	**0.873**	**0.926**	**0.842**	**0.043**	**0.912**	**0.957**	**0.876**	**0.023**	**0.889**	**0.936**	**0.853**	**0.030**

4.6. Quantitative Results

In terms of quantitative analysis, Table 1 suggests that the proposed technique outperforms 18 other SOTA techniques across all evaluation metrics. To ensure a fair comparison, we use a standard evaluation code snippet on the prediction results that are either directly provided by the authors of the other techniques or reproduced by their provided trained models. Moreover, it is evident in the Fig. 7 that the proposed technique consistently utilizes fewer parameters as compared to the others with any backbone. It could be noted that the proposed technique outperforms even the recently proposed sophisticated techniques, including ZoomNet and SegMaR, thereby establishing a new SOTA in the COD.

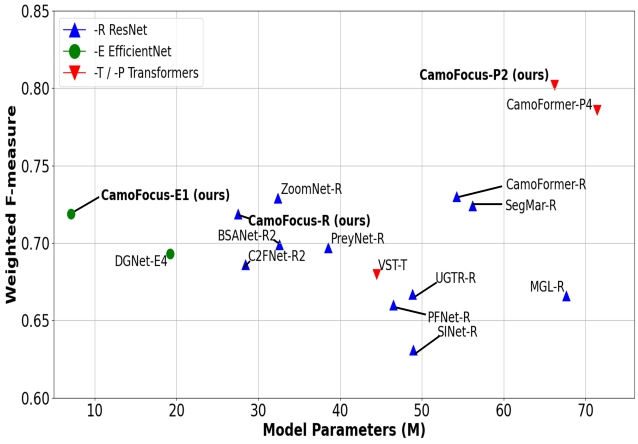

Figure 7. Our proposed technique, with distinct backbones, consistently outperforms existing methods while maintaining a leaner parameter footprint in the majority of scenarios. This compelling balance between efficiency and efficacy underscores its potential as a top-performing solution in various COD applications.

4.7. Ablation Study

In order to demonstrate the impact of each module in the proposed work, we conduct an ablation study by selectively adding and removing modules to the baseline. As a baseline, we consider the final layers of the backbone followed by the 1x1 convolution and passed through a single CRM block. Then, we systematically add the FSM module to demonstrate its contribution to the proposed method. Finally, we add multiple CRM modules to refine the results obtained by the preceding FSM block. Table 2 highlights each module's performance gains in our proposed approach.

4.8. FSM Effectiveness and Discussion

The efficacy of the proposed FSM module is investigated in this section. The results presented in Table 2 demonstrate that the inclusion of FSM significantly enhances the performance of the baseline model. For instance, a huge performance gain in the weighted F-measure F_β^ω can be noticed by adding the FSM module to the baseline, thereby underscoring the effectiveness of the module.

4.9. CRM Effectiveness and Discussion

Since the CRM used in the proposed technique is straightforward consisting of convolution layers with varying dilation and filter sizes, it is illustrated in Table 2 that the inclusion of CRM improves the overall results of the proposed technique across all evaluation metrics. Despite its simple design, the CRM reasonably elevates the performance of our proposed technique across all datasets.

5. Conclusion

In conclusion, our work introduces CamoFocus, a novel approach inspired by Focal Modulation Networks to enhance Camouflaged Object Detection (COD). By efficiently splitting and modulating features, our two core modules, Feature Split and Modulation, in collaboration with Context Refinement, refine camouflaged object-related features. CamoFocus achieves a new COD benchmark, outperforming 18 recent SOTA methods while requiring fewer parameters. It embodies our motivation to improve COD through efficient techniques and promises both effectiveness and efficiency in this domain.

6. Acknowledgement

This research is a part of the joint project "Intelligent Object Detection, Dynamic Scene Analysis, and Activity Recognition for Real-Time UAV Applications." between two esteemed institutions, the Technology Innovation Institute (TII) and Mohamed bin Zayed University of Artificial Intelligence (MBZUAI).

References

[1] Radhakrishna Achanta, Sheila Hemami, Francisco Estrada, and Sabine Susstrunk. Frequency-tuned salient region detection. In *2009 IEEE conference on computer vision and pattern recognition*, pages 1597–1604. IEEE, 2009. 6

[2] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018. 3, 5

[3] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *Proceedings of the IEEE international conference on computer vision*, pages 4548–4557, 2017. 6

[4] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment measure for binary foreground map evaluation. *arXiv preprint arXiv:1805.10421*, 2018. 6

[5] Deng-Ping Fan, Ge-Peng Ji, Ming-Ming Cheng, and Ling Shao. Concealed object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6024–6042, 2021. 2, 5

[6] Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Camouflaged object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2777–2787, 2020. 1, 2, 6, 7

[7] Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr. Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*, 43(2):652–662, 2019. 5

[8] Joanna R Hall, Innes C Cuthill, Roland Baddeley, Adam J Shohet, and Nicholas E Scott-Samuel. Camouflage, detection and identification of moving targets. *Proceedings of the Royal Society B: Biological Sciences*, 280(1758):20130064, 2013. 2

[9] Ge-Peng Ji, Deng-Ping Fan, Yu-Cheng Chou, Dengxin Dai, Alexander Liniger, and Luc Van Gool. Deep gradient learning for efficient camouflaged object detection. *Machine Intelligence Research*, 20(1):92–108, 2023. 6, 7

[10] Qi Jia, Shuilian Yao, Yu Liu, Xin Fan, Risheng Liu, and Zhongxuan Luo. Segment, magnify and reiterate: Detecting camouflaged objects the hard way. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4713–4722, 2022. 2, 6, 7

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 5

[12] Trung-Nghia Le, Tam V Nguyen, Zhongliang Nie, Minh-Triet Tran, and Akihiro Sugimoto. Anabranch network for camouflaged object segmentation. *Computer vision and image understanding*, 184:45–56, 2019. 2, 5

[13] Aixuan Li, Jing Zhang, Yunqiu Lv, Bowen Liu, Tong Zhang, and Yuchao Dai. Uncertainty-aware joint salient object and camouflaged object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10071–10081, 2021. 6

[14] Jiawei Liu, Jing Zhang, and Nick Barnes. Modeling aleatoric uncertainty for camouflaged object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1445–1454, 2022. 7

[15] Nian Liu, Ni Zhang, Kaiyuan Wan, Ling Shao, and Junwei Han. Visual saliency transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4722–4732, 2021. 6, 7

[16] Zhengyi Liu, Zhili Zhang, Yacheng Tan, and Wei Wu. Boosting camouflaged object detection with dual-task interactive transformer. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 140–146. IEEE, 2022. 1, 2, 6, 7

[17] Yunqiu Lv, Jing Zhang, Yuchao Dai, Aixuan Li, Bowen Liu, Nick Barnes, and Deng-Ping Fan. Simultaneously localize, segment and rank the camouflaged objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11591–11601, 2021. 2, 5

[18] Haiyang Mei, Ge-Peng Ji, Ziqi Wei, Xin Yang, Xiaopeng Wei, and Deng-Ping Fan. Camouflaged object segmentation with distraction mining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8772–8781, 2021. 1, 2, 5, 6, 7

[19] Purnendu Mishra and Kishor Sarawadekar. Polynomial learning rate policy with warm restart for deep neural network. In *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)*, pages 2087–2092. IEEE, 2019. 5

[20] Phan Ngoc Lan, Nguyen Sy An, Dao Viet Hang, Dao Van Long, Tran Quang Trung, Nguyen Thi Thuy, and Dinh Viet Sang. Neounet: Towards accurate colon polyp segmentation and neoplasm detection. In *Advances in Visual Computing: 16th International Symposium, ISVC 2021, Virtual Event, October 4-6, 2021, Proceedings, Part II*, pages 15–28. Springer, 2021. 1

[21] Youwei Pang, Xiaoqi Zhao, Tian-Zhu Xiang, Lihe Zhang, and Huchuan Lu. Zoom in and out: A mixed-scale triplet network for camouflaged object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2160–2170, 2022. 2, 5, 6, 7

[22] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 5

[23] Ricardo Pérez-de la Fuente, Xavier Delclòs, Enrique Peñalver, Mariela Speranza, Jacek Wierzchos, Carmen Ascaso, and Michael S Engel. Early evolution and ecology of camouflage in insects. *Proceedings of the National Academy of Sciences*, 109(52):21414–21419, 2012. 1

[24] Przemysław Skurowski, Hassan Abdulameer, J Błaszczyk, Tomasz Depta, Adam Kornacki, and P Kozieł. Animal camouflage analysis: Chameleon database. *Unpublished manuscript*, 2(6):7, 2018. 5

[25] Yujia Sun, Geng Chen, Tao Zhou, Yi Zhang, and Nian Liu. Context-aware cross-level fusion network for camouflaged object detection. *arXiv preprint arXiv:2105.12555*, 2021. 1, 2, 6, 7

[26] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International*

conference on machine learning, pages 6105–6114. PMLR, 2019. 5

[27] Haiwen Wang, Xinzhou Wang, Fuchun Sun, and Yixu Song. Camouflaged object segmentation with transformer. In *Cognitive Systems and Information Processing: 6th International Conference, ICCSIP 2021, Suzhou, China, November 20–21, 2021, Revised Selected Papers 6*, pages 225–237. Springer, 2022. 6, 7

[28] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, 2022. 5

[29] Zongwei Wu, Danda Pani Paudel, Deng-Ping Fan, Jingjing Wang, Shuo Wang, Cédric Demonceaux, Radu Timofte, and Luc Van Gool. Source-free depth for object pop-out. *arXiv preprint arXiv:2212.05370*, 2022. 6, 7

[30] Fan Yang, Qiang Zhai, Xin Li, Rui Huang, Ao Luo, Hong Cheng, and Deng-Ping Fan. Uncertainty-guided transformer reasoning for camouflaged object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4146–4155, 2021. 1, 6, 7

[31] Fan Yang, Qiang Zhai, Xin Li, Rui Huang, Ao Luo, Hong Cheng, and Deng-Ping Fan. Uncertainty-guided transformer reasoning for camouflaged object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4146–4155, October 2021. 2

[32] Jianwei Yang, Chunyuan Li, and Jianfeng Gao. Focal modulation networks. *arXiv preprint arXiv:2203.11926*, 2022. 2, 3, 4

[33] Bowen Yin, Xuying Zhang, Qibin Hou, Bo-Yuan Sun, Deng-Ping Fan, and Luc Van Gool. Camoformer: Masked separable attention for camouflaged object detection. *arXiv preprint arXiv:2212.06570*, 2022. 1, 6, 7

[34] Pang Youwei, Zhao Xiaoqi, Xiang Tian-Zhu, Zhang Lihe, and Lu Huchuan. Zoom in and out: A mixed-scale triplet network for camouflaged object detection. *arXiv preprint arXiv:2203.02688*, 2022. 1

[35] Qiang Zhai, Xin Li, Fan Yang, Chenglizhao Chen, Hong Cheng, and Deng-Ping Fan. Mutual graph learning for camouflaged object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12997–13007, 2021. 1, 3, 6, 7

[36] Miao Zhang, Shuang Xu, Yongri Piao, Dongxiang Shi, Shusen Lin, and Huchuan Lu. Preynet: Preying on camouflaged objects. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 5323–5332, 2022. 6, 7

[37] Qiao Zhang, Yanliang Ge, Cong Zhang, and Hongbo Bi. Tprnet: camouflaged object detection via transformer-induced progressive refinement network. *The Visual Computer*, pages 1–15, 2022. 2, 3

[38] Dehua Zheng, Xiaochen Zheng, Laurence T Yang, Yuan Gao, Chenlu Zhu, and Yiheng Ruan. Mffn: Multi-view feature fusion network for camouflaged object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6232–6242, 2023. 6, 7

[39] Yijie Zhong, Bo Li, Lv Tang, Senyun Kuang, Shuang Wu, and Shouhong Ding. Detecting camouflaged object in frequency domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4504–4513, 2022. 7

[40] Hongwei Zhu, Peng Li, Haoran Xie, Xuefeng Yan, Dong Liang, Dapeng Chen, Mingqiang Wei, and Jing Qin. I can find you! boundary-guided separated attention network for camouflaged object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3608–3616, 2022. 1, 6, 7