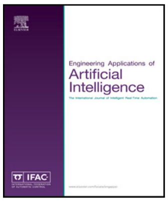
Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Research paper

Camouflaged Object Detection with boundary localization in complex backgrounds

Guangjian Zhang [a,1], Zhengming Yang [a,1], Yong Wang [a,*], Yuliang Chen [b], Duoqian Miao [c]

[a] School of Artificial Intelligence, Chongqing University of Technology, Chongqing, 401120, China
[b] Goldman Sachs, Orlando, FL, 32814, USA
[c] School of Computer Science and Technology, Tongji University, Shanghai, 200092, China

ARTICLE INFO

Keywords:
Camouflaged Object Detection
Complex backgrounds
Edge cues
Pest detection

ABSTRACT

The primary challenge of Camouflaged Object Detection (COD) lies in the high similarity between the target and the complex background, making it difficult for the human eye to distinguish them. Based on the phenomenon that human attention shifts between the target and the background when observing objects, we propose a network model named MENet. This model adopts a three-stage decoupled architecture of "localization-interaction-fusion." In the localization stage, we utilize an attention mechanism-based backbone network (Pyramid Vision Transformer V2, abbreviated as PVT-V2) to generate multi-level features, which can initially locate the target area. In the interaction stage, we design a Contour-Aware Edge Module (CAEM) and an Area Decoder (AD) to capture the target edges and background information, respectively, thereby achieving precise localization of the target boundary and reducing interference from background noise. Furthermore, we developed a Boundary Guidance Module (BGM) that effectively injects boundary cues and relevant background information separately into the multi-level features, enhancing the model's ability to detect target edges in complex backgrounds. In the fusion stage, we design two Feature Fusion Modules (FFM and KFFM) to effectively merge multi-level features with precise boundaries and de-noised features, thereby enhancing the prediction performance of camouflaged objects. Extensive experiments on three challenging benchmark datasets demonstrate that our MENet outperforms many existing state-of-the-art methods. Our method leverages artificial intelligence (AI) techniques to improve the accuracy of camouflaged object and pest detection in complex visual environments. Our code is publicly available at: https://github.com/yang19950966666/MENet.

1. Introduction

To avoid predators, many animals use various materials, lighting conditions, or adapt their colors and textures (Wang et al., 2024a) to blend into their surroundings. Camouflaged Object Detection (COD) (Guo and Huang, 2025) aims to identify such deceptive targets that seamlessly integrate with their backgrounds. Due to the intrinsic low contrast and visual ambiguity, COD presents a unique and challenging task that continues to attract increasing research interest. COD has wide-ranging applications with significant potential value, including species discovery, endangered animal monitoring, image fusion (Wang et al., 2024c), retinal segmentation, post-disaster search and rescue, and pest detection in smart agriculture. Early approaches relied on hand-crafted features (e.g., texture, color, and 3D shape) to detect camouflaged objects, but these methods often failed in complex scenes with subtle differences between foreground and background.

Recent advances in deep learning have led to significant improvements in COD performance. Existing methods can be broadly grouped into several categories: (1) models designed with task-specific modules to enhance key features (e.g., UGTR (Yang et al., 2021), C^2FNet (Chen et al., 2022)); (2) predator-inspired models that mimic hunting behaviors (e.g., SINet, PFNet (Mei et al., 2021)); and (3) perception-driven models that simulate human visual mechanisms, such as edge-focused (BGNet (Sun et al., 2022)), progressive perception (ZoomNet (Pang et al., 2022)), and attention-guided frameworks (SegMaR (Jia et al., 2022), SARNet (Xing et al., 2023)).

Despite their success, these methods still face two major limitations. First, they often fail to accurately preserve object boundaries, especially

[*] Corresponding author.
E-mail addresses: zgj@cqut.edu.cn (G. Zhang), 19950965332@163.com (Z. Yang), ywang@cqut.edu.cn (Y. Wang), ychenx2@gmail.com (Y. Chen), dqmiao@tongji.edu.cn (D. Miao).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.engappai.2025.112047
Received 25 April 2025; Received in revised form 13 July 2025; Accepted 15 August 2025
Available online 26 August 2025

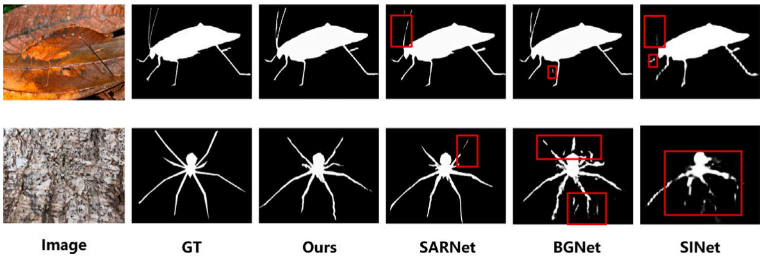


Fig. 1. Our model achieves excellent performance based on a precise edge strategy. Compared with previously proposed state-of-the-art (SOTA) methods (such as SARNet, BGNet, and SINet), our method adjusts the feature fusion strategy to better utilize boundary information, which can not only effectively distinguish object structures and details but also eliminate edge blur and misjudgment problems caused by camouflaged targets that are difficult to distinguish from the surrounding environment. The red box area is the edge blur and misjudgment problem.

when the boundaries are highly similar to the background (e.g., spider legs or insect antennae in Fig. 1), leading to blurred and incomplete segmentation. Second, they are vulnerable to background noise, which can cause misclassification of background regions as foreground, reducing prediction precision and completeness.

To address these challenges, we propose a novel COD framework that explicitly enhances edge localization and background suppression. Specifically, we introduce a Contour-Aware Edge Module (CAEM) and an Area Decoder (AD) to extract refined boundary and contextual background features. These are further integrated using a Boundary Guidance Module (BGM), which injects edge cues into the representation learning process, enhances scene-wise object separation, and mitigates background interference.

Our method bridges the gap between boundary accuracy and contextual robustness, enabling more precise camouflaged object detection in complex environments. Extensive experiments on benchmark datasets demonstrate the superiority of our approach in preserving object integrity and suppressing false positives, highlighting its practical value and methodological novelty.

In summary, our main contributions are as follows:

- We have developed a model, MENet, specifically for COD, which integrates positioning, interaction, integration, and other aspects. By fusing multi-level features and precise boundary features, MENet effectively suppresses background noise and significantly improves the model’s performance in COD tasks.
- We designed a Contour-Aware Edge Module (CAEM) to efficiently locate object boundaries.
- We developed a Boundary Guidance Module (BGM) to better utilize boundary semantics and suppress background noise.
- Experimental results show that our proposed model MENet outperforms most SOTA models in multiple indicators on multiple public datasets.

2. Related work

2.1. Camouflaged object detection

In contrast to the Salient Object Detection (SOD) task, Camouflaged Object Detection (COD) focuses on detecting objects that are not visually salient, mainly because they are often small, occluded, hidden, or self-camouflaging. Due to the different properties of these objects, the detection goals of COD and SOD are quite different. Camouflaged objects are particularly difficult to detect because the differences between them and their surroundings are very subtle, making COD much more complex and difficult than SOD. In order to deal with these challenges, we categorize recent methods into two major groups: The first is the early hand-made feature method, which mainly relies on features such as boundaries, color differences, and contrast. However, these methods usually suffer from limited robustness and poor performance in complex scenarios. In contrast, another type of method that uses deep learning models for camouflaged object detection is more effective in addressing these issues. Deep learning models have a large number of parameters and are good at segmenting camouflaged objects from complex backgrounds. In addition, many deep learning-based methods further improve COD performance by incorporating auxiliary information such as foveation points, boundaries, spatial locations, and image-level labels (Le et al., 2019a). Despite these advances, challenges related to incomplete and imprecise recognition of hidden objects in complex backgrounds remain unresolved. Therefore, we designed a COD network specifically for locating precise boundaries and removing background noise, aiming to complete the COD task with higher efficiency and accuracy.

2.2. Datasets

Camouflaged object detection (COD) presents a unique challenge due to the intrinsic low contrast between objects and their surroundings. In recent years, several public datasets such as CAMO, COD10K, and NC4K have been constructed to support research in this field. These datasets encompass diverse scenarios, including natural camouflage in wildlife (e.g., chameleons blending into vegetation) and artificial camouflage in military contexts, thereby posing significant recognition challenges for detection models. The objects in these datasets span a wide range of species—terrestrial, aquatic, aerial, and amphibious—covering nearly 70 categories of various sizes and shapes. This diversity introduces considerable variation in object appearance, background texture, and contextual information. As such, these datasets have not only played a critical role in benchmarking the performance of COD models but have also exposed key limitations in generalization and robustness observed in earlier methods. By addressing these challenges, our method aims to improve object localization and boundary precision in complex camouflage scenarios.

3. Methodology

3.1. Motivation

When perceiving camouflaged objects, humans often engage in a three-step visual process. Initially, they coarsely identify a region of interest based on global context (localization). Next, attention shifts dynamically between the object and its similar background within this region (interaction), during which edge cues naturally become critical for distinguishing foreground from background. Finally, humans integrate multiple perceptual signals—such as boundary, semantic, and contextual features to form a reliable decision about the object’s existence and shape (fusion). Motivated by this visual perception mechanism, our network is structured into three corresponding stages: localization, interaction, and fusion.

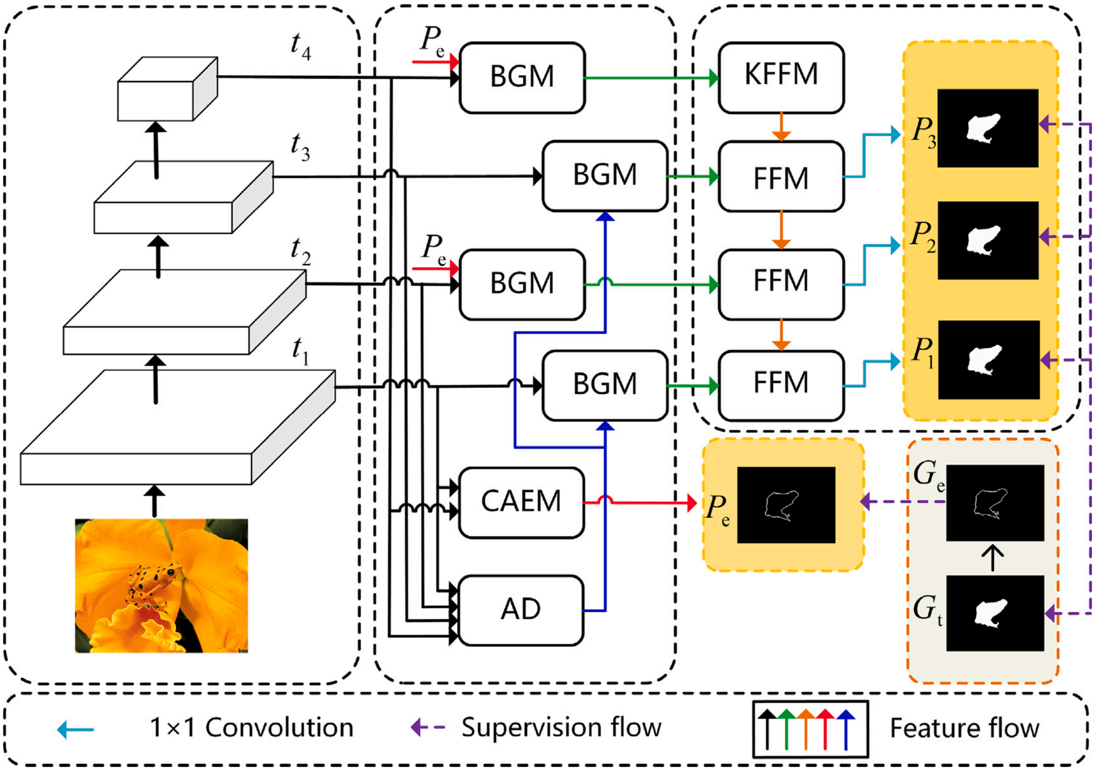


Fig. 2. The overall architecture of MENet consists of a localization module, an interaction module, and a Feature Fusion Module. The backbone PVT-V2 serves as a localization module to locate the rough target object and generate rough target regions of different scales. In the interaction module, CAEM and AD use the located rough target regions to generate precise boundaries and background noise. The two auxiliary cues, precise boundaries and background noise, are then integrated into the rough features through BGM, respectively, to enhance the distinction between foreground and background features. Then, the Feature Fusion Module (including FFM and KFFM) fuses the processed multi-level features together for the final prediction. CAEM generates precise boundaries P_e, where G_t and G_e are the true labels of the object region and boundary, respectively.

3.2. Network overview

In Fig. 2, we show the proposed three-stage model integration framework, namely the localization part, the interaction part, and the fusion part. Specifically, PVT-V2 (Wang et al., 2021) acts as the localization part for preliminary feature extraction. The interaction part consists of four BGM modules, one CAEM module, and one AD module to process background and boundary information. Finally, the feature fusion part consists of three FFM modules and one KFFM module for better fusion of different features.

3.3. Positioning components

Our backbone network uses a pyramid visual transformer (PVT-V2-B3) to extract multi-level features after preliminary localization. PVT-V2 uses a pyramid structure, a hierarchical design, and an efficient attention mechanism for feature extraction, which can generate high-resolution feature maps with low memory consumption. The input image $I \in R^{H\times W\times 3}$ is input to PVT-V2 and will go through steps such as block segmentation, pyramid structure processing, hybrid convolution and self-attention, multi-scale extraction, and Transformer encoder processing. The final output is reshaped into a multi-level feature map of size $\frac{H}{4} \times \frac{W}{4} \times C_i$, denoted as $t_i (i = 1, 2, 3, 4)$. The object boundary guidance module and Feature Fusion Module we proposed are both designed based on this attention-driven feature extraction mechanism.

3.4. Interaction components

When observing a hidden object, the human eye will instinctively distinguish the target object from the surrounding environment and often unconsciously pay attention to the unnatural transition between the two, that is, the boundary. Inspired by this mechanism, we design an interaction module consisting of four Boundary Guidance Module (BGM), a Contour-Aware Edge Module, and an Area Decoder (AD) module. The interaction module fuses cross-layer features of different granularities through BGM with edge features captured by CAEM, greatly reducing the impact of inaccurate boundary information on object boundary error and position misjudgment. In addition, the remaining features are merged with the background information predicted by AD (also through BGM) to mitigate interference from similar backgrounds. This design establishes a robust connection between the feature extraction and fusion stages. The input of the interaction module includes the preliminary positioning features $t_i (i = 1, 2, 3, 4)$ generated by the localization module and the edge information t_e or background features t_a obtained by edge extraction and area encoding of multi-level features. This module produces four outputs $t_i^a (i = 1, 2, 3, 4)$, which are generated by processing the preliminary positioning features $t_i (i = 1, 2, 3, 4)$ and the corresponding edge t_e or background features t_a through four corresponding BGM modules.

3.4.1. Context-aware edge module (CAEM)

In camouflaged object detection (COD), precise boundary modeling is of paramount importance due to the extremely low contrast between foreground objects and the surrounding background. Although many existing methods attempt to integrate low-level and high-level features to enhance object representation, such fusion often leads to the degradation or loss of fine-grained edge details. To mitigate this problem, we propose the Contour-Aware Edge Module (CAEM) (see Fig. 3), which leverages the Coordinate Attention (CA) mechanism to enhance spatial awareness and boundary localization. The CA mechanism (Hou et al., 2021) decomposes 2D spatial attention into two one-dimensional embeddings along the X-axis and Y-axis, allowing the network to capture

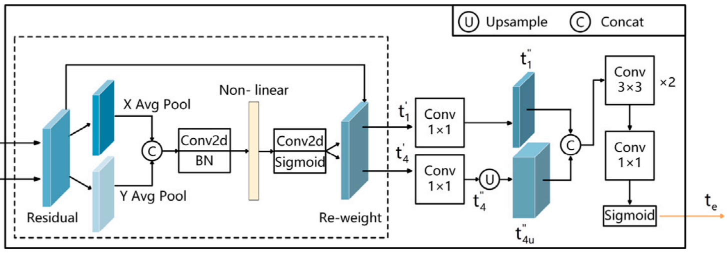


Fig. 3. The architecture of CAEM takes features t_4 containing rich global location information and features t_1 containing local detail information as input, aiming to explore more accurate boundary information.

long-range dependencies in both directions while embedding accurate positional information. Compared to conventional channel or spatial attention mechanisms, CA provides a superior trade-off between computational efficiency and localization precision, making it especially suitable for boundary-aware tasks such as COD. Following the insight from (Sun et al., 2022), we select the encoder features t_1 and t_4 – which exhibit the largest fine-grained gap – as inputs for edge prediction. First, t_1 and t_4 are enhanced by CA to obtain coordinate-aware attention maps t_1' and t_4':

$$t_1' = CA(t_1), \quad t_4' = CA(t_4) \tag{1}$$

Next, the attention-enhanced features are refined via 1 × 1 convolutions:

$$t_1'' = T_{conv1}(t_1'), \quad t_4'' = T_{conv1}(t_4') \tag{2}$$

Then, t_4'' is upsampled to match the resolution of t_1'', and the two are concatenated:

$$t_{cat} = [U(t_4''),\ t_1''] \tag{3}$$

The concatenated feature is further processed by two 3×3 convolutions followed by a 1×1 convolution, and the edge feature map t_e is obtained using a Sigmoid activation function:

$$t_e = \sigma(T_{conv1}(T_{conv3}(T_{conv3}(t_{cat})))) \tag{4}$$

Here, T_{conv1} and T_{conv3} represent 1 × 1 and 3 × 3 convolution operations, respectively; U denotes upsampling; $[\cdot]$ indicates channel-wise concatenation; and σ is the Sigmoid function. As shown in Fig. 7, CAEM effectively enhances edge localization by capturing semantic boundaries more precisely, especially for camouflaged targets with obscure contours. This confirms the effectiveness of CAEM as a lightweight and reliable edge extraction module.

3.4.2. Area encoder (AD)

Since camouflaged objects are visually difficult to distinguish from complex backgrounds, it is easy to mix interference information, such as clutter or shadows in the background, into the detected objects during the prediction process. In order to solve the problem of interference of background noise on camouflaged object detection, we propose a region encoder (AD) to generate background cues t_a, whose input is t_i $(i = 1, 2, 3, 4)$. To generate background cues t_a, we use a pyramid structure to gradually upsample high-level features and fuse them into adjacent low-level features, thereby achieving gradual fusion and denoising of multi-scale information. This design not only ensures that the module can effectively capture multi-scale features but also ensures that the spatial resolution of the output is consistent with the input. Specifically, taking the highest-level feature t_4 as an example, it is first processed by 1 × 1 convolution and then upsampled using bilinear interpolation to ensure that it matches the next-level feature map t_3 in the spatial dimension. Next, the upsampled high-level feature t_4 is fused with the adjacent feature t_3 of the next level through an addition operation. Similar operations are repeated until all multi-level features are fused together. Finally, the number of channels is adjusted through 1 × 1 convolution, and the Sigmoid function is applied to the output feature map, and the final background prediction map t_a is generated after inversion. The implementation process of AD can be expressed by the following formula:

$$t_4' = T_{conv1}(t_4) \tag{5}$$

$$t_{43} = U(t_4') + t_3 \tag{6}$$

$$t_3' = T_{conv1}(t_{43}) \tag{7}$$

$$t_{32} = U(t_3') + t_2 \tag{8}$$

$$t_2' = T_{conv1}(t_{32}) \tag{9}$$

$$t_{21} = U(t_2') + t_1 \tag{10}$$

$$t_a = 1 - \sigma(T_{conv1}(t_{21})) \tag{11}$$

3.4.3. Boundary Guidance Module (BGM)

In order to better integrate precise auxiliary cues into multi-scale features of different granularities, we design a Boundary Guidance Module (BGM) to process edge cues and background information. It is well known that the semantics implied by different feature channels are often different. In order to achieve optimal fusion, we adopt a dual-branch structure. The upper branch introduces a local channel attention mechanism to highlight the features that are more critical to the task according to the different contributions of different channels to the final task. The lower branch uses the spatial attention mechanism to adaptively explore representative areas and suppress irrelevant information. As shown in Fig. 4, given the auxiliary feature t_e and the coarse feature $t_i (i = 1, 2, 3, 4)$, t_e is downsampled and multiplied element-wise with t_i, and then added to the original multi-scale feature t_i. After that, the preliminary fusion feature t_i^e is obtained through 3 × 3 convolution, which is expressed as:

$$t_i^e = T_{conv3}((D(t_e) \otimes t_i) \oplus t_i), \tag{12}$$

where T_{conv3} represents 3 × 3 convolution, D represents downsampling, $\otimes$ represents element-by-element multiplication, and $\oplus$ represents element-wise addition. Then the low-level fusion feature t_i^e is further enhanced. In the upper branch, t_i^e is first processed by global average pooling (GAP), and then the feature obtained by one-dimensional convolution is activated with the Sigmoid activation function to obtain the corresponding attention. The attention mechanism is element-wise multiplied with t_i^e, and finally a 1 × 1 convolution is performed to generate the final output $t_{i_h}^a$ of the upper branch, as shown below:

$$t_{i_h}^a = (\sigma(T_{1D}(GAP(t_i^e))) \otimes t_i^e), \tag{13}$$

Where T_{1D} represents 1D convolution and σ represents the Sigmoid function. In the following branch, t_i^e performs two pooling operations in parallel, average pooling and maximum pooling. Such an operation enables us to obtain two 2D feature maps with different spatial information. The two feature maps are connected, and 1 × 1 convolution and the Sigmoid activation function are performed on the connected features in turn to generate the final spatial attention. The obtained attention is multiplied element-wise with t_i^e, and the dimension is

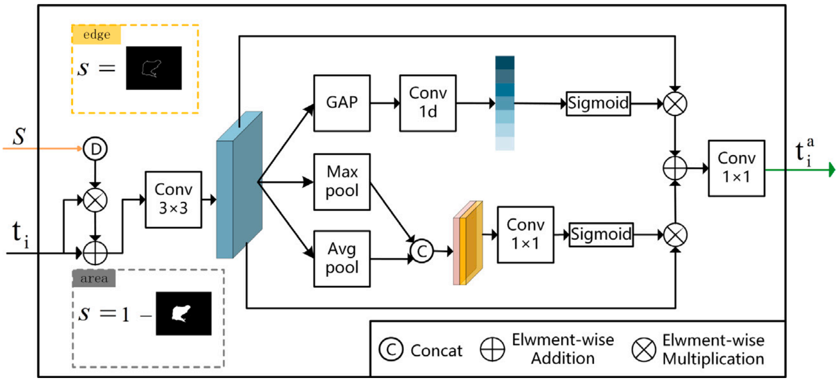


Fig. 4. The architecture of BGM is used to better integrate auxiliary information such as edges and backgrounds.

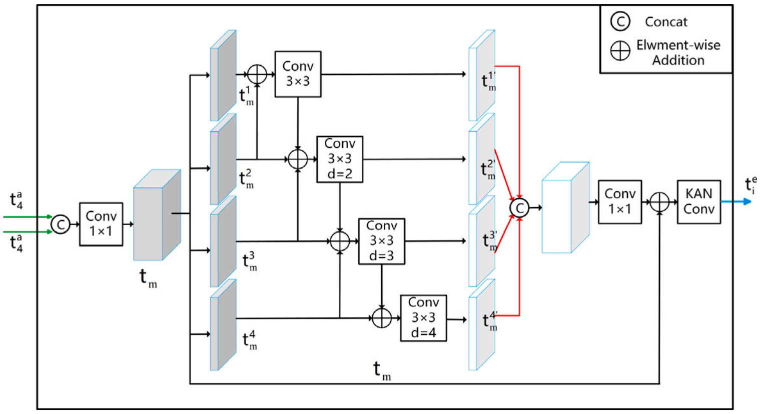


Fig. 5. The architecture of KFFM is used to explore the intrinsic relationship of features and perform feature fusion more effectively.

changed through a 1 × 1 convolution operation to obtain the output result $t^a_{i_l}$ of the lower branch, as shown below:

$$t^a_{i_l} = \sigma(T_{conv1}([Max(t^e_i), Avg(t^e_i)]) \otimes t^e_i), \tag{14}$$

where T_{conv1} represents a 1 × 1 convolution, [*] represents a connection operation, and σ represents a Sigmoid function. After processing the two branches, the approximate area of the camouflaged object t^a_i can be obtained, namely:

$$t^a_i = ((T_{conv1}(t^a_{i_h} \oplus t^a_{i_l}))), \tag{15}$$

The processing of background information t_a is the same as the integration idea of edge clues t_e. When the input of BGM is background information, the dual-branch attention mechanism we proposed can highlight key channels and effectively suppress redundant channels or noise, thereby reducing the impact of background noise.

3.5. Fusion components

In camouflaged object detection, when features with different auxiliary information are fused, the differences between these features may destroy some structures of the target. To solve this problem, we designed a feature fusion component, which includes three Feature Fusion Modules (FFM) and a KAN convolutional Feature Fusion Module (KFFM). This design fully considers the semantic correlation between features of different granularity and types, thereby enhancing the retention of the target structure and achieving the purpose of better recovery of the target object. As shown in Fig. 5, taking the KFFM processing of t^a_4 as an example, we first connect the two input t^a_4 features. Then apply 1 × 1 convolution to obtain the low-level fused feature t_m. Next, t_m is evenly divided into four feature maps of the same shape, denoted as $t^i_m (i = 1, 2, 3, 4)$. Then the segmented features are fused according to the proximity principle, and the fused feature maps are processed through a series of dilated convolutions to optimize the mutual influence between adjacent features and strengthen the correlation. The formula is as follows:

$$t^{j'}_m = T^j_{conv3}(t^{(j-1)'}_m \oplus t^j_m \oplus t^{(j+1)}_m), \quad j \in \{1, 2, 3, 4\}. \tag{16}$$

Here, T^j_{conv3} represents a 3 × 3 dilated convolution. In our experiments, the dilation rates of these convolutions are set to $\{1, 2, 3, 4\}$. The features $t^{j'}{}_m$ are then concatenated. After that, we convolve the concatenated result through a 1 × 1 convolution, and the resulting feature is added to the input feature t_m. The result of the addition is processed by the KAN convolution layer (Liu et al., 2024) (as shown in Fig. 6). KAN convolution represents a highly nonlinear convolution operation that can capture complex dependencies and is well suited for features that require high-level abstraction and complex interactions. The final cross-scale fusion feature t^c_i is expressed as follows:

$$t^c_i = T_{Kconv3}(T_{conv1}([t^{j'}_m]) \oplus t_m), \tag{17}$$

Where [*] represents the connection of feature *, and t^c_i is the final result produced by KFFM. The structure of FFM is similar to KFFM, and FFM enhances feature representation through cross-scale interaction. When using FFM, only replace the KAN convolution (Liu et al., 2024) with the standard 3 × 3 convolution. The output expression of FFM is as follows:

$$t^c_i = T_{conv3}(T_{conv1}([t^{j'}_m]) \oplus t_m), \tag{18}$$

However, it should be noted that the high-level features input to FFM need to be upsampled before they can be connected for subsequent operations. Finally, we can get preliminary prediction results from the three FFM modules. After two-dimensional convolution and upsampling, the final prediction result of the camouflaged object is

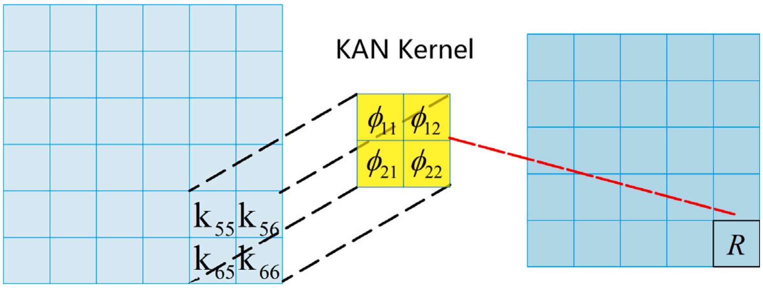


Fig. 6. Illustration of KAN convolution. KAN convolution works similarly to convolution, but instead of adding the dot product between the kernel and the corresponding pixel in the image, a learnable nonlinear activation function is applied to each element and the results are added.

$p_i (i \in 1, 2, 3)$. The KAN convolution expression is as follows, where ϕ represents the activation function.

$$R = \phi_{11}(k_{55}) + \phi_{12}(k_{56}) + \phi_{21}(k_{65}) + \phi_{22}(k_{66}), \tag{19}$$

3.6. Loss function

The loss function consists of two parts: the detection loss of the camouflaged object (G_o) and the detection loss of the camouflaged object boundary (G_e). (L^W_{BCE}) is used to measure the difference between probability distributions, while (L^W_{IOU}) is calculated by measuring the overlap rate between the predicted prediction result and the true label (GT). We apply weighted binary cross-entropy loss (L^W_{BCE}) and weighted IOU loss (L^W_{IOU}) to the camouflaged object mask, which effectively increases the weight distribution of difficult sample pixels. In addition, the dice loss L_{dice} is applied to the camouflaged object boundary to enhance the network's ability to segment fine structures. Therefore, the final objective function L_{total} we designed is expressed as follows:

$$L_{total} = \sum_{i=1}^{3} (L^W_{BCE}(P_i, G_o) + L^W_{IOU}(P_i, G_o)) + \lambda * L_{dice}(P_e, G_e) \tag{20}$$

where P_i is the prediction of the camouflage object, P_e is the prediction of the camouflaged object boundary, and λ is a weighting parameter set to 3 in our experiments.

4. Experiment

4.1. Implementation details

We implemented and trained the model on an NVIDIA GeForce RTX 3090 GPU using the PyTorch framework. During training, all input images were resized to 520 × 520, the batch size was set to 8, the initial learning rate was set to 0.0001, and the AdamW optimizer was used for optimization. The training epochs were 30 times in total, and the learning rate was gradually reduced using a poly learning rate adjustment strategy with a power of 0.9. During the testing phase, the input images were also resized to 520 × 520 for inference, and for the final evaluation, the images were restored to their original size.

4.2. Datasets & evaluation metrics

To ensure consistency with previous works, we use three widely adopted benchmark datasets: CAMO (Le et al., 2019b), COD10K (Fan et al., 2020), and NC4K (Lv et al., 2021). The CAMO and COD10K datasets are used for both training and evaluation, while the NC4K dataset is used exclusively for testing to further assess the model's generalization ability. Model performance is evaluated using four standard metrics: the structural similarity measure (S_α), which evaluates the structural alignment between the prediction and ground truth and reflects the model's ability to preserve object shape and spatial layout; E-measure (E_ϕ), which jointly considers pixel-level precision and recall in a local-to-global manner, indicating the overall alignment and sharpness of the predicted regions; weighted F-measure (F^w_β), which balances precision and recall while emphasizing important foreground pixels, thus highlighting the model's ability to capture salient regions; and mean absolute error (MAE), which computes the average pixel-wise difference between the predicted map and the ground truth, quantifying the global prediction accuracy. These indicators collectively assess different aspects of the model's performance, including contour accuracy, region-level consistency, and pixel-level fidelity, providing a comprehensive evaluation of the predicted results.

4.3. Compared with SOTA methods

Our proposed MENet is compared with 16 SOTA methods on three test datasets in four common indicators. These methods include SINet, C^2FNet (Chen et al., 2022), PFNet (Mei et al., 2021), R-MGL (Zhai et al., 2021), UGTR (Yang et al., 2021), SegMaR (Jia et al., 2022), BGNet (Sun et al., 2022), ZoomNet (Pang et al., 2022), LSR+ (Lv et al., 2023), EAMNet (Sun et al., 2023), SARNet (Xing et al., 2023), GenSAM (Hu et al., 2024), IPNet (Wang et al., 2024b), SDRNet (Guan et al., 2024), TDNet (Sun and Zhang, 2024), TSNet+ (Chen et al., 2025) and BGMRNet (Ye et al., 2025). To ensure the fairness of the experiment, we use the results provided by the authors of the original paper and evaluate the results using the same evaluation tools.

4.3.1. Quantitative evaluation

Table 1 reports the quantitative results comparing various SOTA methods with our method. It is obvious that the overall performance of our method on three commonly used test sets is better than that of all other SOTA models. Specifically, compared with the second-ranked SARNet, our method improves (S_α) by 1% on average, maintains the same performance on (E_ϕ), and outperforms (F^w_β) increased by 1.50% on average. Compared with IPNet released at ICLR 2024, our method improves (S_α) by 3.50%, (E_ϕ) by 1.63%, and (F^w_β) increases by 2.97%. As shown in Table 2, we further enrich the comparison by including the latest 2025 SOTA model, TSNet+. Our model not only surpasses TSNet+ in accuracy across all evaluated metrics but also exhibits a more compact model size. Specifically, relative to BGNet, our method achieves an average improvement of 7.87% in F^w_β across the three benchmark datasets, while reducing the number of parameters by 35.22% and FLOPs by 25.08%. Moreover, our model matches SARNet in terms of parameter count and FLOPs but consistently outperforms it across multiple evaluation metrics. These enhancements stem from our carefully designed and efficient network architecture.

4.3.2. Qualitative evaluation

Fig. 8 shows a qualitative evaluation with other COD methods. The test objects cover four major categories: land, sea, air, and amphibians, and include complex conditions such as natural camouflage, artificial camouflage, target occlusion, and multi-target scenes. By visually comparing the above objects, we can intuitively see that our method

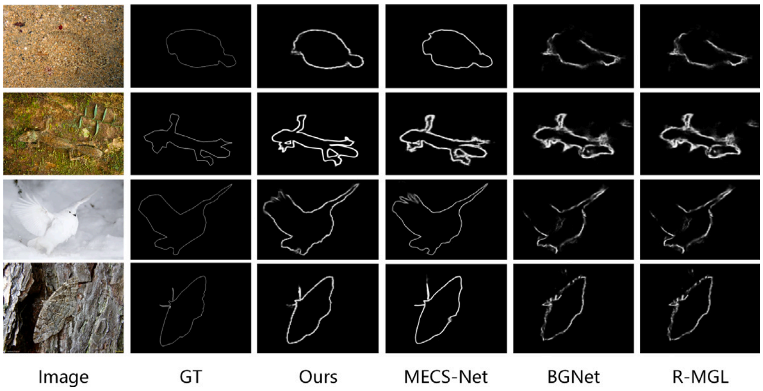


Fig. 7. Example of edge extraction obtained by our designed Context-Aware Edge Module (CAEM).

Table 1
We perform quantitative comparisons with state-of-the-art COD methods on three datasets. '↑' and '↓' indicate that larger values are better and smaller values are better, respectively. **bold** indicates the best result, and — indicates that the model was not tested on this dataset.

Method	Pub./Year	CAMO				COD10K				NC4K			
		S_α ↑	E_ϕ ↑	F_β^w ↑	M ↓	S_α ↑	E_ϕ ↑	F_β^w ↑	M ↓	S_α ↑	E_ϕ ↑	F_β^w ↑	M ↓
SINet	CVPR'20	0.751	0.771	0.606	0.100	0.771	0.806	0.551	0.051	0.808	0.871	0.723	0.058
C^2FNet	IJCAI'21	0.796	0.854	0.719	0.080	0.813	0.890	0.686	0.036	0.838	0.897	0.762	0.049
UGTR	ICCV'21	0.783	0.821	0.683	0.086	0.818	0.850	0.667	0.035	0.839	0.874	0.749	0.052
SegMaR	CVPR'22	0.815	0.872	0.742	0.071	0.833	0.895	0.724	0.033	–	–	–	–
BGNet	IJCAI'22	0.812	0.870	0.749	0.073	0.831	0.901	0.722	0.033	0.851	0.907	0.788	0.044
ZoomNet	CVPR'22	0.820	0.892	0.752	0.066	0.838	0.911	0.729	0.029	0.853	0.912	0.784	0.059
EAMNet	ICME'23	0.831	0.890	0.763	0.064	0.839	0.907	0.733	0.029	0.862	0.916	0.801	0.040
SARNet	TCSVT'23	0.868	**0.927**	0.828	0.047	0.864	0.931	0.777	0.024	0.886	**0.937**	0.842	0.032
TDNet	IA'24	0.789	0.855	0.683	0.046	0.768	0.849	0.664	0.048	–	–	–	–
GenSAM	AAAI'24	0.719	0.775	0.659	0.113	0.775	0.838	0.681	0.067	–	–	–	–
IPNet	EAAI'24	0.864	0.924	0.836	0.047	0.850	0.922	0.785	0.026	–	–	–	–
SDRNet	KBS'24	0.872	0.924	0.826	0.049	0.871	0.924	0.785	0.023	0.889	0.934	0.842	0.032
TSNet+	IJON'25	0.834	0.885	0.774	0.066	0.854	0.913	0.754	0.028	0.866	0.914	0.803	0.040
BGMR-Net	VC'25	0.870	0.921	0.834	0.048	0.860	0.921	0.770	0.025	0.885	0.932	0.848	0.033
MENet	Ours	**0.874**	0.920	**0.836**	**0.046**	**0.881**	**0.932**	**0.806**	**0.022**	**0.892**	0.930	**0.850**	**0.032**

Table 2
To enhance the persuasiveness of the experimental results, we utilized the same evaluation tool to assess the outcomes, ensuring that the results were evaluated under the same input image size.

Method	In-size	Param	FLOPs	CAMO				COD10K				NC4K			
				S_α ↑	E_ϕ ↑	F_β^w ↑	M ↓	S_α ↑	E_ϕ ↑	F_β^w ↑	M ↓	S_α ↑	E_ϕ ↑	F_β^w ↑	M ↓
BGNet	520^2	77.8M	93.3G	0.812	0.870	0.749	0.073	0.831	0.901	0.722	0.033	0.851	0.907	0.788	0.044
SARNet	520^2	48.9M	79.5G	0.868	**0.927**	0.828	0.047	0.864	0.931	0.777	0.024	0.886	**0.937**	0.842	**0.032**
TSNet+	520^2	52.3M	51.3G	0.834	0.885	0.774	0.066	0.854	0.913	0.754	0.028	0.866	0.914	0.803	0.040
Ours	520^2	50.5M	70.5G	**0.874**	0.920	**0.836**	**0.046**	**0.881**	**0.932**	**0.806**	**0.022**	**0.892**	0.930	**0.850**	**0.032**

provides accurate predictions for each camouflaged object, such as bird feathers and frog webbed feet, which have high prediction accuracy. For these detection objects with complex structures and high similarity to the environment, our model performs well in locating edge information while effectively reducing the interference caused by background noise, thereby obtaining a finer object structure.

4.3.3. *Boundary exploration*

Fig. 7 separately demonstrates the advantages of our method in edge localization and provides a visual comparison with BGNet. Judging from the results, although BGNet significantly improves the performance of the COD task by extracting edge features, it still has problems with edge blur and inaccurate positioning, resulting in unsatisfactory prediction results. Experiments show that our method effectively solves the problem of inaccurate boundary positioning and has an excellent ability to capture the fine structure of objects, enabling MENet to achieve excellent performance in camouflaged object detection.

4.4. *Ablation study*

To evaluate the contributions of the key components, we conducted a comprehensive ablation study involving BGM, AD, CAEM, FFM, and KFFM. The baseline model (“B”) is built upon the PVT-V2 backbone, utilizing basic connection and convolution operations. Notably, the AD and CAEM modules, as well as FFM and KFFM, work synergistically, and their combined effect is explicitly annotated in the ablation experiments (see Table 3). The results offer insights into both the individual and joint impacts of these components on overall performance. For visual comparison, outputs from models (a) to (e) are shown in Fig. 9.

4.4.1. *Effectiveness of auxiliary clues (AD and CAEM)*

The CAEM module captures fine-grained edge structures, helping the model distinguish target boundaries in challenging scenes with high background similarity. Meanwhile, the AD module suppresses irrelevant background information, focusing attention on salient foreground regions. These two modules are deeply coupled and designed

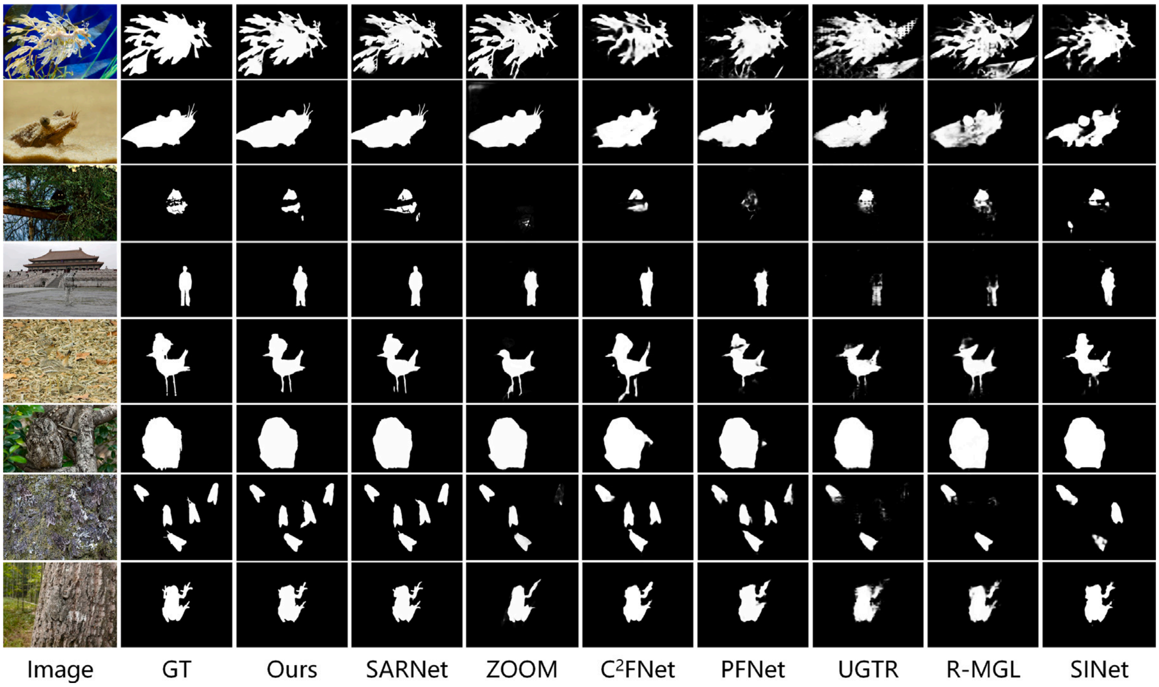


Fig. 8. Visual comparison between our method and other state-of-the-art methods. It is clear that our method can accurately predict various camouflaged objects, and the overall structure and details of the prediction results are better than other state-of-the-art methods.

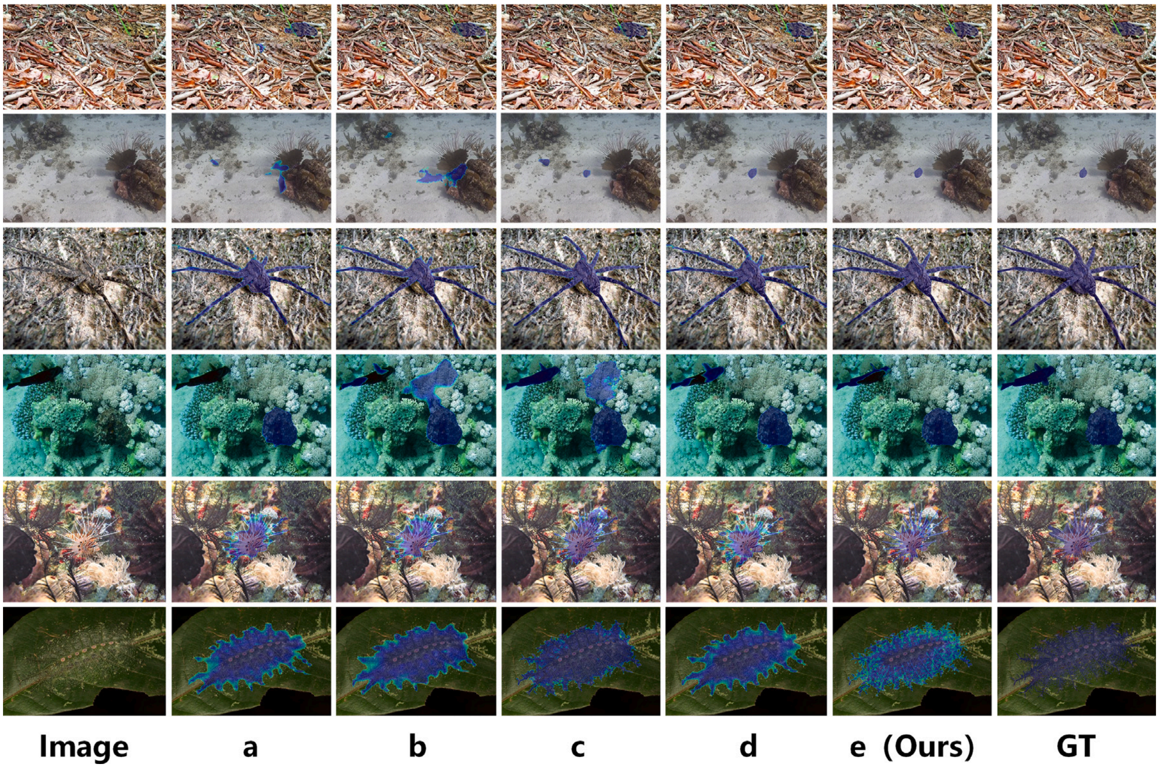


Fig. 9. Visual comparison of target recognition in each part. a-e corresponds to Table 3.

to function collaboratively: CAEM provides structural cues, while AD reinforces semantic focus, forming an effective edge-background interaction mechanism.

To further demonstrate their synergistic benefits, we provide intermediate feature visualizations in Fig. 11, comparing models with and without the combined AD and CAEM modules. The visual results clearly show that the joint inclusion of these modules enhances both boundary precision and object localization. Consistently, Table 3 shows that the model incorporating both modules (model d) achieves substantial performance gains over the baseline (model a), with 3.00% improvement in S_α, 5.30% in E_ϕ, and 8.60% in F_β^w on COD10K. These findings confirm that our edge-background fusion

Table 3
Ablation study of MENet. “B” denotes the baseline model. Key modules evaluated include BGM, AD, CAEM, FFM, and KFFM. The “&” symbol marks module pairs applied jointly due to their synergistic effect (i.e., AD with CAEM, FFM with KFFM). Best results per metric are highlighted in bold.

Model	Method	CAMO				COD10K				NC4K			
		$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^w \uparrow$	$M \downarrow$
a	B	0.862	0.900	0.777	0.059	0.848	0.867	0.696	0.033	0.872	0.901	0.776	0.045
b	B+BGM	0.867	0.900	0.781	0.058	0.854	0.867	0.699	0.032	0.874	0.900	0.776	0.045
c	B+BGM+AD&CAEM	0.872	0.919	0.818	0.050	0.878	0.920	0.776	0.026	0.889	0.928	0.828	0.037
d	B+FFM&KFFM	0.873	0.916	0.825	0.049	0.878	0.920	0.782	0.024	0.889	0.927	0.833	0.035
e	MENet(Ours)	**0.874**	**0.920**	**0.836**	**0.046**	**0.881**	**0.932**	**0.806**	**0.022**	**0.892**	**0.930**	**0.850**	**0.032**

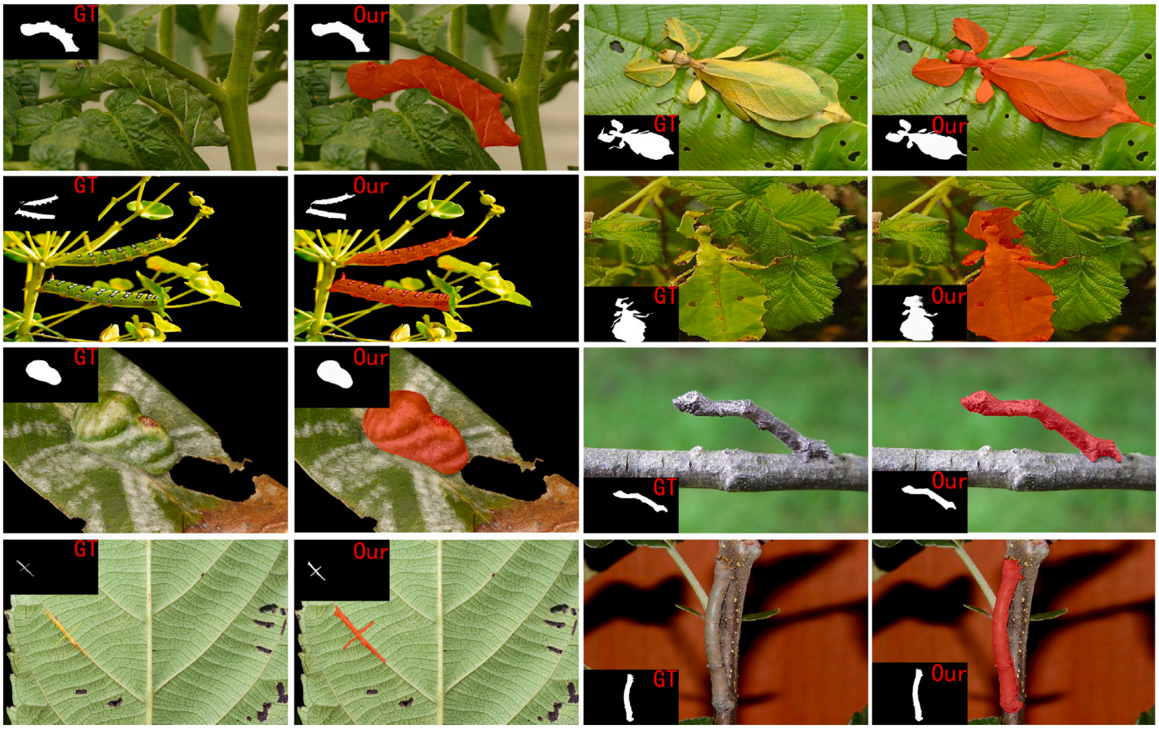


Fig. 10. This figure illustrates the effectiveness of MENet in detecting camouflaged pests within complex agricultural environments. The selected images cover representative camouflage scenarios, such as cabbage loopers with background-similar colors (e.g., Row 1, Column 1) and highly concealed small pests (e.g., Row 4, Column 1), both of which present significant challenges due to strong background interference. MENet demonstrates strong boundary awareness by accurately extracting object contours (e.g., Row 3, Column 1), while effectively suppressing background noise to reduce false positives. Even in cases involving small or multiple targets, the model consistently produces precise predictions that closely match the ground truth (GT). These results validate the practical applicability and robustness of MENet for real-world agricultural pest monitoring tasks.

strategy significantly boosts the model’s detection capability for camouflaged objects.

4.4.2. Effectiveness of KFFM and FFM

In order to better utilize auxiliary information and enhance the correlation between features, we designed FFM and KFFM to achieve better fusion of multi-level features with different auxiliary cues. In Table 3, the model (e) containing KFFM and FFM shows significantly better performance than the model (a) without these modules. The model achieved significant improvements in all indicators of all datasets. Therefore, the addition of KFFM and FFM better integrates features of different levels and types, enhances the model’s ability to explore complete objects and capture details, and especially improves the ability to detect camouflaged objects that are difficult to detect in complex scenes.

5. Application

Pests are the main factor causing crop yield reduction and disease spread. Pest outbreaks will also lead to increased pesticide use and cause pollution to the ecological environment. Therefore, achieving efficient and accurate pest detection is crucial for smart agriculture and sustainable development.

Although pest detection in smart agriculture has made some progress, traditional visual methods still have difficulty in identifying camouflaged pests. To this end, this paper proposes a COD model, MENet, which can be used to detect camouflaged pests. The model effectively improves the recognition ability of camouflaged targets by locating edges and suppressing backgrounds. The experimental results are shown in Fig. 10, which shows that MENet has excellent detection performance in complex backgrounds.

In addition, MENet can be applied to scenarios such as smart farmland monitoring, drone inspections, and smart pesticide application to achieve automatic identification and precise prevention and control of pests and promote the development of smart agriculture.

6. Conclusion

In this paper, we study the localization of precise boundaries and the reduction of complex background interference to improve the performance of camouflaged object detection (COD). To this end, we

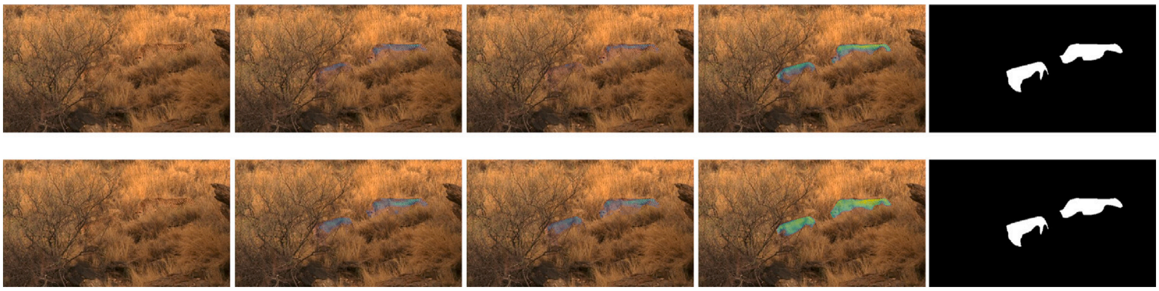


Fig. 11. Visualization of intermediate feature heatmaps comparing the effects of the AD and CAEM modules on object localization and boundary awareness. The upper part shows the results without the integration of the AD and CAEM modules, while the lower part presents the results with these modules included. Each section contains the original image, three predicted outputs ranked by accuracy, and the ground truth (GT) for comparison. The visual results clearly demonstrate that the model without the AD and CAEM modules performs poorly under complex background interference, resulting in imprecise and incomplete boundary detection. In contrast, with the inclusion of the AD and CAEM modules, the model not only achieves more accurate boundary localization but also maintains continuous focus on the objects within the boundaries, highlighting the synergistic benefits of the edge-background fusion strategy.

specifically design modules such as CAEM, AD, BGM, and KFFM and integrate these components into a multi-layer feature fusion network (MENet) dedicated to edge localization and background denoising. By accurately locating the edges of detected targets and suppressing similar backgrounds, our method achieves accurate prediction of camouflaged objects while capturing finer details and preserving the complete structure of the object. Moreover, our proposed model is also effective in pest detection (experimental results are shown in Fig. 10), indicating that it has great potential in smart agricultural pest control applications.

CRediT authorship contribution statement

Guangjian Zhang: Investigation, Data curation. **Zhengming Yang:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology. **Yong Wang:** Resources, Project administration, Funding acquisition, Conceptualization. **Yuliang Chen:** Writing – review & editing, Supervision. **Duoqian Miao:** Validation, Supervision.

Declaration of competing interest

All the authors declare that they have no competing financial interests or personal relationships that could influence the work reported in this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant 61976158 and Grant 61673301.

Data availability

Data will be made available on request.

References

Chen, Geng, Liu, Si-Jie, Sun, Yu-Jia, Ji, Ge-Peng, Wu, Ya-Feng, Zhou, Tao, 2022. Camouflaged object detection via context-aware cross-level fusion. IEEE Trans. Circuits Syst. Video Technol..

Chen, Tianyou, Ruan, Hui, Wang, Shaojie, Xiao, Jin, Hu, Xiaoguang, 2025. A three-stage model for camouflaged object detection. Neurocomputing 614, 128784.

Fan, Deng-Ping, Ji, Ge-Peng, Sun, Guolei, Cheng, Ming-Ming, Shen, Jianbing, Shao, Ling, 2020. Camouflaged object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2777–2787.

Guan, Juwei, Fang, Xiaolin, Zhu, Tongxin, Qian, Weiqi, 2024. Sdrnet: Camouflaged object detection with independent reconstruction of structure and detail. Knowl.-Based Syst. 112051.

Guo, Cunhan, Huang, Heyan, 2025. Enhancing camouflaged object detection through contrastive learning and data augmentation techniques. Eng. Appl. Artif. Intell. 141, 109703.

Hou, Qibin, Zhou, Daquan, Feng, Jiashi, 2021. Coordinate attention for efficient mobile network design. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13713–13722.

Hu, Jian, Lin, Jiayi, Gong, Shaogang, Cai, Weitong, 2024. Relax image-specific prompt requirement in SAM: A single generic prompt for segmenting camouflaged objects. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, (11), pp. 12511–12518.

Jia, Qi, Yao, Shuilian, Liu, Yu, Fan, Xin, Liu, Risheng, Luo, Zhongxuan, 2022. 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.

Le, Trung-Nghia, Nguyen, Tam V, Nie, Zhongliang, Tran, Minh-Triet, Sugimoto, Akihiro, 2019a. Anabranch network for camouflaged object segmentation. Comput. Vis. Image Underst. 184, 45–56.

Le, Trung-Nghia, Nguyen, Tam V, Nie, Zhongliang, Tran, Minh-Triet, Sugimoto, Akihiro, 2019b. Anabranch network for camouflaged object segmentation. Comput. Vis. Image Underst. 184, 45–56.

Liu, Ziming, Wang, Yixuan, Vaidya, Sachin, Ruehle, Fabian, Halverson, James, Soljačić, Marin, Hou, Thomas Y, Tegmark, Max, 2024. Kan: Kolmogorov-arnold networks. arXiv preprint arXiv:2404.19756.

Lv, Yunqiu, Zhang, Jing, Dai, Yuchao, Li, Aixuan, Barnes, Nick, Fan, Deng-Ping, 2023. Towards deeper understanding of camouflaged object detection. IEEE Trans. Circuits Syst. Video Technol..

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

Mei, Haiyang, Ji, Ge-Peng, Wei, Ziqi, Yang, Xin, Wei, Xiaopeng, Fan, Deng-Ping, 2021. Camouflaged object segmentation with distraction mining. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8772–8781.

Pang, Youwei, Zhao, Xiaoqi, Xiang, Tian-Zhu, Zhang, Lihe, Lu, Huchuan, 2022. 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.

Sun, Dongyue, Jiang, Shiyao, Qi, Lin, 2023. Edge-aware mirror network for camouflaged object detection. In: 2023 IEEE International Conference on Multimedia and Expo. ICME, IEEE, pp. 2465–2470.

Sun, Yujia, Wang, Shuo, Chen, Chenglizhao, Xiang, Tian-Zhu, 2022. Boundary-guided camouflaged object detection. In: IJCAI. pp. 1335–1341.

Sun, Hang, Zhang, Cong, 2024. Double-branch camouflaged object detection method based on intra-layer and inter-layer information integration. IEEE Access.

Wang, Yaming, Chen, Jiatong, Fang, Xian, Jiang, Mingfeng, Ma, Jianhua, 2024a. Dual cross perception network with texture and boundary guidance for camouflaged object detection. Comput. Vis. Image Underst. 248, 104131.

Wang, Xin, Ding, Jiajia, Zhang, Zhao, Xu, Junfeng, Gao, Jun, 2024b. Ipnet: Polarization-based camouflaged object detection via dual-flow network. Eng. Appl. Artif. Intell. 127, 107303.

Wang, Yong, Pu, Jianfei, Miao, Duoqian, Zhang, L, Zhang, Lulu, Du, Xin, 2024c. SCGRFuse: An infrared and visible image fusion network based on spatial/channel attention mechanism and gradient aggregation residual dense blocks. Eng. Appl. Artif. Intell. 132, 107898.

Wang, Wenhai, Xie, Enze, Li, Xiang, Fan, Deng-Ping, Song, Kaitao, Liang, Ding, Lu, Tong, Luo, Ping, Shao, Ling, 2021. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 568–578.

Xing, Haozhe, Gao, Shuyong, Wang, Yan, Wei, Xujun, Tang, Hao, Zhang, Wenqiang, 2023. Go closer to see better: Camouflaged object detection via object area amplification and figure-ground conversion. IEEE Trans. Circuits Syst. Video Technol. 33 (10), 5444–5457.

Yang, Fan, Zhai, Qiang, Li, Xin, Huang, Rui, Luo, Ao, Cheng, Hong, Fan, Deng-Ping, 2021. Uncertainty-guided transformer reasoning for camouflaged object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4146–4155.

Ye, Qian, Li, Qingwu, Huo, Guanying, Liu, Yan, Zhou, Yan, 2025. Boundary-guided multi-scale refinement network for camouflaged object detection. Vis. Comput. 1–27.

Zhai, Qiang, Li, Xin, Yang, Fan, Chen, Chenglizhao, Cheng, Hong, Fan, Deng-Ping, 2021. Mutual graph learning for camouflaged object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12997–13007.

Guangjian Zhang received the master’s degree from Guizhou University in 2004. He is currently an Associate Professor with the Liangjiang School of Artificial Intelligence, Chongqing University of Technology. His research interests include intelligent systems and their applications.

Zhengming Yang is currently pursuing a master’s degree at the School of Artificial Intelligence, Chongqing University of Technology. His research interests include computer vision and camouflaged object detection.

Yong Wang is a professor at the School of Artificial Intelligence, Chongqing University of Technology, Chongqing, China. His research interests include artificial intelligence and multimedia technology.

Yuliang Chen received his Ph.D. degree from Louisiana State University in 2010. He is a software engineer at The Bank of New York Mellon Corporation in New York City, USA. His research interests include artificial intelligence and cloud computing.

Duoqian Miao received the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 1997. He is currently a Professor with the School of Electronics and Information Engineering, Tongji University, Shanghai, China.