Fast Camouflaged Object Detection via Multi-Scale Feature Enhanced Network

Bingqin Zhou
Hangzhou Dianzi University
Kun Yang
yangkun@hdu.edu.cn
Hangzhou Dianzi University
Zhigang Gao
Hangzhou Dianzi University

Research Article

Keywords: Camouflaged object detection, Multi-scale feature enhancement, Convolutional neural network, Fast detection

Posted Date: December 11th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3708075/v1

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at Signal, Image and Video Processing on February 28th, 2024. See the published version at https://doi.org/10.1007/s11760-024-03051-1.

Fast Camouflaged Object Detection via Multi-Scale Feature Enhanced Network

Bingqin Zhou[1], Kun Yang[1,2*], Zhigang Gao[1]

[1]School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, 310018, China.
[2] Key Laboratory of Brain Machine Collaborative Intelligence of Zhejiang Province, Hangzhou, 310018, China.

*Corresponding author(s). E-mail(s): yangkun@hdu.edu.cn;
Contributing authors: 222050260@hdu.edu.cn;

Abstract

The aim of camouflaged object detection (COD) is to identify objects that are hidden or camouflaged in the visual scene. Since camouflaged objects have fuzzy boundaries and are very similar to their surroundings, the task of COD, especially multi-scale COD, is still challenging. Based on ERRNet, we proposed a multi-scale feature enhanced network (MSFENet). Specifically, we have developed a multi-scale feature enhancement module (MFEM), which adopts a coarse-to-fine manner to improve the ability of a single layer to represent multi-scale information. This module can extract more complete large-scale target feature information and retain much more small-scale target feature information and less regional background information. The experimental results on publicly available datasets show that our proposed MSFENet outperforms 10 mainstream methods. The ablation studies show that the proposed module is effective in improving the detection performance of multi-scale camouflaging objects and improving the overall performance. Compared with ERRNet, the average $\boldsymbol{S_\alpha}$, $\boldsymbol{E_\phi}$ and $\boldsymbol{F_\beta^w}$ scores of the MSFENet are 1.2%, 0.6% and 2.5% higher for the multi-scale COD task. In addition, the proposed MSFENet can be directly used for real-time detection due to its fast inference capability (i.e. 75.3 frames per second).

Keywords: Camouflaged object detection, Multi-scale feature enhancement, Convolutional neural network, Fast detection

1 Introduction

The aim of camouflaged object detection (COD) is to identify and segment objects that are concealed within visual scenes. Compared with other tasks (ordinary object detection [16, 23] and salient object detection (SOD) [21, 30]), camouflaged object detection is more challenging because the camouflaged object is highly similar to the background in shape, texture, color, etc. Moreover, the visual recognition between its edge and the surrounding environment is extremely low [25], as shown in the Fig. 1. Camouflaged object detection technology has many potential applications, such as covid-19 lung infection segmentation [7], polyp segmentation [6], military multiple camouflaged pattern design [9], locust detection [34], entertainment art [2], etc.

Fig. 1 Examples of camouflaged object, G_c is true mask map.

Traditional camouflaged object detection methods mainly describe the camouflage target by making manual features (such as shape texture [24], edge [33], 3D [20] convexity, etc.). However, when encountering complex scenes, these traditional methods usually have the problems of long feature extraction time, poor mobility and low detection performance. In recent years, with the emergence of large-scale camouflaged object data sets [5] and the development of deep learning, researchers have proposed many deep learning based camouflaged object detection models [5, 12, 26], whose performance far exceeds traditional methods. However, the existing camouflaged object detection models rarely consider the heuristics of biological vision research [19], ignore the role of weak boundary cues and global feature cues, and do not model well the cross-comparison stage between potential camouflaged objects and the surrounding environment. To solve the above problems, researchers proposed ERRNet [12], which enables the camouflaged object to be detected more accurately. However, ERRNet performs poorly in detecting multi-scale camouflaged objects (see Table 4). The difficulty in detecting multi-scale camouflaged objects is that multi-scale camouflaged objects have very large scale changes [21]. The current solution is to extract multi-scale features so that the model can cope with scale changes and improve performance.

At present, the common multi-scale feature extraction modules (PPM [35], FAM [17], ASPP [1], RFB [31], etc.) adopt parallel structures and expand receptive fields through pooled layers of different sizes or convolutional layers with different dilation rates to extract multi-scale features. However, microstructural information and small target feature information may be lost in the pooling layer with large step length

and size, and the convolutional layers utilizing a high dilation rate may cause the extracted small target feature to contain too many features of the surrounding region, which has a negative impact on the overall performance. The detection of multi-scale camouflaged objects is still a challenge.

Based on ERRNet, we proposed a multi-scale feature enhanced network (MSFENet), where a multi-scale feature enhanced module (MFEM) has been designed to handle with the input features serially and characterize multi-scale features from coarse to fine. Specifically, the MFEM module consists of a serial structure with multiple branches and a channel attention mechanism. Different branches include distinct quantities of sub-sampling layers (i.e. max-pooling layer) and a receptive field extension module (RFEM). The number of sub-sampling layers is reduced by branch, so that more microstructural information and small target feature information can be retained. The RFEM module uses asymmetric convolution and dilatational convolution with lower dilation rate, which can enlarge the receptive field and reduce the extraction of regional background features around small target features. The channel attention mechanism can highlight channels that are highly responsive to camouflaged objects.

On three publicly available COD datasets, we validated MSFENet's performance from multiple perspectives and compared it with 10 mainstream models. Experimental results showed that the overall performance of MSFENet is better than those of the popular methods. Compared with the baseline method of ERRNet, our method has higher mean S_α, E_ϕ and F_β^W scores on the whole Test data (ALL-Test). Specifically, they have relative improvements of 1.7%, 1.3%, 2.8%, and absolute improvements of 1.3%, 1.1%, 1.9%, respectively. Furthermore, our method also has higher mean S_α, E_ϕ and F_β^W scores on the multi-scale target test subset (MT-Test). Specifically, they have relative improvements of 1.6%, 0.7%, 3.9%, and absolute improvements of 1.2%, 0.7%, 2.5%, respectively. These results showed that our method has a better performance of multi-scale camouflaged object detection. Further, the ablation experiments showed that the modules of MFEM and RFEM are effective in improving the overall performance. Our major contributions can be summed up as follows:

1. In order to solve the problem of poor multi-scale camouflaged object detection, we proposed a multi-scale feature enhanced network (MSFENet). Experimental results showed that the overall performance of MSFENet is better than that of ten mainstream methods. What's more, the MSFENet has outstanding detection performance for multi-scale camouflage objects and can be directly used for real-time detection due to its high inference speed (i.e. 75.3 frames per second).
2. We designed a multi-scale feature enhancement module (MFEM), which improves the ability of a single layer to refine multi-scale information in a multi-granularity way. MFEM can extract more complete large-scale target feature information and retain more small target feature information with less regional background information.

2 Related works

2.1 Camouflaged object detection

Traditional camouflaged object detection methods mainly detect the camouflaged object through various features made by hand (such as edge shape texture [24], edge [33], 3D [20] convexity, etc.). Wei et al. [32] used the 45° search algorithm to calculate the co-occurrence matrix of the camouflaged surface (target), and developed a texture similarity metric to evaluate the effectiveness of the camouflage. Yan et al. [11] proposed a method based on normalized gray aggregate histogram to evaluate the camouflage effect of the edge between the hidden target and the background by using the gray spatial distribution of the edge. Pan et al. [20] proposed a method based on 3D concavity to identify camouflaged objects in images. However, the handcrafted features have limited expressive ability. When COD encounters challenging tasks (e.g., the foreground is highly similar to the background), the detection performance of traditional methods is severely degraded.

Recently, many deep learning-based COD methods have been developed. Fan et al. (SINet) [5] constructed a novel dataset (COD10K) for training and proposed the first COD model inspired by animal hunting that can achieve accurate identification. Sun et al. (C2FNet) [26] proposed a new COD model based on deep learning to mine abundant contextual information and effectively integrate information between layers to achieve good performance. Zhu et al. (TINet) [38] designed a new texture perception refinement model to amplify the subtle texture differences between the camouflaged area and the surrounding environment. Fan et al. (PraNet] [6] obtained a good recognition effect by establishing the relationship between the camouflaged area and the boundary cues. Zhuge et al. [39] proposed CubeNet, where an X-shaped connection is constructed to perform feature fusion through a multi-input multi-output structure to refine features and improve performance. Existing models of camouflaged object detection rarely consider the heuristics of biological vision research [19], ignore the role of weak boundary cues and global feature cues, and do not well model the cross-comparison stage between potential camouflaged objects and the surrounding environment. Recently, Ji et al. [12] tried to simulate the process by which the human visual system cross-compares objects and proposed an edge-based reversible recalibration network (ERRNet) to improve the accuracy of recognition. However, ERRNet performs poorly when detecting multi-scale camouflaged objects. Multi-scale camouflaged object detection remains a challenge.

2.2 Multi-scale feature extraction

In recent years, people have realized the importance of multi-scale feature extraction. A standard convolutional layer can solely capture information at a single scale, but the target often contains multiple scales, so extracting multi-scale features is conducive to improving the performance of the network. Lin et al. [15] proposed the feature pyramid network (FPN), which constructs multiple layers of features of different scales to detect targets of different scales. Wang et al. (SRM) [27] used the

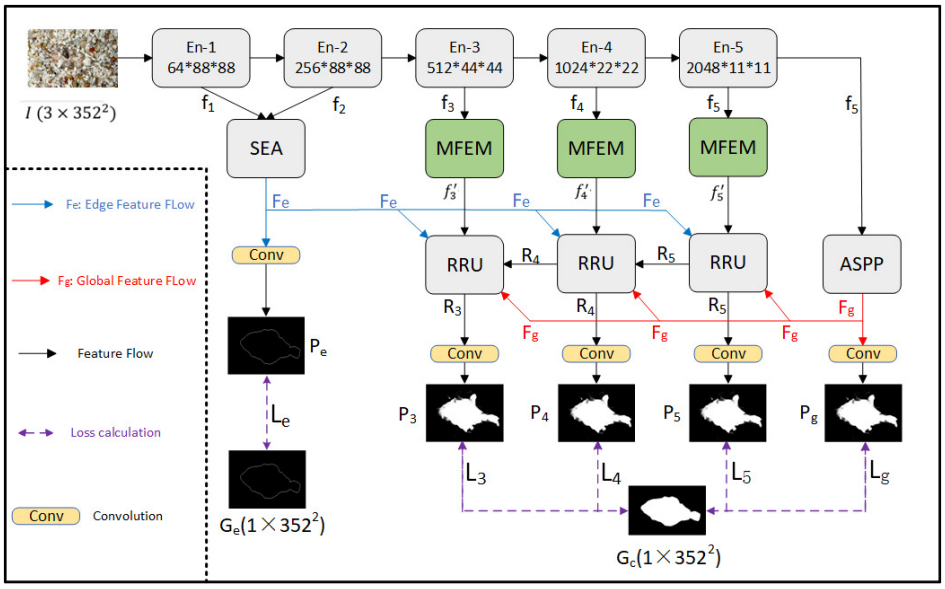

Fig. 2 The overall pipeline of the proposed MSFENet.

pyramid pool module (PPM) [35] to produce multi-layer information, and then integrated these multi-scale information to obtain global contextual feature information. Influenced by the idea of PPM module, Liu et al. (PoolNet) [17] designed the feature aggregation module (FAM). Firstly, FAM transforms input features into multiple information space by average pooling layers of different sizes. These multi-scale information are then integrated through element-wise addition to capture rich contextual information. The model then attempts to use multiple dilated convolutions to extract multi-scale features. Li et al. (MGA) [14] used the atrous spatial pyramid pooling module (ASPP) [1] to capture extensive connections in feature maps, acquiring multi-scale context information. Fan et al. (SINetV2) [8] introduced the receptive field block module (RFB) [31] to simulate the receptive field structure of the system of human visual perception, thereby improving the performance.

However, the PPM module and the FAM module both acquire multi-scale feature information through down-sampling layers of different sizes, which may lead to the loss of tiny structure information and small target feature information, and these information can not be recovered. ASPP and RFB use dilated convolutions with different dilation rates to expand the receptive field through different branches. However, large dilation rates may cause small target features to be mixed with many regional background features, making small target localisation inaccurate.

3 Methodology

3.1 Overview

Based on ERRNet [12], this paper proposed a multi-scale feature enhanced network (MSFENet), for camouflaged object detection, which mainly includes: The backbone networks ResNet-50 [10], Selective Edge Aggregation module (SEA) [12], our designed Multi-scale Feature Enhancement Module (MFEM), Atrous Spatial Pyramid Pooling module (ASPP) [1], and Reversible Re-calibration Unit (RRU)[12]. Fig. 2 shows the overall pipeline of MSFENet. The backbone network ResNet-50 extracts five feature hierarchy *En-i*, $i \in \{1, 2, 3, 4, 5\}$ from the input image from low to high. The SEA module extracts the rich edge feature information (F_e) contained in the low-level features (*En-1*, *En-2*). The MFEM module mines the multi-scale feature information in the high-level features (*En-3*, *En-4*, *En-5*) from coarse to fine. The extracted multi-scale information contains more microstructural information and small target feature information with less regional background information. The ASPP module extracts the global feature information (F_g) contained in the highest level feature (*En-5*). The RRU module is able to perform a coarse-to-fine reverse recalivity of the prediction against the potential camouflaged region and its complementary region. Given an input image $I \in R^{H \times W \times 3}$, the whole process of feature flow can be expressed as follows.

$$f_{i,i\in\{1,2,3,4,5\}} = ResNetf50(I \in R^{H \times W \times 3}) \tag{1}$$
$$F_e = SEA(f_1, f_2) \tag{2}$$
$$f_3', f_4', f_5' = MFEM(f_3, f_4, f_5) \tag{3}$$
$$F_g = ASPP(f_5) \tag{4}$$
$$R_5 = RRU(F_e, F_g, f_5') \tag{5}$$
$$R_{i,i\in\{3,4\}} = RRU(F_e, F_g, f_5', R_{i+1}) \tag{6}$$
$$P_{i,i\in\{3,4,5\}} = Conv(R_i) \tag{7}$$
$$P_e = Conv(F_e), P_g = Conv(F_g) \tag{8}$$

$f_{i,i\in\{1,2,3,4,5\}}$ represents the five layers of features extracted by ResNet-50 from the input image $I \in R^{H \times W \times 3}$, and F_e represents the edge feature information extracted by the SEA module. f_3', f_4', f_5' represent the multi-scale features of f_3, f_4 and f_5 enhanced by MFEM module, F_g represents the global feature information extracted by ASPP module, $R_{i,i\in\{3,4,5\}}$ represents the feature information calibrated by RRU module, $Conv$ represents the 1×1 convolutional layer. $P_{i,i\in\{3,4,5\}}$, P_e, P_g represent the generated prediction map information.

Finally, we use the true edge map G_e with the predicted edge map P_e to calculate the edge loss L_e, and the true mask map G_c with the predicted mask maps $P_{i,i\in\{3,4,5\}}$ and P_g to calculate the loss $L_{i,i\in\{3,4,5\}}$, L_g, respectively.

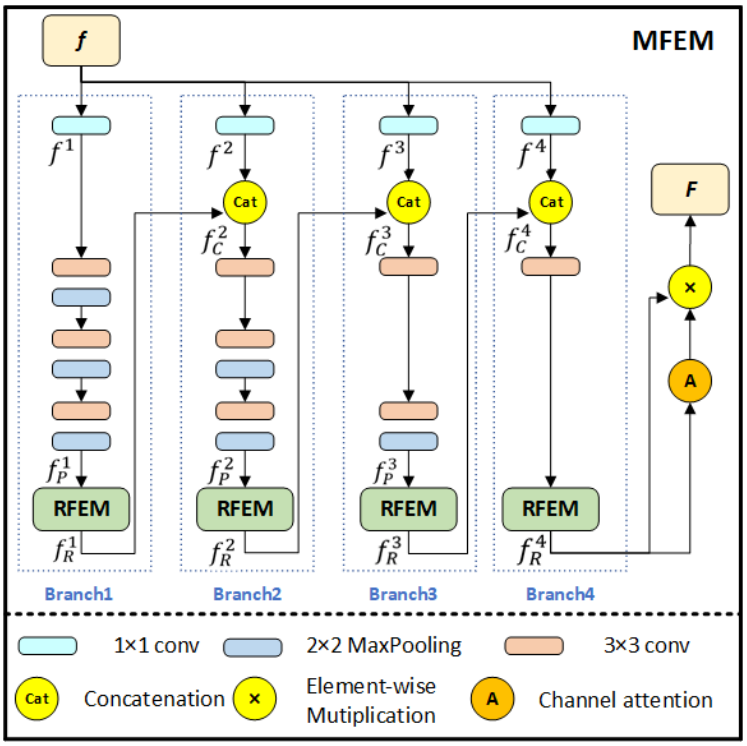

Fig. 3 The architecture of our MFEM module, where f is input and F is output.

3.2 Multi-scale feature enhancement module

The structure of the proposed MFEM is shown in Fig. 3. MFEM employs several branches to sequentially process an input feature f, extracting and enhancing multi-scale features progressively from coarse to fine. Specifically, MFEM includes four branches. MFEM is comprised of four branches, each housing a 1×1 convolutional layer tasked with channel size compression, thus reducing the amount of computation and improving the computational efficiency. The compressed features can be represented as $\{f^i : i = 1, 2, 3, 4\}$, where i denotes the index of branch.

Following the acquisition of f^1, we successively use three groups of 3×3 convolutional layers and 2×2 maximum pooling layers to expand the receptive field. Then, the output features are fed into the receptive field expansion module (RFEM) to further expand the receptive field and obtain richer multi-scale context information, while avoiding the extracted small target feature information is mixed with too much background information (the details of the RFEM module will be introduced in Section. 3.3). The procedure can be expressed as follows. C_3 represents a 3×3 convolutional layer, P_m represents a 2×2 Max pooling layer, and R represents the RFEM module.

$$f_P^1 = P_m(C_3(P_m(C_3(P_m(C_3(f^1)))))) \tag{9}$$

$$f_R^1 = R(f_P^1) \tag{10}$$

Although multiple Max pooling layers and RFEM modules expand the receptive field, they lose detailed microstructural information, making the feature information of large targets incomplete. In addition, multiple downsampling also leads to the loss of feature information of small targets. To solve these problems, we combine f_R^1 and f^2 through channel connections to recover the lost microstructural information of large targets and the feature information of small targets, while maintaining the abundant context information. Subsequently, a 3×3 convolutional layer is used to adjust the output of the channels number of the fused features to 64. We then reduce the number of downsampling layers and successively use two 3×3 convolutional layers plus 2×2 maximum pooling layers to model larger contexts and to characterize features at different scales. The RFEM module is applied in a similar way in the second branch and the process can be formulated follows, where $up(f_R^1, f^2)$ represents the bilinear interpolation procedure that upsamples f_R^1 to the same size as f^2 and Cat denotes feature fusion. It is worth noting that because the second branch uses fewer downsampling layers, f_R^2 has richer semantic feature information than f_R^1, i.e. f_R^2 retains finer microstructural information and more small target feature information.

$$f_C^2 = Cat(up(f_R^1, f^2), f^2) \tag{11}$$

$$f_P^2 = P_m(C_3(P_m(C_3(C_3(f_C^2))))) \tag{12}$$

$$f_R^2 = R(f_P^2) \tag{13}$$

Likewise, within the third and fourth branches, we progressively decrease the count of subsampling layers and continuously aggregate high-resolution input features. Upon acquiring the result from the fourth branch, we apply the channel attention mechanism to highlight those channels that are highly sensitive to the camouflaged object. The final output of MFEM F is formulated as follows, where f_R^4 denotes the output result from the fourth branch, P_a is the global average pooling layer, C_1 is the 1×1 convolutional layer, σ is the sigmoid function, and $\bigotimes$ is the element-wise multiplication operation.

$$F = \sigma(C_1(C_1(P_a(f_R^4)))) \bigotimes f_R^4 \tag{14}$$

3.3 Receptive field expansion module

In order to further enrich the context information obtained by each branch of MFEM and make up for the reduction of receptive field caused by the reduced sub-sampling layers of the second, third and fourth branches, we design the Receptive Field Expansion Module (RFEM). RFEM utilizes the characteristics of the human visual system (that is, a set of different sizes of receptive fields [31] plays a key role in object detection), and jointly uses dilated convolution with small dilation rate and asymmetric convolution to expand the receptive field at the same time, so as to obtain richer multi-scale context information and extract small object feature information with less regional background information. The structure of the RFEM is shown in Fig. 4, and

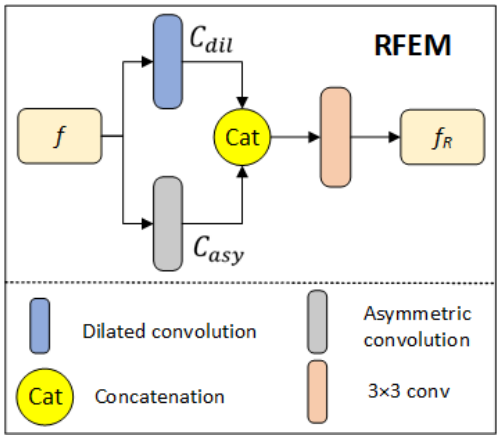

Fig. 4 The architecture of our RFEM module, where f is input and f_R is output.

the whole process can be expressed as follows.

$$f_R = C_3(Cat(C_{dil}(f), C_{asy}(f))) \tag{15}$$

Where f_R represents output feature of the RFEM, C_3 represents 3×3 convolutional layer, Cat represents connection operation, C_{dil} and C_{asy} are dilated convolution and asymmetric convolution, respectively. We set the dilation rate of the dilated convolution to 2, because a smaller dilation rate can extract small object features with less surrounding background information. Asymmetric convolutions include a standard 3×3 convolution, a vertical 3×1 convolution and a horizontal 1×3 convolution, which share the same sliding window. The process of asymmetric convolution can be formulated as follows:

$$C_{asy}(x) = (C_{3\times3} \bigodot x) \bigoplus (C_{3\times1} \bigodot x) \bigoplus (C_{1\times3} \bigodot x) \tag{16}$$

Where x represents the input feature, $\bigodot$ represents the two-dimensional convolution operator, *bigoplus* represents element-wise summation, $C_{3\times3}$, $C_{3\times1}$, $C_{1\times3}$ represent the standard 3×3 convolution, the vertical 3×1 convolution and the horizontal 1×3 convolution, respectively.

3.4 Loss function

For training multiple branches of the network, the joint supervision strategy of the ERRnet [12] is utilized in this paper to acquire knowledge from the source domain of multiple modalities (i.e. real mask map G_c and real edge map G_e). The total loss function L_{all} is composed of two parts: the loss L_e of the edge supervision of the camouflaged object from the microscopic perspective and the loss $L_{i,i\in\{3,4,5,g\}}$ of the

mask supervision of the camouflaged object.

$$L_{all} = L_e + \sum_{i \in \{3,4,5,g\}} L_i \tag{17}$$

As there is a large imbalance between positive and negative samples, which may lead to a large deviation of the loss at the macro level in the learning process, only weighted Binary Cross-Entropy (BCE) is used, i.e. $L_e = L^W_{BCE}$. $L_{i,i\in\{3,4,5,g\}}$ supervises the camouflaged object mask from micro and macro perspectives, consisting of weighted Binary Cross-Entropy (BCE) and weighted Intersection-of-Union (IoU), i.e. $L_{i,i\in\{3,4,5,g\}} = L^W_{i_{BCE}} + L^W_{i_{IoU}}$.

4 Experiments

4.1 Dataset

4.1.1 Total dataset

We evaluate the proposed MSFENet from multiple perspectives on three publicly available datasets: COD10K [5], CAMO [13], and CHAMELEON [28]. The COD10K dataset consists of 3040 training images and 2026 testing images. The CAMO dataset consists of 1000 training images and 250 testing images. The CHAMOELEON dataset consists of 76 testing images. The training set (Train) used in the experiments contains the training images of COD10K and CAMO, and the test set (ALL-Test) contains the testing images of COD10K, CAMO and CHAMOELEON.

4.1.2 Multiscale object test subset

In order to better evaluate the detection performance of the network for multi-scale camouflaged objects, the ALL-Test is divided into a test subset of multi-scale target (MT-Test) (containing multi-scale target images) and a test subset of non-multi-scale target image (NoMT-Test) (the remaining part of the ALL-Test after removing multi-scale target images), depending on whether the test set contains multi-scale targets (i.e. large targets, small targets, multiple targets). The summary of the two test subsets is shown in TABLE 1.

Table 1 Summary of different test sets on three COD datasets.

	COD10K [5]	CAMO [13]	CHAMELEON [28]	Total
ALL-Test	2026	250	76	2352
MT-Test	636	106	48	770
NoMT-Test	1390	144	48	1582

4.2 Implementation details and Evaluation metrics

4.2.1 Implementation details

The proposed MSFENet is implemented in the Pytorch framework by using the standard Adam algorithm with RTX 4090 GPU acceleration. Before training, the size of the input image (the corresponding real mask map Gc and the real edge map Ge) is uniformly adjusted to 352×352. For data augmentation, only multi-scale input images are used. The parameters of the backbone Resnet-50 are initialized using the weights of the ImageNet pre-trained model, and the parameters of other layers are randomly initialized. In the training phase, the epoch is set to 100 with a batch size of 36, and the learning rate starts at 1e-4, divided by 10 every 50 epochs. The inference process is carried out on the total test set (ALL-Test) and the multi-scale target subset (MT-Test). All images are resized to 352×352 and fed to the trained model to obtain the final prediction results without any post-processing. The inference speed reaches up to 75.3 FPS.

4.2.2 Evaluation metrics

We choose the P_3 of MSFENet as the final result, and use four widely used criteria to quantitatively evaluate the performance of the model.

Structure-measure [3]: S_α calculates the structural similarity between P_3 and G_c by considering both the region part and the target part, where S_0 and S_r are the structural similarity of target sense and region sense, respectively. The similarity between S_0 and S_r is balanced by α=0.5.

$$S_\alpha = \alpha \cdot S_0(P_3, G_c) + (1-\alpha) \cdot S_r(P_3, G_c) \tag{18}$$

Mean Enhanced-measure [4]: E_ϕ captures image-level statistics and local pixel matching information via an Enhanced alignment matrix ϕ_{FM}.

$$E_\phi = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} \phi_{FM}(i,j) \tag{19}$$

Weighted F-measure [18]: F_β^W use a weighting function for the precision and recall errors, where $Precision^W$ and $Recall^W$ denote weighted precision and recall with β=0.3.

$$F_\beta^W = \frac{(1+\beta^2) \times Precision^W \times Recall^W}{\beta^2 \times Precision^W \times Recall^W} \tag{20}$$

Mean Absolute Error [22]: M calculates the mean absolute error between the predicted map P_3 and the true map G_c for all image pixels, where W and H denote the

width and height of the real map G_c.

$$M = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} |P_3(i,j) - G_c(i,j)| \tag{21}$$

Table 2 Average S_α, E_ϕ, F_β^W and M scores of our model and mainstream models on the three COD datasets in All-Test.

methods	**year**	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$
FPN [15]	2017	.725	.717	.495	.094
UNet++ [37]	2019	.639	.696	.414	.110
CPD [31]	2019	.775	.788	.588	.075
EGNet [36]	2019	.772	.806	.598	.070
SINet [5]	2020	.797	.823	.632	.065
PraNet [6]	2020	.806	.857	.685	.061
UCNet [29]	2021	.798	.869	.697	.057
TINet [38]	2021	.816	.870	.702	.056
CubeNet [39]	2022	.815	.877	.704	.054
ERRNet [12]	2022	.806	.870	.698	.056
Ours	-	**.819**	**.881**	**.717**	**.053**

4.3 Overall performance comparison

The experimental results of our MSFENet are compared with 10 mainstream methods: (1) FPN [15], (2) UNet++ [37], (3) CPD [31], (4) EGNet [36], (5) SINet [5], (6) UCNet [29], (7) PraNet [6], (8) TINet [38], (9) CubaNet [39] and (10) ERRNet [12]. Table 2 shows the average prediction results. The detailed results of the different methods on three datasets are shown in Table 3. The overall performance of MSFENet is better than that of these existing methods. Compared with the method of ERRNet, our method has larger S_α, E_ϕ, F_β^W scores and smaller M scores, which indicates that our method can identify the camouflaged object more accurately. Specifically, the average S_α, E_ϕ and F_β^W scores of our method on the three COD datasets are relative improved by 1.7%, 1.3%, 2.8%, absolutely improved by 1.3%, 1.1%, 1.9%, respectively (Table 2).

4.4 Results of the multi-scale target test subset

In order to thoroughly evaluate our method for multi-scale camouflaged object detection, we conduct a series of tests on the multi-scale target test subset and compare our proposed MSFENet with the baseline method of ERRNet. Firstly, according to the parameter weights given by literature [12], we reproduce the ERRNet method and evaluate its performance on multi-scale target Test subset (MT-Test), non-multi-scale

Table 3 Performance comparison of our method with mainstream models.

methods	year	COD10K [5]				CAMO [13]				CHAMELEON [28]			
		$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$
FPN [15]	2017	.697	.691	.411	.075	.684	.677	.483	.131	.794	.783	.590	.075
UNet+ [37]	2019	.623	.672	.350	.086	.599	.653	.392	.149	.695	.762	.501	.094
CPD [31]	2019	.747	.770	.508	.059	.726	.729	.550	.115	.853	.866	.706	.052
EGNet [36]	2019	.737	.779	.509	.056	.732	.768	.583	.104	.848	.870	.702	.050
SINet [5]	2020	.771	.806	.551	.051	.751	.771	.606	.100	.869	.891	.740	.044
PraNet [6]	2020	.789	.839	.629	.045	.769	.833	.663	.094	.860	.898	.763	.044
UCNet [29]	2021	.776	.867	.633	.042	.739	.811	.640	.094	.880	.929	.817	.036
TINet [38]	2021	.793	.848	.645	.043	.781	.847	.678	.087	.874	.916	.783	.038
CubeNet [39]	2022	.795	.864	.644	.041	.778	.838	.682	.085	.873	.928	.787	.037
ERRNet [12]	2022	.780	.867	.629	.044	.761	.817	.660	.088	.877	.927	.805	.036
Ours	-	**.795**	**.871**	**.649**	**.041**	**.784**	**.843**	**.695**	**.084**	**.878**	**.93**	**.806**	**.034**

Table 4 Average S_α, E_ϕ, F_β^W and M scores of the ERRNet on All-Test, MT-Test and NoMT-Test.

ERRNet [12]	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$
All-Test	.806	.870	.698	.056
MT-test	.784	.836	.667	.073
NoMT-test	.826	.900	.715	.045

Table 5 Average S_α, E_ϕ, F_β^W and M scores of our method and the ERRNet on MT-Test.

methods	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$
ERRNet [12]	.784	.836	.667	.073
Ours	**.797**	**.842**	**.692**	**.068**

target test subset (NoMT-Test) and total test set (ALL-Test). The average results are shown in Table 4.

Compared with the results on the ALL-Test, the average S_α, E_ϕ and F_β^W scores of the ERRNet on the MT-Test are relatively reduced by 2.6%, 3.8%, 4.5%, absolutely reduced by 2.2%, 3.4%, 3.1%, respectively. The average M value is relatively improved by 33.7% and absolutely improved by 1.7%. Since the MT-Test only contains images of multi-scale camouflaged objects, this result shows that the ERRNet performs poorly in detecting multi-scale camouflaged objects. Compared with the results on the ALL-Test, the average S_α, E_ϕ and F_β^W scores of the ERRNet on the NoMT-Test are relatively improved by 2.6%, 3.5%, 2.5%, and absolutely improved by 2.0%,

3%, 1.7%, respectively. The average M value is relatively reduced by 17.8% and absolutely reduced by 1.1%. As the NoMT-Test does not contain multi-scale targets, this result shows from another perspective that the ERRNet performs poorly in detecting multi-scale camouflaged objects.

Similarly, the average results of our method on the MT-Test are shown in Table 5. Compared with the ERRNet, the average S_α, E_ϕ and F_β^W scores of our method on the MT-Test are relatively improved by 1.6%, 0.7%, 3.9%, and absolutely improved by 1.2%, 0.6%, 2.5%, respectively. The average M value is relatively reduced by 7.6% and absolutely reduced by 0.5%. These results indicate that our method outperforms the baseline of ERRNet for the detection of multi-scale camouflaged objects.

Fig. 5 presents the prediction images of our method and ERRNet on six input images, which representing different detection tasks of multi-scale camouflaged object. For large target (i.e. Pic 1 and Pic 2) and medium-sized target (i.e. Pic 3), our method can achieve a more comprehensive detection of the primary elements, compared with the baseline ERRNet. Our method can locate the position of small target more accurately than the ERRNet (i.e. Pic 4 and 5). In addition, our method can detect multiple targets more completely than the ERRNet (i.e. Pic 6).

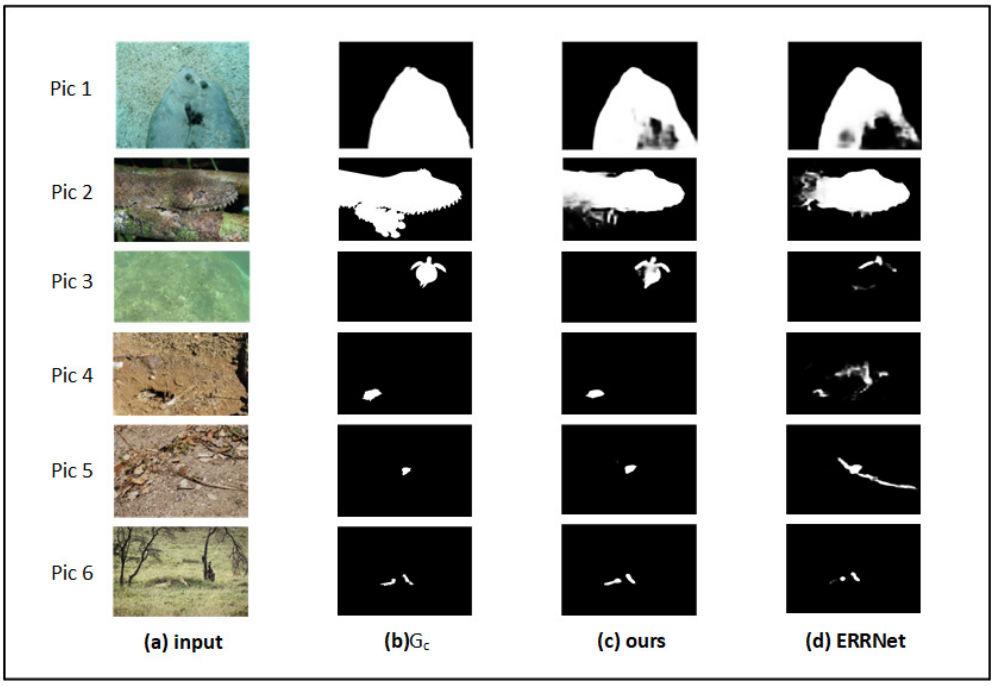

Fig. 5 Visual comparison of the predicted maps of our method and the ERRNet, where (a) *input* denotes the input images , (b) G_c denotes the true mask map, and (c) *ours* denotes our method.

4.5 Ablation studies

4.5.1 The effectiveness of MFEM

Table 6 Quantitative results for the ablation studies on the effectiveness of MFEM.

methods	COD10K [5]				CAMO [13]				CHAMELEON [28]			
	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$
model_base	.779	.852	.616	.045	.767	.817	.657	.092	.854	.913	.761	.041
model_RFB	.786	.864	.634	.043	.775	.831	.676	.086	.870	.917	.786	.036
model_ASPP	.783	.861	.632	.044	.776	.833	.674	.087	.868	.924	.781	.038
model_PPM	.785	.862	.632	.044	.772	.827	.668	.088	.863	.919	.773	.040
model_FAM	.789	.868	.639	.042	.781	.838	.681	.086	.871	.926	.787	.035
model_MFEM	**.795**	**.871**	**.649**	**.041**	**.784**	**.843**	**.695**	**.084**	**.878**	**.93**	**.806**	**.034**

To verify whether the MFEM module can improve the performance, the MFEM is replaced by a 3×3 convolutional layer, called model_base. In addition, the MFEM is replaced by RFB [31], ASPP [1], PPM [35] and FAM [17], denote as model_RFB, model_ASPP, model_PPM and model_FAM, respectively. The experiment are carried out on the All-Test and the results are shown in Table 6. It is obvious that model_base is the worst, the other four comparison models (i.e. model_RFB, model_ASPP, model_PPM and model_FAM) are in the middle and our model_MFEM is the best. These results show that the MFEM module can better extract the multi-scale feature information from the high-level features *En-3*, *En-4*, *En-5*, thus improving the COD performance.

4.5.2 The effectiveness of RFEM

Table 7 Quantitative results for the ablation studies on the effectiveness of RFEM.

methods		COD10K [5]				CAMO [13]				CHAMELEON [28]			
		$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^W \uparrow$	$M \downarrow$
model_DC	model_DCMT	.785	.858	.628	.045	.775	.831	.681	.085	.867	.915	.787	.036
	model_DCS	.791	.866	.64	.043	.779	.838	.689	.084	.872	.921	.794	.035
model_RFEM		**.795**	**.871**	**.649**	**.041**	**.784**	**.843**	**.695**	**.084**	**.878**	**.93**	**.806**	**.034**

To verify the effectiveness of the RFEM module, for each branch of the MFEM, we replace the RFEM module by a single dilated convolution, called model_DC. Diluted convolution with a high dilation rate could cause the extracted small target feature information to carry too much regional background information, resulting in performance degradation. In order to explore this effect, we divide model_DC into

model_DCS (i.e. the dilation rate of dilated convolution in each branch is set to 2) and model_DCMT (i.e. the dilation rates of dilated convolution in four branches are set to 1, 3, 5 and 7, respectively). The experiment results on the ALL-Test show that model_DCMT is the worst, model_DCS is in the middle, and our model_RFEM is the best (Table 7). These result reveal that the RFEM module can better expand the receptive field and extract more contextual information than a single dilated convolution, while avoiding the performance degradation caused by the large dilation rate.

4.6 Discussions

To address the problem of multi-scale camouflaged object detection, we proposed the multi-scale feature enhanced network (MSFENet). On the ALL-Test, our method outperforms 10 mainstream methods in overall performance (including the baseline method of ERRNet). On the multi-scale target test subset (MT-test), the multi-scale camouflaged object detection performance of our method is also better than the baseline method of ERRNet. This is attributed to the multi-scale enhancement module (MFEM) proposed in this paper. The multi-scale feature information extracted by the MFEM from high-level features (*En-3*, *En-4*, *En-5*) contains more microstructural information and more small target feature information with less regional background information, so that the model can better deal with scale changes. In addition, the inference speed of our method reaches up to 75.3 frames per second (FPS), which is much larger than the real-time inference speed of 30 FPS [12].

Our method initially relies on the ASPP module to extract global feature cues for preliminary predictions, identifying potential areas where the camouflaged object might exist. Then our method refines the coarse prediction. However, when encountering more complex scenes (such as cluttered background, large area occlusion, discontinuous camouflaged object shape), the camouflaged object area may be located inaccurately, resulting in poor detection performance although the preliminary prediction is optimized by the network. In the future, we will further optimize our method to solve this problem.

5 Conclusion

In this paper, we proposed a novel deep learning-based method for camouflaged object detection. Specifically, based on the ERRNet, we designed a multi-scale feature extraction enhancement module to improve the ability of a single layer to refine multi-scale information in a multi-granularity way. Moreover, the module can retain more microstructural information and small target feature information, expand the receptive field and avoid the extracted small target feature information to mix with too much regional background information. Extensive experimental results demonstrate the effectiveness of our method.

Declarations

- Funding

This work was supported in part by Key Laboratories of Sensing and Application of Intelligent Optoelectronic System in Sichuan Provincial Universities (ZNGD2217)

- Ethics approval
  The content of our submission is not applicable for human and/ or animal studies.
- Availability of data and materials
  The data that support the findings of this study are openly available in https://github.com/taozh2017/FAPNet

References

1. Chen LC, Papandreou G, Kokkinos I, et al (2017) Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence 40(4):834–848

2. Chu HK, Hsu WH, Mitra NJ, et al (2010) Camouflage images. ACM Trans Graph 29(4):51–1

3. Fan DP, Cheng MM, Liu Y, et al (2017) Structure-measure: A new way to evaluate foreground maps. In: Proceedings of the IEEE international conference on computer vision, pp 4548–4557

4. Fan DP, Gong C, Cao Y, et al (2018) Enhanced-alignment measure for binary foreground map evaluation. arXiv preprint arXiv:180510421

5. Fan DP, Ji GP, Sun G, et al (2020) Camouflaged object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2777–2787

6. Fan DP, Ji GP, Zhou T, et al (2020) Pranet: Parallel reverse attention network for polyp segmentation. In: International conference on medical image computing and computer-assisted intervention, Springer, pp 263–273

7. Fan DP, Zhou T, Ji GP, et al (2020) Inf-net: Automatic covid-19 lung infection segmentation from ct images. IEEE transactions on medical imaging 39(8):2626–2637

8. Fan DP, Ji GP, Cheng MM, et al (2021) Concealed object detection. IEEE transactions on pattern analysis and machine intelligence 44(10):6024–6042

9. Hall JR, Matthews O, Volonakis TN, et al (2021) A platform for initial testing of multiple camouflage patterns. Defence Technology 17(6):1833–1839

10. He K, Zhang X, Ren S, et al (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778

11. Huang Y, Wu W, Gong Y, et al (2011) A new method of edge camouflage evaluation based on the gray polymerization histogram. Optical Technique, Papers 37(5):601–606

12. Ji GP, Zhu L, Zhuge M, et al (2022) Fast camouflaged object detection via edge-based reversible re-calibration network. Pattern Recognition 123:108414

13. Le TN, Nguyen TV, Nie Z, et al (2019) Anabranch network for camouflaged object segmentation. Computer vision and image understanding 184:45–56

14. Li H, Chen G, Li G, et al (2019) Motion guided attention for video salient object detection. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 7274–7283

15. Lin TY, Dollár P, Girshick R, et al (2017) Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2117–2125

16. Lin TY, Goyal P, Girshick R, et al (2017) Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision, pp 2980–2988

17. Liu JJ, Hou Q, Cheng MM, et al (2019) A simple pooling-based design for real-time salient object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3917–3926

18. Margolin R, Zelnik-Manor L, Tal A (2014) How to evaluate foreground maps? In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 248–255

19. Merilaita S, Scott-Samuel NE, Cuthill IC (2017) How camouflage works. Philosophical Transactions of the Royal Society B: Biological Sciences 372(1724):20160341

20. Pan Y, Chen Y, Fu Q, et al (2011) Study on the camouflaged target detection method based on 3d convexity. Modern Applied Science 5(4):152

21. Pang Y, Zhao X, Zhang L, et al (2020) Multi-scale interactive network for salient object detection. In: IEEE conference on computer vision and pattern recognition, pp 9413–9422

22. Perazzi F, Krähenbühl P, Pritch Y, et al (2012) Saliency filters: Contrast based filtering for salient region detection. In: 2012 IEEE conference on computer vision and pattern recognition, IEEE, pp 733–740

23. Ren S, He K, Girshick R, et al (2015) Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing

systems 28

24. Sengottuvelan P, Wahi A, Shanmugam A (2008) Performance of decamouflaging through exploratory image analysis. In: 2008 First International Conference on Emerging Trends in Engineering and Technology, IEEE, pp 6–10

25. Shi CJ, Ren BJ, Wang ZW, et al (2022) A survey of camouflaged object detection with deep learning. Journal of Frontiers of Computer Science & Technology 16(12)

26. Sun Y, Chen G, Zhou T, et al (2021) Context-aware cross-level fusion network for camouflaged object detection. arXiv preprint arXiv:210512555

27. Wang T, Borji A, Zhang L, et al (2017) A stagewise refinement model for detecting salient objects in images. In: Proceedings of the IEEE international conference on computer vision, pp 4019–4028

28. Wei J, Wang S, Huang Q (2020) F^3net: fusion, feedback and focus for salient object detection. In: Proceedings of the AAAI conference on artificial intelligence, pp 12321–12328

29. Wu YH, Liu Y, Zhang L, et al (2021) Regularized densely-connected pyramid network for salient instance segmentation. IEEE Transactions on Image Processing 30:3897–3907

30. Wu YH, Liu Y, Zhang L, et al (2022) Edn: Salient object detection via extremely-downsampled network. IEEE Transactions on Image Processing 31:3125–3136

31. Wu Z, Su L, Huang Q (2019) Cascaded partial decoder for fast and accurate salient object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3907–3916

32. Xu WD, LU XL, Bing C, et al (2002) An evaluation model of camouflage equipment effect based on texture analysis. Journal of Armaments 23(3):329–331

33. Xue F, Yong C, Xu S, et al (2016) Camouflage performance analysis and evaluation framework based on features fusion. Multimedia Tools and Applications 75:4065–4082

34. Yi D, Su J, Chen WH (2019) Locust recognition and detection via aggregate channel features. Poster Papers p 112

35. Zhao H, Shi J, Qi X, et al (2017) Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 6230–6239

36. Zhao JX, Liu JJ, Fan DP, et al (2019) Egnet: Edge guidance network for salient object detection. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 8779–8788

37. Zhou Z, Siddiquee MMR, Tajbakhsh N, et al (2019) Unet++: Redesigning skip connections to exploit multiscale features in image segmentation. IEEE transactions on medical imaging 39(6):1856–1867

38. Zhu J, Zhang X, Zhang S, et al (2021) Inferring camouflaged objects by texture-aware interactive guidance network. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp 3599–3607

39. Zhuge M, Lu X, Guo Y, et al (2022) Cubenet: X-shape connection for camouflaged object detection. Pattern Recognition 127:108644