Camouflaged Object Detection via Scale-Feature Attention and Type-Feature Attention

Yi Liu(✉) and Hui Meng

School of Computer Science and Artificial Intelligence, Aliyun School of Big Data, and School of Software, Changzhou University, Changzhou, China
liuyi0089@gmail.com, menghui199908@outlook.com

Abstract. Camouflaged object detection targets at identifying and segmenting objects hidden in the surroundings. Due to the various shapes and sizes, and highly non-discriminative features of camouflaged objects, it is a challenge for Convolutional Neural Networks (CNNs) to detect them from the background. To tackle the first problem of various shapes and sizes, we propose a Scale-Feature Attention (SFA), which can effectively integrate feature information of different scales, so that the model can comprehensively perceive and understand the visual characteristics of different sizes of camouflaged objects. Additionally, the traditional CNN model is difficult to capture the part-whole relationship of camouflaged objects. To solve the second problem of CNNs' limitation, we propose a Type-Feature Attention (TFA) to integrate contrast from CNNs and part-whole relations from CapsNets, which will improve the identification and object wholeness of camouflaged objects. Experiments on three camouflaged object detection benchmark datasets show that both the proposed SFA and TFA achieve significant performance improvement, which verifies the superiority of the proposed method.

Keywords: Camouflaged object detection · Scale-Feature Attention · Type-Feature Attention · CapsNets

1 Introduction

Camouflaged Object Detection (COD) aims to identify and segment objects with high similarity to the surroundings. Due to the task property, COD has been used in a wide range of applications, such as industrial defect detection [2] and medical image analysis [6]. However, because of the object camouflage, COD is still a challenge that need be solved urgently.

Traditional COD methods [9, 8, 24] that utilize hand-crafted features such as color, intensity, and motion difficultly capture subtle differences between the camouflaged object and the background. In recent years, deep learning has made remarkable progress for the task of COD. CNNs COD networks [32, 20] mostly capture the contrast features to distinguish the camouflaged targets from their surroundings. However, the high similarity between foreground and background makes these methods difficult to identify camouflaged objects accurately. Different from CNNs that aims to dig into contrast semantic, Capsule Networks (CapsNets) explores the part-whole relations to capture the whole object, which will help to detect the whole camouflaged object. Liu et al. [16] firstly explored the possibility of combination of CNNs and CapsNets to the COD task. However, this study made few studies for the primitive integration of these two types of semantics.

To tackle this problem, in the paper, we propose a Type-Feature Attention (TFA) mechanism to integrate features from CNNs and CapsNets. Specifically, the Vision Transformer (ViT) architecture is utilized to implement the dual cross-attention mechanism. On one hand, CNNs features and CapsNets feature are projected into Q vector, and K & V vectors, respectively, in which way CapsNets attend CNNs. On the other hand, CapsNets and CNNs features features are projected into Q, and K & V vectors, respectively, in which way CNNs attend CapsNets. The dual cross-attention mechanism will integrate the contrast of CNNs and part-whole relational property of CapsNets well, which will improve the model ability of identification and object wholeness of camouflaged objects.

Our main contributions can be summarized as follows:

(1) We propose a scale-feature attention to improve the detection of camouflaged objects with various shapes and sizes.

(2) We design a type-feature attention to integrate the contrast from CNNs and object wholeness from CapsNets for camouflaged object detection.

(3) Experiments on three datasets demonstrate the superiority of the proposed method.

2 Related Work

2.1 Camouflaged Object Detection

In recent years, deep learning-based camouflaged object detection has made remarkable progress. Most of the existing camouflaged object detection models extract features based on classical networks [5, 11, 33], and further enhance the features through various strategies to optimize the accurate prediction ability of the concealed object. Recent studies have introduced tasks such as localization [21] and edge detection [7] into it. Lv et al. [21] proposed a ranking-based COD network to simultaneously locate, segment and rank camouflaged objects. He et al. [7] proposed a feature decomposition and edge reconstruction model to decompose foreground and background features through wavelet and learn an accurate edge reconstruction task.

In addition, there are some methods to design camouflaged object detection models by simulating the dynamic vision of predators. Pang et al. [25] proposed ZoomNet, a camouflaged object detection network with three scales, simulating the zoom in and out strategy adopted by humans when observing objects. Jia et al. [10] adopted an iterative multi-stage detection framework, integrating segmentation, magnification, and reiteration strategies to effectively solve the problem of camouflaged object detection. Although the CNN model performs well in object detection tasks, it is limited in global feature capture and remote dependency modeling because of its limited receptive field.

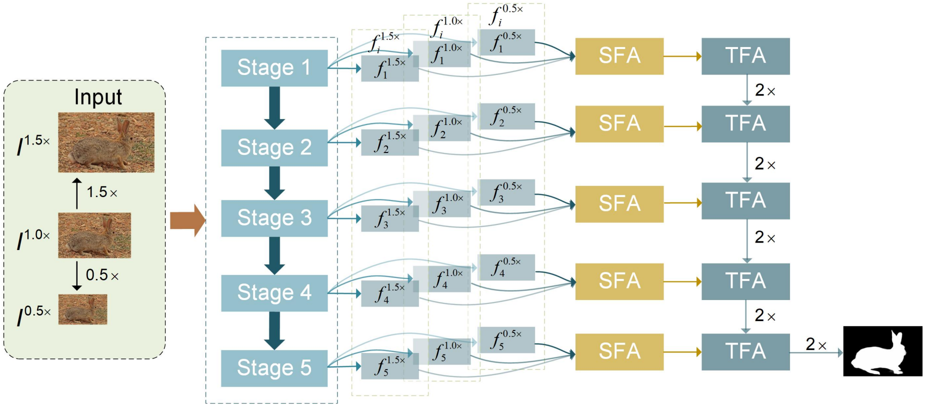

Fig. 1. Overview of the proposed STANet framework. Input features of the three scales are initially extracted by ResNet50, and then input into the SFA for feature interaction and scale aggregation. In the TFA, CNN features and CapsNet features are interacted through Vision Transformer. Then, channel-level feature interaction and layer by layer sampling were carried out in the CFIU to derive the ultimate prediction outcome.

2.2 CapsNet

The initial proposal of capsule networks was made by Geoffrey Hinton in Dynamic Routing between Capsules [26]. Compared with traditional deep networks, capsule networks have stronger ability to recognize complex hierarchical structures and spatial relationships. Based on this advantage, many studies [22, 30, 13] have attempted to explore the potential capabilities of capsule networks. McIntosh et al. [22] proposed a capsule network-based vision text routing mechanism for locating actors and actions from video and text, enabling a more effective video-text integrated localization method. Yu et al. [30] proposed the Inverse Graphics Capsule Network (IGC-Net) for acquiring hierarchical 3D facial representations through the analysis of extensive unlabeled image datasets.

Due to the excellent performance of CapsNets, they have been successfully applied to object detection tasks [18, 17, 19]. In terms of salient object detection, Liu et al. [14] proposed a Belief Capsule Network (BCNet) for deep unsupervised salient object detection, solving the problems of poor object integrity and low-quality pseudo labels. In terms of camouflaged object detection, Liu et al. [16] integrated CapsNets into an encoder-decoder architecture of two-stage to tackle the task of COD. Different from previous works, our method integrates the advantages of CNN and CapsNet, and aims to deeply mine the most salient features of camouflaged targets by utilizing the exceptional global modeling capability of Vision Transformer.

3 Methodology

Fig. 1 illustrates the general framework of STANet. Given a camouflaged image, we first use the backbone network ResNet50 to extract a series of features, and then use the Scale-Feature Attention (SFA) to integrate feature maps of different scales and add the joint attention mechanism to explore the noteworthy parts of the channel and spatial dimension. In addition, Type-Feature Attention (TFA) feeds the features processed by CNN and CapsNet into the interactive Vision Transformer. By interacting these two types of features, we can achieve the fusion of contrast and object wholeness. Finally, the Channel Feature Interaction Unit(CFIU) further grouped the features and used the iterative approach to get the final segmentation prediction.

3.1 Scale-Feature Attention

The module structure of SFA is shown in Fig. 2. As shown in the figure, first of all, the module receives 64-channel inputs from different scales $f_i^l \in \mathbb{R}^{C\times H_1\times W_1}$, $f_i^m \in \mathbb{R}^{C\times H\times W}$ and $f_i^s \in \mathbb{R}^{C\times H_2\times W_2}$, where l, m and s represent feature map sizes of 1.5×, 1.0×, and 0.5×, respectively. Then the features of various scales are unified into the same scale size through downsampling and upsampling operations. To effectively capture the feature information across various scales, we first fuse the features at adjacent scales and calculate the corresponding attention map. We then multiply the features of adjacent scales with the attention map separately to obtain the weighted features:

$$\begin{aligned} f_i^{lma} &= f_i^l \odot \mathbf{A}_1^l + f_i^m \odot \mathbf{A}_1^m, \\ f_i^{msa} &= f_i^m \odot \mathbf{A}_2^m + f_i^s \odot \mathbf{A}_2^s, \end{aligned} \tag{1}$$

where $\odot$ means element-by-element multiplication, $\{\mathbf{A}_1^{l,m}, \mathbf{A}_2^{m,s}\}$ represent the attention maps of features at adjacent scales. The attention map of adjacent scale features is calculated as follows:

$$\begin{aligned} \mathbf{A}_1^{l,m} &= \sigma(CBR(Conv(Cat(f_i^l, f_i^m)))), \\ \mathbf{A}_2^{m,s} &= \sigma(CBR(Conv(Cat(f_i^m, f_i^s)))), \end{aligned} \tag{2}$$

where $CBR(\cdot)$ represents the Conv-BN-Relu layer, $Conv(\cdot)$ represents a 3×3 convolution, $C\mathrm{at}(\cdot)$ denotes the concatenation operation and $\sigma(\cdot)$ indicates softmax operation. In order to obtain more comprehensive feature information, we also integrate the features of three scales, and the calculation method is similar to the above:

$$f_i^{lmsa} = f_i^l \odot \mathbf{A}_3^l + f_i^m \odot \mathbf{A}_3^m + f_i^s \odot \mathbf{A}_3^s, \tag{3}$$

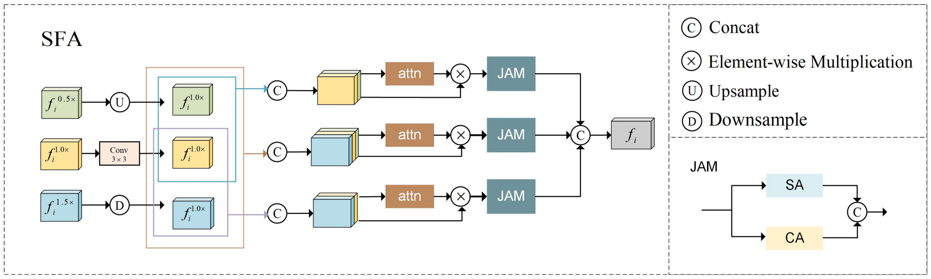

Fig. 2. Details of the proposed SFA.

where $\mathbf{A}_3^{l,m,s}$ represents the attention map of the features at three scales. The attention map calculation process of the three scale features is similar to that of the adjacent scale features:

$$\mathbf{A}_3^{l,m,s} = \sigma(CBR(Conv(Cat(f_i^l, f_i^m, f_i^s)))). \tag{4}$$

At this point, we have obtained weighted features that not only enhance feature representation, but also interact feature information at different scales to explore multi-level structure and semantics in the image.

Finally, the joint attention module receives the weighted features for calculating both spatial and channel attention, so as to further optimize the representation ability and perception range of features, and obtain the optimized features. Then the processed features are concatenated to acquire the final output f_i:

$$f_i = CBR(Cat(JA(f_i^{lma}), JA(f_i^{msa}), JA(f_i^{lmsa}))), \tag{5}$$

where $JA(\cdot)$ donates the joint attention containing spatial and channel attention. Through the design of multi-scale feature fusion and attention mechanism, the SFA can effectively leverage the benefits of features across various scales and capture more comprehensive feature information.

3.2 Type-Feature Attention

To maximize the utilization of the contrast of CNNs and object wholeness of CapsNets, we designed a type-feature attention module of CNN and CapsNet to integrate their features and improve the ability of accurately identifying camouflaged targets.

A. CNN and CapsNet Feature Interaction. As shown in Fig. 3, firstly, we obtain the CNN feature $f_c \in \mathbb{R}^{C\times H\times W}$ from the SFA, and then generate the capsule network feature $f_d \in \mathbb{R}^{C\times H\times W}$ using the lightweight designed DCR [15]. In order to realize the interaction and fusion of these two types of features through the Vision Transformer, we first segment the CNN features and the

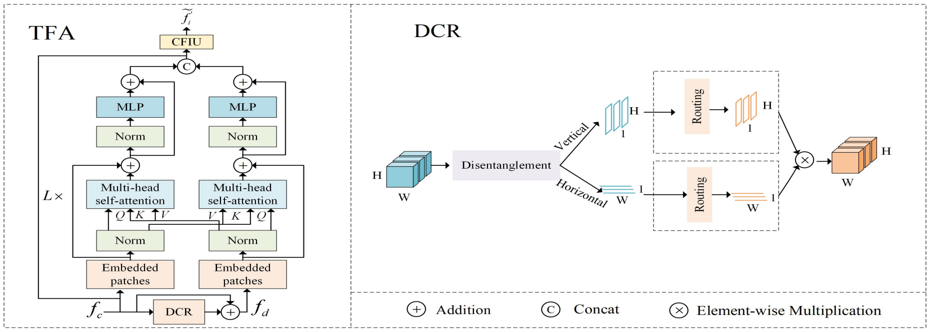

Fig. 3. Details of the proposed TFA with CFIU [25] and DCR modules.

capsule network features into a series of flat 2D patches $\{\mathbf{x}_p^l \in \mathbb{R}^{P^2 \times C} | p = 1, ..., N\}$, where N represents the numerical value of patches and $l \in \{c, d\}$. Then, the image block embedded $\mathbf{x}_p^l$ and position encoded $\mathbf{E}_{pos}^l$ constitute the embedded input vector $\mathbf{z}_0^l$, which is specifically expressed as:

$$\mathbf{z}_0^l = [\mathbf{x}_1^l \mathbf{E}^l; \mathbf{x}_2^l \mathbf{E}^l; ...; \mathbf{x}_N^l \mathbf{E}^l] + \mathbf{E}_{pos}^l, \tag{6}$$

where $\mathbf{E}^l \in \mathbb{R}^{(P^2 \cdot C) \times D}$ refers to the patch embedding projection, and $\mathbf{E}_{pos}^l \in \mathbb{R}^{(N+1) \times D}$ indicates the position embedding. $\mathbf{z}_0^l$ is then fed into a transformer encoder consisting of a serial stack of L transformer encoder blocks. It is composed of a self-attention mechanism with multiple heads (MSA) and a multi-layer perceptron (MLP), which are calculated as follows:

$$\begin{cases} \mathbf{z}_i^{l'} = MSA(\mathbf{z}_{i-1}^l, \mathbf{z}_{i-1}^{\bar{l}}) + \mathbf{z}_{i-1}^l, \\ \mathbf{z}_i^l = MLP(LN(\mathbf{z}_i^{l'})) + \mathbf{z}_i^{l'}, \end{cases} \text{i = 1,..., L} \tag{7}$$

where $LN(\cdot)$ is the layer normalization. The symbol l and $\bar{l}$ represent two different variants belonging to the set $\{c, d\}$.

In order to interact with these two types of features, we exchange information from one type of feature with the other. Specifically, CNN features and capsule network features are first added to layer normalization and multi-head attention, and Query, Key and Value are calculated respectively, and then an exchange of Key and Value occurs between the two types of features. Finally, in order to retain the valid information of the original feature, we residual link the feature that has passed the multi-head attention with the original feature after the interaction. Specifically defined as:

$$MSA(\mathbf{z}_i^l, \mathbf{z}_i^{\bar{l}}) = soft\max(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V}, \tag{8}$$

where d_k is the dimension of $\mathbf{K}$. The definitions of $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$ are as follows:

$$(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = (LN(\mathbf{z}_i^l)\mathbf{W}_Q, LN(\mathbf{z}_i^{\bar{l}})\mathbf{W}_K, LN(\mathbf{z}_i^{\bar{l}})\mathbf{W}_V), \tag{9}$$

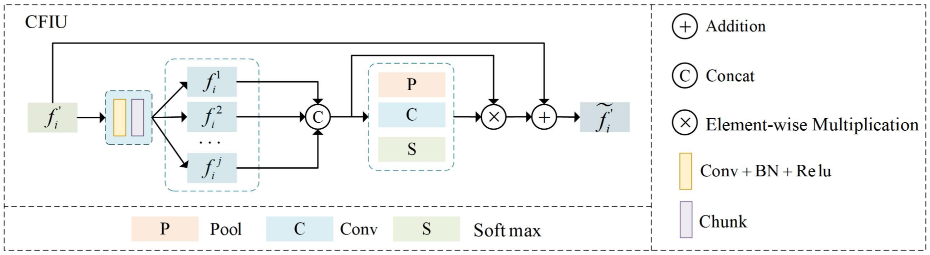

Fig. 4. Details of the CFIU.

where $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{D \times D}$ are projection matrix of the fully connected layer and D is embedding dimension.

Finally, the obtained CNN features and CapsNet features $\mathbf{z}_L^l$, $\mathbf{z}_L^{\bar{l}}$ from the patch sequence should be reshaped to match the desired feature format. Then, further concatenate the reshaped features to get the final interactive feature $f_i^{'}$:

$$f_i^{'} = Conv(Cat(Cat(R(\mathbf{z}_L^l), R(\mathbf{z}_L^{\bar{l}})), f_c)), \tag{10}$$

where $Conv(\cdot)$ represents a 3×3 convolution and $C\mathrm{at}(\cdot)$ denotes the concatenation operation. $R(\cdot)$ donates a reshaping operation to reshape patch sequences with $D \times \frac{HW}{P^2}$ to feature with $D \times H \times W$.

B. Channel Feature Interaction Unit. The final fusion of the interaction features with type feature attention is performed using the CFIU. As shown in Fig. 4, We first input $f_i^{'}$ into the Conv+BN+Relu layer and perform convolution, batch normalization, and activation operations to extract more discriminative and expressive feature representations. Considering the importance of spatial information of features and channel interaction, we split the current features into J-level features, denoted as f_i^j:

$$f_i^j = Chunk(\mathrm{Re}lu(BN(Conv(f_i^{'})))),j \in (1, ..., J), \tag{11}$$

where $Conv(\cdot)$ represents a 3×3 convolution, $BN(\cdot)$ donates a batch normalization layer and $Relu(\cdot)$ represents the activation layer. To ensure comprehensive retention of the information pertaining to each level feature and avoid information loss during transmission, we pass the upper level feature f_i^{j-1} to the lower level feature f_i^j, so that the upper level feature and the lower level can have channel-based interaction. This can promote the information flow and mutual integration of different levels of features. Specifically expressed as:

$$f_i^{j+1} = Conv(Cat(f_i^j, Conv(f_i^{j+1}))),j \in (1, ..., J-1). \tag{12}$$

Next, each level of features are concatenated to obtain f_i^c, the calculation process is as follows:

$$f_i^c = Conv(Cat(f_i^1, ..., f_i^j)), j \in (1, ..., J). \tag{13}$$

The concatenated features are then fed into the Pool+Conv+Softmax layer for weight calculation. Then, the features are weighted and finally added with the initial input feature $f_i^{'}$ to obtain the output $\widetilde{f_i^{'}}$ of CFIU, which is expressed as follows:

$$\widetilde{f_i^{'}} = S(Conv(P(f_i^c))) \otimes f_i^c + f_i^{'}, \tag{14}$$

where $P(\cdot)$ stands for adaptive averaging pooling, $Conv(\cdot)$ stands for 3×3 convolution, and $S(\cdot)$ stands for softmax activation function.

Finally, as shown in Fig. 1, the final prediction is obtained by upsampling and layer-by-layer fusion of the features acquired by CFIU.

3.3 Loss Functions

The proposed STANet consists of two loss functions, binary cross entropy loss (BCEL) and uncertainty-aware loss (UAL). The binary cross entropy loss function is the most commonly used loss function at present, and its mathematical expression is usually as follows:

$$L_{BCEL}^n = -\mathbf{G}_n \ln(\mathbf{P}_n) - (1 - \mathbf{G}_n) \ln(1 - \mathbf{P}_n), \tag{15}$$

where $\mathbf{P}_n$ and $\mathbf{G}_n$ are the values of the prediction map and ground truth at pixel n.

Because most of the camouflaged objects have similar features to the surrounding environment, the model is prone to misjudgment when detecting the camouflaged objects. To solve this problem, we also introduce another loss function UAL, which enhances the model's ability to recognize the uniqueness of camouflaged objects. Finally, the overall loss function of STANet can be formulated as:

$$L_{total} = L_{BCEL}^n + \lambda L_{UAL}, \tag{16}$$

where λ is the equilibrium coefficient and the UAL loss we use adjusts λ dynamically based on the cosine strategy.

4 Experiment

4.1 Implementation Details

Our model is built on the pytorch, with ResNet50 selected as the backbone network, which has been pretrained on the ImageNet dataset. The base size of the input image is 512×512, and data enhancement techniques are used to achieve multi-scale inputs. We use the optimizer SGD with momentum of 0.9 and weight decay of 0.0005, and initial learning rate of 0.05 to train the model. The model was trained for a total of 40 epochs in the training process, with a batch size of 12 employed. On a single V100 block, it took about 8 hours to train, and the test image size was also adjusted to 512×512.

Table 1. Performance of different methods on three benchmarks. The symbols "↑" and "↓" suggest that higher and lower values are preferable, correspondingly. The optimal performance for each group is indicated by **bold**.

Method	CHAMELEON-Test				COD10K-Test				NC4K-Test			
	$S_m \uparrow$	$E_m \uparrow$	$F_\beta \uparrow$	$MAE \downarrow$	$S_m \uparrow$	$E_m \uparrow$	$F_\beta \uparrow$	$MAE \downarrow$	$S_m \uparrow$	$E_m \uparrow$	$F_\beta \uparrow$	$MAE \downarrow$
SINet	0.872	0.946	0.827	0.034	0.766	0.874	0.679	0.043	0.808	0.883	0.769	0.058
SLSR	0.890	0.948	0.841	0.030	0.804	0.892	0.715	0.037	0.840	0.907	0.804	0.048
MGL-R	0.893	0.941	0.833	0.031	0.814	0.890	0.710	0.035	0.833	0.893	0.782	0.053
PFNet	0.882	0.945	0.828	0.033	0.800	0.890	0.701	0.040	0.829	0.898	0.784	0.053
UJSC	0.891	0.955	0.847	0.030	0.809	0.891	0.721	0.035	0.842	0.907	0.806	0.047
C2FNet	0.888	0.946	0.844	0.032	0.813	0.900	0.723	0.036	0.838	0.904	0.795	0.049
UGTR	0.888	0.940	0.819	0.031	0.817	0.890	0.711	0.036	0.839	0.899	0.787	0.052
ZoomNet	0.902	0.958	0.864	**0.023**	0.838	0.911	0.766	**0.029**	0.853	0.912	0.818	**0.043**
FEDER	0.887	0.954	0.868	0.030	0.822	0.905	0.768	0.032	0.847	**0.915**	0.833	0.044
Ours	**0.917**	**0.970**	**0.899**	**0.023**	**0.856**	**0.921**	**0.809**	**0.029**	**0.863**	**0.915**	**0.841**	0.044

4.2 Datasets and evaluation metrics

We evaluated experiments on three widely used public COD datasets, including CHAMELEON [27], COD10K [5], and NC4K [21]. In this work, we select 3040 images from COD10K and 1000 images from CAMO as the training set, and the rest of the camouflaged images are used for testing. We used four commonly used evaluation metrics, namely S-measure(S_m) [3], mean absolute error(MAE), F-measure(F_β) [1] and E-measure(E_m) [4]. It should be emphasized that for F_β, E_m and S_m, superior performance is achieved with higher values. For MAE, the performance improves as the value decreases.

4.3 Comparsion with state-of-the-art methods

We contrasted the proposed model with various advanced methods, such as SINet [5], SLSR [21], MGL-R [31], PFNet [23], UJSC [12], C2FNet [28], UGTR [29], ZoomNet [25] and FEDER [7]. To guarantee a fair comparison, all methods utilize the camouflaged prediction map provided by the authors.

1. Quantitative Comparison: Table 1 shows the performance of various methods on three public datasets. It is obviously that our method surpasses the other cutting-edge methods in terms of all three datasets. Specifically, compared to ZoomNet, which exhibits the second highest average performance, S_m, E_m, F_β are improved by 1.5%, 1.2%, 3.5% respectively on the CHAMELEON dataset. On the COD10k dataset, S_m, E_m, F_β are improved by 1.8%, 1.0%, 4.3% respectively. On the NC4K dataset, S_m, E_m, F_β are increased by 1.0%, 0.3%, 2.3% respectively.

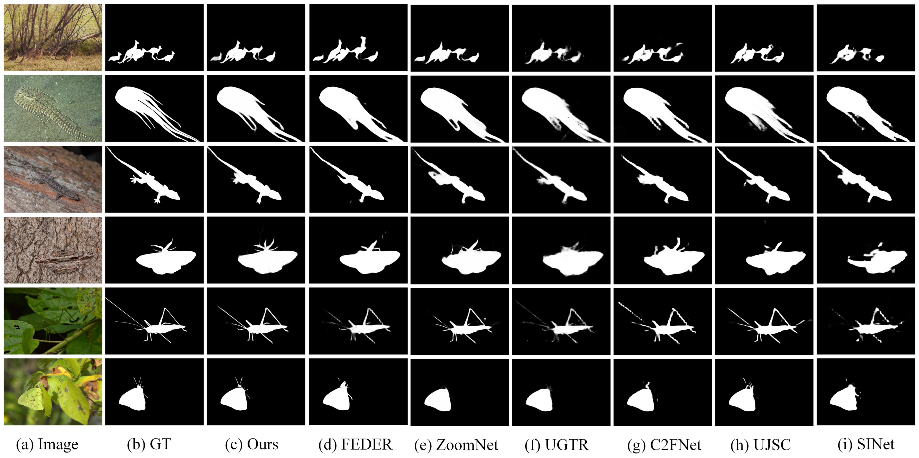

Fig. 5. Visual comparison of our method with the current state of the art methods. (a) Images, (b) GT, (c) Our method, (d)-(i) Other advanced methods include: FEDER, ZoomNet, UGTR, C2FNet, UJSC and SINet.

2. Visual Comparison: The visual comparison of our method with other methods is illustrated in Fig. 5. As can be seen from the figure, our model shows more optimal performance when dealing with complex scenes. This is due to our model's ability to effectively combine the contrast of CNNs and object wholeness of CapsNets, and capture different levels of detail through its multi-scale design, thereby leading to more accurate predictions and superior visual outputs.

4.4 Ablation study

In order to validate the effectiveness of each proposed component module, we conduct ablation studies on CHAMELEON, COD10K, and NC4K datasets. B denotes the baseline model, S denotes the Scale-Feature Attention, T denotes the type-feature attention part of TFA, and C denotes the channel feature interaction part of TFA.

Effectiveness of SFA. As shown in table 2, in order to demonstrate the efficacy of SFA, only this module is retained and other components of STANet model are removed (NO.②). Compared with baseline NO.①, our SFA has improved and enhanced all indicators of the three datasets. The results verify the effectiveness of the SFA in detecting camouflaged objects of different sizes, and further enable the model to fully and deeply mine more abundant feature information.

Effectiveness of TFA. To investigate the influence of TFA on model performance, we performed ablation experiments on the TFA. As shown in table 2, when we added type-feature attention to the model (NO.③), compared with

Table 2. Ablation study on the effect of different modules on CHAMELEON, COD10K and NC4K. The symbols "↑" and "↓" suggest that higher and lower values are preferable, correspondingly. The optimal performance for each group is indicated by **bold**.

No.	CHAMELEON-Test				COD10K-Test				NC4K-Test			
	$S_m\uparrow$	$E_m\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_m\uparrow$	$E_m\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$	$S_m\uparrow$	$E_m\uparrow$	$F_\beta\uparrow$	$MAE\downarrow$
①B	0.893	0.947	0.869	0.031	0.826	0.903	0.0.763	0.034	0.845	0.906	0.819	0.050
②B+S	0.906	0.964	0.889	0.025	0.842	0.913	0.791	0.032	0.851	0.910	0.825	0.048
③B+S+T	0.909	0.956	0.891	0.027	0.847	0.915	0.796	0.030	0.855	0.914	0.834	0.045
④B+S+T+C	**0.917**	**0.970**	**0.899**	**0.023**	**0.856**	**0.921**	**0.809**	**0.029**	**0.863**	**0.915**	**0.841**	**0.044**

NO.②, all performance indicators of the model were significantly improved. This experimental result fully proves that type-feature attention can effectively integrate the advantages of different networks, realize the integration of spatial details and object integrity. In order to demonstrate the efficacy of CFIU, we add CFIU (NO.④) on the basis of NO.③. As shown in table 2, the significant improvement of various evaluation indicators on the three datasets indicates that the module can optimize the prediction results of the network, which is attributed to the full use of the hierarchical relationship and channel interaction of features, and further improve the expression ability of features.

5 Conclusion

In this paper, we propose a COD network based on the interaction of CNN and CapsNet features. Specifically, we design a Scale-Feature Attention at first, which aims to fully capture and integrate multi-scale features to improve the model's overall perception of feature information. Then, we propose a Type-Feature Attention, which cleverly integrates two different network features and realizes the complementary and enhanced feature information. Finally, we use Channel Feature Interaction Unit to effectively interact and integrate features at the channel level. The proposed STANet shows excellent performance on three widely used datasets, surpassing cutting-edge methods through numerous experiments.

Acknowledgements. This work was supported in part by the National Natural Science Foundation of Jiangsu Province under Grant BK20221379; the National Natural Science Foundation of China under Grant 62306048; the Changzhou University CNPC-CZU Innovation Alliance under Grant CCIA2023-01; the Changzhou Leading Innovative Talent Introduction and Cultivation Project under Grant 20221460; and by the Changzhou Applied Basic Research Fund Project Grant CQ20230092 and CJ20235036.

References

[1] Achanta, R., Hemami, S., Estrada, F., Susstrunk, S.: Frequency-tuned salient region detection. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 1597–1604. IEEE (2009)

[2] Bhajantri, N.U., Nagabhushan, P.: Camouflage defect identification: a novel approach. In: 9th International Conference on Information Technology (ICIT'06). pp. 145–148. IEEE (2006)

[3] Fan, D.P., Cheng, M.M., Liu, Y., Li, T., Borji, A.: Structure-measure: A new way to evaluate foreground maps. In: Proceedings of the IEEE international conference on computer vision. pp. 4548–4557 (2017)

[4] Fan, D.P., Gong, C., Cao, Y., Ren, B., Cheng, M.M., Borji, A.: Enhanced-alignment measure for binary foreground map evaluation. arXiv preprint arXiv:1805.10421 (2018)

[5] Fan, D.P., Ji, G.P., Sun, G., Cheng, M.M., Shen, J., Shao, L.: Camouflaged object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2777–2787 (2020)

[6] Fan, D.P., Ji, G.P., Zhou, T., Chen, G., Fu, H., Shen, J., Shao, L.: Pranet: Parallel reverse attention network for polyp segmentation. In: International conference on medical image computing and computer-assisted intervention. pp. 263–273. Springer (2020)

[7] He, C., Li, K., Zhang, Y., Tang, L., Zhang, Y., Guo, Z., Li, X.: Camouflaged object detection with feature decomposition and edge reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22046–22055 (2023)

[8] Hou, J.Y.Y.H.W., Li, J.: Detection of the mobile object with camouflage color under dynamic background based on optical flow. Procedia Engineering **15**, 2201–2205 (2011)

[9] Huerta, I., Rowe, D., Mozerov, M., Gonzàlez, J.: Improving background subtraction based on a casuistry of colour-motion segmentation problems. In: Iberian Conference on Pattern Recognition and Image Analysis. pp. 475–482. Springer (2007)

[10] Jia, Q., Yao, S., Liu, Y., Fan, X., Liu, R., Luo, Z.: Segment, magnify and reiterate: Detecting camouflaged objects the hard way. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4713–4722 (2022)

[11] Le, T.N., Nguyen, T.V., Nie, Z., Tran, M.T., Sugimoto, A.: Anabranch network for camouflaged object segmentation. Computer vision and image understanding **184**, 45–56 (2019)

[12] Li, A., Zhang, J., Lv, Y., Liu, B., Zhang, T., Dai, Y.: Uncertainty-aware joint salient object and camouflaged object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10071–10081 (2021)

[13] Liu, Y., Cheng, D., Zhang, D., Xu, S., Han, J.: Capsule networks with residual pose routing. IEEE Transactions on Neural Networks and Learning Systems (2024)

[14] Liu, Y., Dong, X., Zhang, D., Xu, S.: Deep unsupervised part-whole relational visual saliency. Neurocomputing **563**, 126916 (2024)

[15] Liu, Y., Zhang, D., Liu, N., Xu, S., Han, J.: Disentangled capsule routing for fast part-object relational saliency. IEEE Transactions on Image Processing **31**, 6719–6732 (2022)

[16] Liu, Y., Zhang, D., Zhang, Q., Han, J.: Integrating part-object relationship and contrast for camouflaged object detection. IEEE Transactions on Information Forensics and Security **16**, 5154–5166 (2021)

[17] Liu, Y., Zhang, D., Zhang, Q., Han, J.: Part-object relational visual saliency. IEEE transactions on pattern analysis and machine intelligence **44**(7), 3688–3704 (2021)

[18] Liu, Y., Zhang, Q., Zhang, D., Han, J.: Employing deep part-object relationships for salient object detection. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1232–1241 (2019)

[19] Liu, Y., Zhou, L., Wu, G., Xu, S., Han, J.: Tcgnet: Type-correlation guidance for salient object detection. IEEE Transactions on Intelligent Transportation Systems (2023)

[20] Luo, N., Pan, Y., Sun, R., Zhang, T., Xiong, Z., Wu, F.: Camouflaged instance segmentation via explicit de-camouflaging. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17918–17927 (2023)

[21] Lv, Y., Zhang, J., Dai, Y., Li, A., Liu, B., Barnes, N., Fan, D.P.: Simultaneously localize, segment and rank the camouflaged objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11591–11601 (2021)

[22] McIntosh, B., Duarte, K., Rawat, Y.S., Shah, M.: Visual-textual capsule routing for text-based video segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9942–9951 (2020)

[23] Mei, H., Ji, G.P., Wei, Z., Yang, X., Wei, X., Fan, D.P.: Camouflaged object segmentation with distraction mining. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8772–8781 (2021)

[24] Pan, Y., Chen, Y., Fu, Q., Zhang, P., Xu, X., et al.: Study on the camouflaged target detection method based on 3d convexity. Modern Applied Science **5**(4), 152 (2011)

[25] Pang, Y., Zhao, X., Xiang, T.Z., Zhang, L., Lu, H.: Zoom in and out: A mixed-scale triplet network for camouflaged object detection. In: Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. pp. 2160–2170 (2022)

[26] Sabour, S., Frosst, N., Hinton, G.E.: Dynamic routing between capsules. Advances in neural information processing systems **30** (2017)

[27] Skurowski, P., Abdulameer, H., Błaszczyk, J., Depta, T., Kornacki, A., Kozieł, P.: Animal camouflage analysis: Chameleon database. Unpublished manuscript **2**(6), 7 (2018)

[28] Sun, Y., Chen, G., Zhou, T., Zhang, Y., Liu, N.: Context-aware cross-level fusion network for camouflaged object detection. arXiv preprint arXiv:2105.12555 (2021)

[29] Yang, F., Zhai, Q., Li, X., Huang, R., Luo, A., Cheng, H., Fan, D.P.: Uncertainty-guided transformer reasoning for camouflaged object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4146–4155 (2021)

[30] Yu, C., Zhu, X., Zhang, X., Zhang, Z., Lei, Z.: Graphics capsule: Learning hierarchical 3d face representations from 2d images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20981–20990 (2023)

[31] Zhai, Q., Li, X., Yang, F., Chen, C., Cheng, H., Fan, D.P.: Mutual graph learning for camouflaged object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12997–13007 (2021)

[32] Zhong, Y., Li, B., Tang, L., Kuang, S., Wu, S., Ding, S.: Detecting camouflaged object in frequency domain. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4504–4513 (2022)

[33] Zhu, H., Li, P., Xie, H., Yan, X., Liang, D., Chen, D., Wei, M., Qin, J.: I can find you! boundary-guided separated attention network for camouflaged object detection. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 3608–3616 (2022)