Mixture-of-Queries Transformer: Camouflaged Instance Segmentation via Queries Cooperation and Frequency Enhancement

Weiwei Feng[1,2] , **Nanqing Xu**[3] , **Tengfei Liu**[2*] and **Weiqiang Wang**[2]
[1]Zhejiang University
[2]Ant Group
[3]Independent Researcher
{fengww,xnq}@mail.ustc.edu.cn, aaron.ltf@antgroup.com, weiqiang.wwq@antgroup.com

Abstract

Due to the high similarity between camouflaged instances and the surroundings and the widespread camouflage-like scenarios, the recently proposed camouflaged instance segmentation (CIS) is a challenging and relevant task. Previous approaches achieve some progress on CIS, while many overlook camouflaged objects' color and contour nature and then decide on each candidate instinctively. In this paper, we contribute a Mixture-of-Queries Transformer (MoQT) in an end-to-end manner for CIS based on two key designs (a Frequency Enhancement Feature Extractor and a Mixture-of-Queries Decoder). First, the Frequency Enhancement Feature Extractor is responsible for capturing the camouflaged clues in the frequency domain. To expose camouflaged instances, the extractor enhances the effectiveness of contour, eliminates the interference color, and obtains suitable features simultaneously. Second, a Mixture-of-Queries Decoder utilizes multiple newly initialized experts of queries (a group of queries considered an expert) in each layer for spotting camouflaged characteristics with cooperation. These experts collaborate to generate outputs with the mixture-of-queries mechanism, refined hierarchically to a fine-grained level for more accurate instance masks. Coupling these two components enables MoQT to use multiple experts to integrate effective clues of camouflaged objects in both spatial and frequency domains. Extensive experimental results demonstrate our MoQT outperforms 19 state-of-the-art CIS approaches on both COD10K and NC4K datasets.

1 Introduction

Camouflage is a naturally evolved strategy for animals to hide themselves via adapting their body's coloring to match the surroundings, which is used for hunting prey or avoiding detection by natural enemies, as shown in Figure 1(a). Since there is a lot of demand for understanding the widespread camouflage-like scenarios, (*e.g.*, polyp segmentation [Fan *et al.*, 2020b], lung infection segmentation [Fan *et al.*, 2020c], search-and-rescue work [Turić *et al.*, 2010], manipulated image/video detection and segmentation [Xu and Feng, 2023; Zhang *et al.*, 2024]), the task of predicting the location and instance-level masks of camouflaged objects (*i.e.*, Camouflaged Instance Segmentation, CIS) has been proposed. Therefore, CIS is worth studying and has gradually received more attention in recent years. However, it also has challenges due to high intrinsic similarities between the target objects and the background.

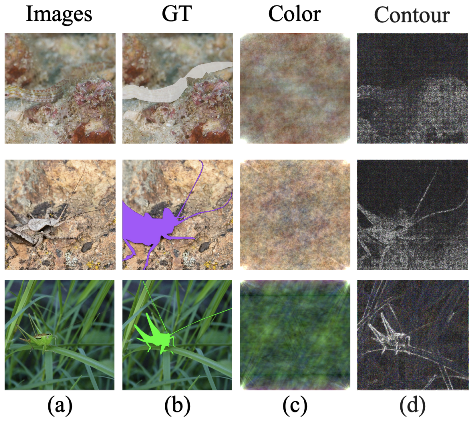


Figure 1: The Nature of Camouflaged Objects. Careful contrast of the camouflaged inputs (a) and the corresponding ground truth (b), color (c) (reconstructed with only amplitude component of Fourier transformation) and contour (d) (reconstructed with only phrase component of Fourier transformation) information shows the important priori principle of camouflaged objects: Low-level statistics like color contain more information from the surroundings while high-level semantics like contour tend to preserve more camouflaged characteristics.

Compared to the tremendous development in generic instance segmentation [Bolya *et al.*, 2019; Wang *et al.*, 2020a; Wang *et al.*, 2020b; Ren *et al.*, 2015; He *et al.*, 2017; Cai and Vasconcelos, 2019; Chen *et al.*, 2019], camouflaged instance segmentation remains an under-explored issue, and only a few efforts have been made to study it in the past three years [Pei *et al.*, 2022; Luo *et al.*, 2023; Dong *et al.*, 2023; Li *et al.*, 2024; Le *et al.*, 2023]. CFL [Le *et al.*, 2021] is a first attempt. It is a two-stage method that fuses general in-

[*] Corresponding author

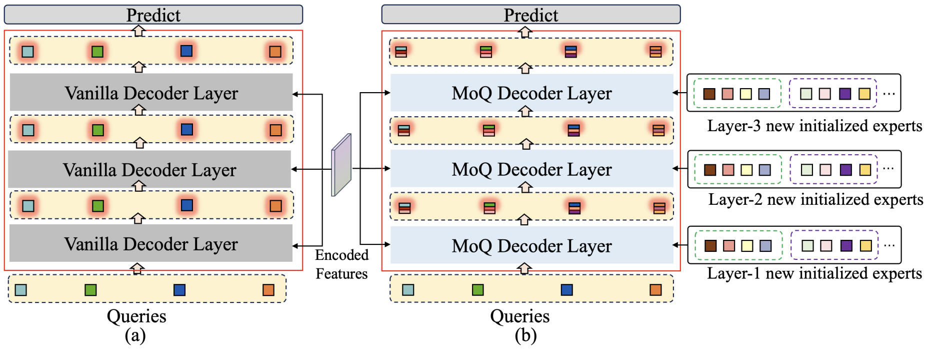


Figure 2: The illustration of the standard transformer decoder and Mixture-of-Queries Transformer decoder. (a) There are only a group of queries regarded as candidates initialized before feed into standard transformer decode. (b) There are multiple new initialized groups of queries (a group of queries considered an expert) in each Mixture-of-Queries decoder layer for explicitly refining the candidates, where each layer includes a standard decoder layer and a mixture-of-queries layer.

stance segmentation methods for camouflaged instance segmentation but has limited performance. Subsequently, OSFormer [Pei *et al.*, 2022] is proposed as the first one-stage method for CIS. It takes advantage of a transformer network, which achieves a flexible framework that can be trained end-to-end for camouflaged instance segmentation. Recently, DCNet [Luo *et al.*, 2023] has been proposed to segment camouflaged instances via explicit de-camouflaging and achieves CIS by jointly modeling pixel-level camouflage decoupling and instance-level camouflage suppression. In the same period, UQFormer [Dong *et al.*, 2023] adopts a typical DETR-like architecture [Carion *et al.*, 2020] and exploits the information on object edges.

Although the aforementioned works have made some progress, two fundamental inspirations have not been taken into account: the priori of camouflage principles and the human habit of segmenting camouflaged instances. **(1)** The priori of camouflage principles: Only when you know how to camouflage can you see through camouflage. Many years ago, zoologists discovered that animals can camouflage themselves by matching their colors or patterns with the background. To look deeper into camouflage, we sample some images of camouflaged animals and analyze them thoroughly. Since it is hard to spot camouflaged objects in the surroundings, we perform the Fourier transform on these images to discover some clues in the frequency domain. We first decompose these camouflaged images into phrase and amplitude components and reconstruct images from only phrase component and amplitude component, respectively, presented in Figure 1. It is easy to find that the phase component of the Fourier spectrum preserves high-level semantics (contours and semantics) of original images, while the amplitude component contains low-level statistics (colors and styles). Therefore, enhancing the influence of contours and eliminating the interference of colors would certainly benefit the performance on CIS. **(2)** The human habit of segmenting camouflaged instances: When humans segment a camouflaged image, their visual system instinctively sweeps across the scene and determines some candidates. Then, the visual system gradually searches for valuable clues throughout the scene to obtain accurate segmentation masks. For some heavily camouflaged scenes with highly accurate segmentation like some medical image datasets [Fu *et al.*, 2019], it may even refine the masks labeled by multiple experts. Gradually refining and integrating the decisions of multiple experts are potentially effective for CIS. Therefore, it makes sense to take full advantage of both inspirations of **(1)** and **(2)** for improving the performance of the CIS task.

Motivated by the above discussions, we proposed a Mixture-of-Queries Transformer (MoQT) trained in an end-to-end manner for CIS, which includes a Frequency Enhancement Feature Extractor (FEFE) based on modeling the colors and contours of camouflaged instances and a Mixture-of-Queries Decoder (MoQ Decoder) in transformer architecture referring to the segmentation process of multi-experts collaboration. First, inspired by the camouflage principles discussed above, we design a Frequency Enhancement Feature Extractor to capture more clues of camouflaged instances in the frequency domain. Specifically, we propose to adopt Fourier spectrum amplitude and phase to model image color information and contour information, respectively, as shown in Figure 1. With the help of color and contour information, we design a contour enhancement module and a color removal module, which can increase the contour effect while eliminating color interference. This mechanism in the frequency domain is suitable for debunking the principle of animal camouflaging, which is reasonable for providing gains on CIS. Second, for the Mixture-of-Queries Decoder, which is different from the standard DETR framework, we design multiple expert groups of queries inspired by the success of Mixture-of-Experts (MoE) mechanism. In DETRs, object queries are wonderful designs in decoders, which have two roles: a) candidates for objects and b) interaction with encoded features. And there are only a group of queries regarded as candidates initialized before feeding into the standard transformer decoder, as shown in Figure 2(a). These candidates are refined for segmentation by learning the parameters of each decoder layer, where each decoder layer determines whether the refined candidates are suitable for final prediction. Differently, we design multiple groups of queries (a group of queries considered an expert) in each decoder layer for explicitly refining the candidates, which can be illustrated in Figure 2(b). In each layer, there is an expert from the last layer and several new initialized experts. And we propose a gating network in each decoder layer to mix these experts, deciding which newly initialized experts are selected to refine candidates explicitly for the next layer forwarding. The gating network accepts the encoded features as input, and the output is the weights of recombination of the various experts. This mechanism can refine outputs hierarchically to a fine-grained level via a mixture of experts, which can generate more accurate instance masks, as shown in Figure 2(b). In favor of these two designs, our method can utilize multiple expert queries to integrate effective clues of camouflaged objects in both spatial and frequency domains, which can achieve outstanding segmentation performance.

In summary, our main contributions are three-folds. **(1)** We propose a Mixture-of-Queries Transformer (MoQT) trained in an end-to-end manner for CIS, which takes advantage of the priori of camouflage principles and refers to the human habit of segmenting camouflaged instances. To the best of our knowledge, this is the first attempt to introduce MoE into DETR-like frameworks for segmentation. **(2)** We proposed a Frequency Enhancement Feature Extractor (FEFE) and a Mixture-of-Queries Decoder (MoQ Decoder) for our MoQT, where FEFE is used for color removal and contour enhancement. The MoQ Decoder aims to mix multiple groups of queries hierarchically to provide more accurate predictions. **(3)** Extensive experimental results on COD10K and NC4K show consistent performance gains compared with 19 baseline methods and verify the superiority of our method.

2 Related Work

2.1 Camouflaged Object Detection

Camouflaged Object Detection is usually considered as one of the most important origins of CIS and aims to identify the camouflaged objects from the background and has witnessed the development of art and biology [Fan *et al.*, 2020a; Le *et al.*, 2019]. Early research [Pan *et al.*, 2011; Sengottuvelan *et al.*, 2008] in COD mainly uses handcrafted features (*e.g.*, gradient, texture, and intensity features) to tell the camouflaged objects from their surroundings. Later, deep learning (DL) improves COD's performance in an end-to-end manner, and plenty of DL-based methods [Pang *et al.*, 2022; Yang *et al.*, 2021; Xu *et al.*, 2021; Zhong *et al.*, 2022; Mei *et al.*, 2021; Ren *et al.*, 2021] have been proposed. For example, ZoomNet [Pang *et al.*, 2022] discusses how to capture camouflaged objects in complex surroundings in a multi-scale manner. Moreover, UGTR [Yang *et al.*, 2021] combines the benefits of both Bayesian learning and transformer-based reasoning to handle camouflaged object detection with probabilistic and deterministic information. Some works [Xu *et al.*, 2021; Zhong *et al.*, 2022] even go beyond the RGB domain and explore frequency clues for better performance. In this paper, a Frequency Enhancement Feature Extractor, which refines frequency clues with contour enhancement and color removal, is adopted and allows full rein to both the camouflaged characteristics and the surrounding textures.

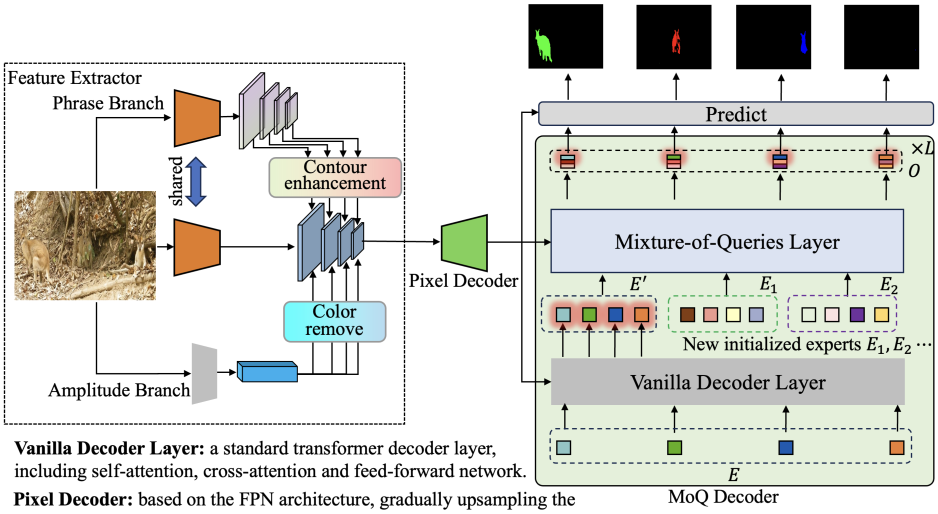
2.2 Camouflaged Instance Segmentation

Camouflage Instance Segmentation (CIS) learns most lessons from traditional instance segmentation. The purpose of instance segmentation is to assign pixel-level mask prediction for various instances. Nowadays, instance segmentation methods can be roughly divided into two parts: One-stage approaches [Bolya *et al.*, 2019; Wang *et al.*, 2020a; Wang *et al.*, 2020b] and two-stage approaches [Ren *et al.*, 2015; He *et al.*, 2017; Cai and Vasconcelos, 2019; Chen *et al.*, 2019]. Two-stage methods apply mask segmentation after proposal region detection, such as Faster R-CNN [Ren *et al.*, 2015], Mask R-CNN [He *et al.*, 2017], Cascade R-CNN [Cai and Vasconcelos, 2019], and HTC [Chen *et al.*, 2019]. CFL [Le *et al.*, 2021], the first attempt in CIS, also applies two-stage instance segmentation methods. However, one-stage methods show faster inference than two-stage methods and achieve comparable performance. For example, YOLACT [Bolya *et al.*, 2019] adopts two parallel tasks to produce nonlocal prototype masks with adaptive coefficients. Furthermore, SOLO [Wang *et al.*, 2020a] and SOLO-v2 [Wang *et al.*, 2020b] predict the instances' center and then decouple the instance masks with kernel feature learning. Recently, researchers have found transformers [Cheng *et al.*, 2021; Cheng *et al.*, 2022] show excellent performance on instance segmentation with the assistance of attention mechanisms and instance-specific prototypes. Therefore, transformer-based methods like OSFormer [Pei *et al.*, 2022], DCNet [Luo *et al.*, 2023] and UQFormer [Dong *et al.*, 2023] utilize transformers in CIS and achieve great progress. Inspired by [Pei *et al.*, 2022; Luo *et al.*, 2023; Dong *et al.*, 2023], our Mixture-of-Queries Transformer (MoQT) introduces a Mixture-of-Queries Decoder (MoQ Decoder) in the transformer decoder to combine the capabilities of multi-experts hierarchically, which enhances camouflage semantics and refines details of instance masks.

3 Method

3.1 Architecture Overview

The overall framework of our proposed model is presented in Figure 3. The whole architecture of our method is a typical MaskFormer-like [Cheng *et al.*, 2021] model, composed of a Frequency Enhancement Feature Extractor (FEFE), a Pixel Decoder, and a Mixture-of-Queries decoder (MoQ Decoder). In the FEFE, we get valuable multi-scale features enhanced by the Fourier transform for revealing the camouflaged clues, where the phase component and amplitude can be used for modeling the information of contours and colors, respectively. We use a contour Enhancement Module (CEM) and a Color Remove Module (CRM) to mine the potential information of contours and eliminate the interference of colors for capturing clues of camouflaged instances. Then, the Pixel Decoder (based on FPN [Lin *et al.*, 2017]) gradually upsamples low-resolution features from the output of the backbone to generate high-resolution per-pixel embeddings. The MoQ Decoder computes from per-pixel embeddings and some initialized experts (a series of queries) to get the output prediction. Specifically, in MoQ Decoder, we propose a Mixture-of-Queries Layer (MoQ Layer) after each decoder layer and



Vanilla Decoder Layer: a standard transformer decoder layer, including self-attention, cross-attention and feed-forward network.

Pixel Decoder: based on the FPN architecture, gradually upsampling the features to generate per-pixel embeddings for feeding into the MoQ decoder.

Figure 3: The Architecture of Our Proposed Model. Our method mainly consists of a Frequency Enhancement Feature Extractor (FEFE), a Pixel Decoder, and a Mixture-of-Queries Decoder (MoQ Decoder). (1) The FEFE captures suitable camouflaged clues with the contour enhancement and color remove modules in the frequency domain. (2) The Pixel Decoder is the same as previous works, based on the FPN architecture, which is used to gradually upsample low-resolution features from the output of the FEFE to generate high-resolution per-pixel embeddings. (3) The MoQ Decoder determines object candidates by multiple cooperation expert queries and hierarchically refines the instance masks with encoded features.

transform M experts (each expert includes N queries) via self and cross attention mechanisms, where the MoQ Layer is used to combine the M experts of queries hierarchically. Finally, following previous work, we use a mask head and a matching algorithm to output the CIS prediction.

3.2 Frequency Enhancement Feature Extractor

As mentioned in our Introduction, the camouflage clues are mainly comprised of high-level semantics (*e.g.*, contours and semantics) and low-level statistics (*e.g.*, colors and styles), which can be reflected by the phrase and amplitude components of Fourier spectrum, respectively. As shown in Figure 1, it is believed that enhancing the influence of contours and eliminating the interference of colors would certainly benefit the performance of segmenting the camouflaged instances. Thus, to explore the camouflaged clues, we design FEFE (Frequency Enhancement Feature Extractor) to model the colors and contours of camouflaged objects, where the phrase and amplitude components are applied to identify the camouflaged semantics from surroundings in FEFE. Specifically, suppose H and W are the height and width of the input, and the Fourier transformation $\mathcal{F}(x)$ performed on each channel with a given camouflaged image $x \in \mathcal{R}^{3\times H\times W}$ can be denoted as:

$$\mathcal{F}(x) = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1} x[i,j]e^{-J2\pi(\frac{i}{H}u+\frac{j}{W}v)} = \mathcal{A}(x)e^{J\mathcal{P}(x)}, \tag{1}$$

where J represents the imaginary unit, $\mathcal{A}(x)$ (modeling colors) and $\mathcal{P}(x)$ (modeling contours) are the amplitude and phrase components.

Then, we can get multi-scale image features $\mathbf{F}^k \in \mathcal{R}^{H\times W\times C}, k \in \{2,3,4,5\}$ extracted from a backbone network with the origin image x. Besides, we feed $\mathcal{A}(x)$ into a lightweight CNN and a 1×1 convolution to obtain the global camouflaged color information $\mathbf{F}_{color} \in \mathcal{R}^C$, and eliminate its interference via Color Remove Module (CRM). While for the phrase component $\mathcal{P}(x)$, due to that $\mathcal{P}(x)$ includes some information on contours and textures, extracting multi-scale features of $\mathcal{P}(x)$ can present unique advantages to mining camouflaged clues. Thus, we feed $\mathcal{P}(x)$ into the backbone and obtain hierarchical features $\mathbf{F}^k_{contour}, k \in \{2,3,4,5\}$ to explore the effects of contours as much as possible by the Contour Enhancement Module (CEM). Formally, the process of FEFE, including CRM and CEM for each scale feature $\mathbf{F} \in \{\mathbf{F}^2, \mathbf{F}^3, \mathbf{F}^4, \mathbf{F}^5\}$, can be expressed as:

$$\begin{aligned}
&\mathbf{F}_{enhance} = \lambda\mathbf{F}\odot\mathbf{M}_{color} + (1-\lambda)\mathbf{F}\odot\mathbf{M}_{contour},\\
&\mathbf{M}_{color} = \delta\,\mathrm{Conv}(\mathrm{avg\_c}((\mathbf{F}-\mathbf{F}_{color})^2)),\\
&\mathbf{M}_{contour} = \delta\,\mathrm{Conv}\left(\delta(\mathrm{MLP}(\mathrm{avg\_s}(\mathbf{F}_{contour})))\odot\mathbf{F}\right),
\end{aligned} \tag{2}$$

where avg_c and avg_s indicate average pooling along spatial and channel axis, and δ is an activation function. CRM and CEM are designd to generate $\mathbf{M}_{color}$ and $\mathbf{M}_{contour}$, respectively. With the above module, we can get multi-scale refined features $\mathbf{F}^k_{enhance}, k \in \{2,3,4,5\}$. Further, to acquire more fine-grained features for more accurate segmentation, we fuse $\mathbf{F}^k_{enhance}, k \in \{2,3,4,5\}$ by feeding these features into the pixel decoder based on FPN [Lin *et al.*, 2017] architecture, which is used to gradually upsample low-resolution features from the output of the FEFE to generate high-resolution per-pixel embeddings $\mathcal{X} \in \mathcal{R}^{\frac{H}{4}\times\frac{W}{4}\times C}$.

3.3 Mixture-of-Queries Decoder (MoQ Decoder)

In order to capture camouflaged instances, the popular transformer-based architecture like MaskFormer [Cheng *et al.*, 2021] and Mask2Former [Cheng *et al.*, 2022], proposes

Methods	COD10K-Test			NC4K-Test			Params(M)
	AP	AP$_{50}$	AP$_{75}$	AP	AP$_{50}$	AP$_{75}$	
Mask R-CNN [He *et al.*, 2017]	25.0	55.5	20.4	27.7	58.6	22.7	43.9
MS R-CNN [Huang *et al.*, 2019]	30.1	57.2	28.7	31.0	58.7	29.4	60.0
Cascade R-CNN [Cai and Vasconcelos, 2019]	25.3	56.1	21.3	29.5	60.8	24.8	71.7
HTC [Chen *et al.*, 2019]	28.1	56.3	25.1	29.8	59.0	26.6	76.9
BlendMask [Chen *et al.*, 2020]	28.2	56.4	25.2	27.7	56.7	24.2	35.8
Mask Transfiner [Ke *et al.*, 2022]	28.7	56.3	26.4	29.4	56.7	27.2	44.3
YOLACT [Bolya *et al.*, 2019]	24.3	53.3	19.7	32.1	65.3	27.9	-
CondInst [Tian *et al.*, 2020]	30.6	63.6	26.1	33.4	67.4	29.4	34.1
QueryInst [Fang *et al.*, 2021]	28.5	60.1	23.1	33.0	66.7	29.4	-
SOTR [Guo *et al.*, 2021]	27.9	58.7	24.1	29.3	61.0	25.6	63.1
SOLOv2 [Wang *et al.*, 2020b]	32.5	63.2	29.9	34.4	65.9	31.9	46.2
MaskFormer [Cheng *et al.*, 2021]	38.2	65.1	37.9	44.6	71.9	45.8	45.0
Mask2Former [Cheng *et al.*, 2022]	39.4	67.7	38.5	45.8	73.6	47.5	43.9
OSFormer [Pei *et al.*, 2022]	41.0	71.1	40.8	42.5	72.5	42.3	46.6
DCNet [Luo *et al.*, 2023]	45.3	70.7	47.5	52.8	77.1	56.5	53.4
UQFormer [Dong *et al.*, 2023]	45.2	71.6	46.6	47.2	74.2	49.2	37.5
CamoFourier [Le *et al.*, 2023]	43.5	**74.8**	42.7	44.9	75.6	44.3	-
MSPNet [Li *et al.*, 2024]	39.7	69.8	39.8	41.8	71.8	42.3	48.1
AQS [Dong *et al.*, 2024]	44.8	72.0	46.4	48.1	74.3	50.4	34.4
Ours	**48.0**	73.0	**51.8**	**54.7**	**78.5**	**59.0**	61.6

Table 1: Performance Comparison of Various Methods. The best results are in **bold**.

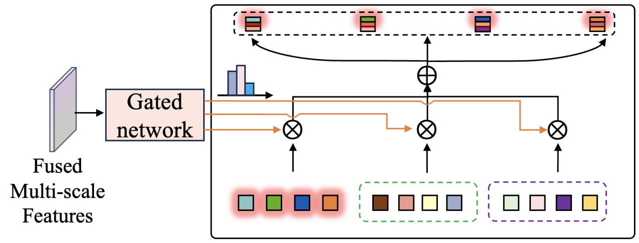


Figure 4: Illustration of the proposed Mixture-of-Queries Layer with a gated network to output the weights of each expert. The output is the weighted mixture of each input experts.

a set of queries to identify whether each pixel belongs to a camouflaged instance. Meanwhile, as discussed in our Introduction, humans may segment camouflaged instances by gradually searching and multi-person collaboration. Inspired by the discussion, we propose a Mixture-of-Queries Decoder (MoQ Decoder) for hierarchically segmenting camouflaged instances.

Mixture-of-Queries Mechanism: Different from the standard Transformer decoder architecture, in each layer, we introduce a Mixture-of-Quries Layer (MoQ Layer) after the original decoder layer, and initialize M experts E_i, $i \in [1, M]$, where each expert contains N queries q_i, $i \in [0, N-1]$, $q_i \in \mathcal{R}^d$. Each query is responsible for an object candidate. So, we have $E_i = \{q_0, q_1, \cdots, q_{N-1}\}$, $q_i \in \mathcal{R}^d$. Further, the detailed architecture of the designed MoQ Layer is illustrated in Figure 4, where the gated network G outputs a sparse $(M+1)$-dimensional vector $G(x) \in \mathcal{R}^{M+1}$ to indicate the weights of each expert. Therefore, given the input of each decoder layer E and the M initialized experts $[E_1, E_2, \cdots, E_M]$, the output y of the MoQ Layer can be written as follows:

$$y = \mathbf{E} \cdot \mathrm{softmax}(G(\mathcal{X})), \tag{3}$$

where $\mathcal{X}$ is the output of pixel decoder, and $G(\mathcal{X})$ indicates the output of the gated network and $\mathbf{E} = [E', E_1, E_2, \cdots, E_M]$. E' is the output of the standard transformer decoder layer fed with E. Besides, the forward process of each MoQ Decoder Layer (including a original decoder layer and a MoQ layer) can be formulated as:

$$\begin{aligned}
&Q = W^Q \cdot E, \quad K = W^K \cdot \mathcal{X}, \quad V = W^V \cdot \mathcal{X}, \\
&E' = \mathrm{LN}\left(E + \mathrm{crossattention}(Q, K, V)\right), \\
&E' = [E', E_1, E_2, \cdots, E_M] \cdot \mathrm{softmax}(G(\mathcal{X})), \\
&O = \mathrm{LN}(E' + \mathrm{MLP}\left(E'\right)),
\end{aligned} \tag{4}$$

where LN is layer normalization and MLP denotes the multi-layer perception network. During the training process, to provide deep supervision, we follow Mask2Former to adopt auxiliary losses with additional mask prediction heads and Hungarian match loss after each MoQ Layer.

3.4 Objective Function

As shown in Figure 3, with the fused feature $\mathcal{X} \in \mathcal{R}^{\frac{H}{4} \times \frac{W}{4} \times C}$ output by Pixel Decoder and the instance candidates $\hat{E} \in \mathcal{R}^{N \times C}$ generated by the MoQ Decoder, we can finally obtain the segmentation map, which can be formulated as:

$$\mathrm{Mask} = \mathcal{X} \times \hat{E}. \tag{5}$$

To train the whole network, following DETR [Carion *et al.*, 2020], we adopt a Hungarian matching algorithm to match a ground truth label with each predicted segment instance. If no suitable label exists, a special label ("no object") is assigned. Therefore, including the instances and mask supervision, the objective function contains three terms: Cross-entropy Loss $\mathcal{L}_{CE}$ for the instance score, Focal Loss $\mathcal{L}_{focal}$ and Dice Loss $\mathcal{L}_{dice}$ for the mask predictions after each MoQ Layer, written as:

$$\mathcal{L}_{total} = \sum_{l=0}^{L} \mathcal{L}_{CE} + \alpha \cdot \mathcal{L}_{focal} + \beta \cdot \mathcal{L}_{dice}, \tag{6}$$

where L means the amount of decoder layers. By default, we set $\alpha = 20$ and $\beta = 1$.

FEFE	MoQ	COD10K-Test			NC4K-Test		
		AP	AP$_{50}$	AP$_{75}$	AP	AP$_{50}$	AP$_{75}$
✓		45.8	71.3	49.3	53.2	77.8	57.3
	✓	47.1	72.8	50.6	53.9	78.0	58.1
✓	✓	48.0	73.0	51.8	54.7	78.5	59.0

Table 2: Performance Comparison of Proposed Modules. We perform an ablation study on COD10K and NC4K to validate our proposed modules' effectiveness. "FEFE" and "MoQ" represent Frequency Enhancement Feature Extractor and Mixture-of-Queries Decoder, respectively.

Method	Backbone	COD10K	NC4K
OSFormer	ResNet-50	41.0	42.5
DCNet		45.3	52.8
Ours		48.0	54.7
OSFormer	ResNet-101	42.0	44.4
DCNet		46.8	53.5
Ours		48.6	55.4
OSFormer	Swin-Tiny	47.7	50.2
DCNet		50.3	56.3
Ours		51.4	58.1
OSFormer	Swin-Small	52.1	56.7
DCNet		52.3	58.4
Ours		53.2	59.2

Table 3: Performance Comparison of Various Backbones. We evaluate multiple methods' performance with various backbones on COD10K and NC4K.

4 Experiments

4.1 Experimental Setups

Following the mainstream works of CIS [Dong *et al.*, 2023; Luo *et al.*, 2023], we evaluate our method in two datasets: COD10K and NC4K. COD10K includes 3040 training images and 2026 testing images, while NC4K contains 4121 test images for evaluating the generalization of proposed models. To provide a fair comparison, we train models in the training set in COD10K, and meanwhile test models in both test sets of COD10K and NC4K, which is a standard setting proposed in previous works [Luo *et al.*, 2023; Pei *et al.*, 2022]. In order to comprehensively evaluate the models, We use AP_{50}, AP_{75}, and AP scores as evaluation metrics to quantify the performance of our method and baselines [Luo *et al.*, 2023; Dong *et al.*, 2023; Pei *et al.*, 2022]. Besides, it is free to check the supplementary material for more details of the method and experiments.

4.2 Comparison with State-of-the-Art Methods

The CIS task is a relatively novel task that has been proposed in recent years, and only a few previous works are involved in this task, such as OSFormer [Pei *et al.*, 2022], DCNet [Luo *et al.*, 2023], and UQFormer [Dong *et al.*, 2023]. Consequently, we also adopt several popular generic instance segmentation methods as baselines on the CIS task for a more comprehensive test. And for a fair comparison, the backbone of these methods is configured as ResNet-50. The performance comparison results are shown in Section 3.2. It is easy to observe that our proposed model can consistently outperform the state-of-the-art methods by a large margin on both COD10K and NC4K test sets.

Decoder Layers	COD10K-Test			NC4K-Test			Params(M)
	AP	AP$_{50}$	AP$_{75}$	AP	AP$_{50}$	AP$_{75}$	
2	46.5	72.0	50.2	53.5	78.1	57.7	54.7
4	47.0	72.3	51.0	53.6	78.3	57.8	57.9
6	**48.0**	**73.0**	**51.8**	**54.7**	**78.5**	**59.0**	61.6
8	47.5	72.4	51.3	53.7	78.4	57.9	65.4
10	46.9	71.5	50.0	53.4	78.0	57.1	68.4
12	47.2	72.0	50.2	53.8	78.2	58.1	71.2

Table 4: Performance Comparison of Various Number of Decoder Layers. We apply various numbers of decoder layers, and the performance is shown as follows. The best results are in bold.

(1) Results on COD10K. As shown in Section 3.2, we compare our proposed model with 6 CIS models (*i.e.*, OSFormer [Pei *et al.*, 2022], DCNet [Luo *et al.*, 2023], *etc.*), 13 generic instance segmentation models. Our model can achieve 51.8% in AP_{75}, which outperforms the second best method DCNet [Luo *et al.*, 2023] by 4.3% in AP_{75}. In AP, our model also gets a performance improvement of 2.6%. Notice that our method does not achieve the highest value in AP_{50}, instead of a comparable performance of 73.0% in AP_{50}. These results indicate that our method can acquire more accurate segmentation masks of camouflaged objects.

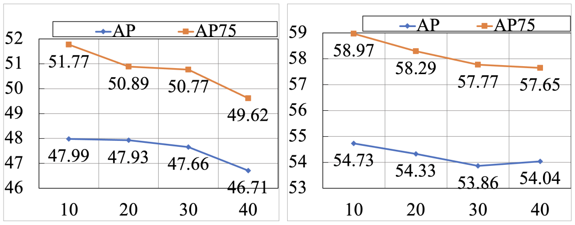
(2) Results on NC4K. Likewise, we evaluate these methods on NC4K dataset, and the results on this test set reflect the generalization ability of these models. Our model yields 59.0% in AP_{75}, while the previous best method DCNet is 56.5%, which demonstrates that our method gets an obvious gain of 2.5% in AP_{75}, suggesting a great generalization ability of our model as well. In AP, our model achieves the highest performance metrics of 54.7%, surpassing the second best method (DCNet) by 1.9%. Besides, our model also obtains a 1.35% improvement in AP_{50}. The overall metrics of various AP values reflect our method's obvious superiority over other baselines.

4.3 Ablation Studies and Visualizations

To look deeper into our proposed method, in this section, we present a series of ablation studies to demonstrate the effectiveness of each proposed module.

Effectiveness of proposed modules. To explore the effectiveness of the proposed FEFE and MoQ Decoder, we validate the importance of each component by removing them one at a time. As shown in Section 3.4, the performance without MoQ Decoder drops by 2.2 % in AP, 1.7% in $AP_{AP_{50}}$ and 2.5% in $AP_{AP_{75}}$ on COD10K-Test. On NC4K-Test, the metrics of AP, AP_{50} and AP_{75} are also reduced by 1.5%, 0.7% and 1.7%, respectively. Similarly, if the components of FEFE are ablated, there is a drop in segmentation performance as well. For example, on COD10K-Test, the performance just achieves 47.1% in AP, 72.8% in AP_{50} and 50.6% in AP_{75}, which are consistently lower than that without any modules ablated (as shown in the last row of Section 3.4). The reduced performance demonstrates that these two proposed modules can capture clues of camouflaged instances and provide accurate segmentation. With both modules, our method can lead to huge performance gains in evaluation metrics.

Various backbones. To further explore the potential of our model, we equip it with different feature extractor backbones, such as ResNet-50 [He *et al.*, 2016], ResNet-101 [He



(a) (b)

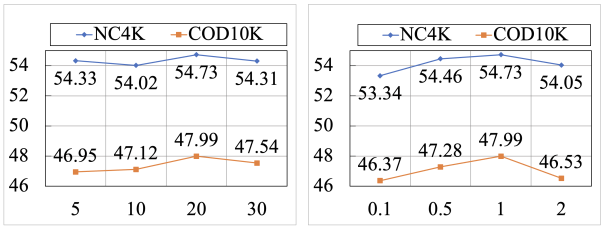
Figure 5: Performance Comparison of Various Numbers of Queries in Each Expert. AP and AP$_{75}$ of our MoQT with various numbers of queries on COD10K-Test (a) and NC4K-Test (b) are shown.

et al., 2016], SwinTransformer-Tiny (Swin-Tiny) [Liu *et al.*, 2021], and SwinTransformer-Small (Swin-Small) [Liu *et al.*, 2021]. For a fair comparison with baselines, all these models are pretrained on ImageNet-1k. The results are presented in Section 3.4. With the same backbone, our method shows the best performance among compared baselines, which indicates our method outperforms the state-of-the-art methods. For example, when ResNet-101 is the backbone, the metrics of AP of our method are 48.6% and 55.4% on COD10K and NC4K datasets, respectively, while the second best method just reaches 46.8% and 53.5%. With a larger backbone, the results also prove that our method has the potential for further improvement.

Analysis of the number of decoder layers. We apply auxiliary losses after each decoding layer, as formulated in Equation (6). Hence, the number of decoder layers L is important for the segmentation performance. As presented in Section 3.4, we vary the number of decoder layers, picked from the set $\{2, 4, 6, 8, 10, 12\}$. We find that the overall performance of the model improves with the increase of L. And when $L = 6$, the model can get the best performance. There is no additional performance gain when the L continues to increase, which may be caused by limited data to train the model for further improvement.

Ablation on the number of queries. Object queries are essential in the transformer architecture for prediction. Therefore, we study the performance with different numbers of queries in each expert group. As shown in Figure 5, we change the number of queries from 10 to 40 and evaluate the performance metrics of AP and AP_{75} in both COD10K and NC4K test sets. In fact, the number of queries in each group should be larger than the actual count of objects to avoid instance fusion, which is determined by the dataset distribution. Moreover, it can be seen that when the number is set as 10, our model obtains the best performance on both datasets. For example, when the number is 10, AP and AP_{75} is 47.99% and 51.77% on COD10K, respectively. Meanwhile, AP and AP_{75} reach 54.73% and 58.97%.

Impacts about Hyper-parameters. We study the impacts of the Hyper-parameters α and β in Equation (6). On both COD10K and NC4K datasets, when $\alpha = 20$, the best performance is achieved, proved by the metrics of $AP = 47.99\%$ and 54.73%, respectively, shown in Figure 6(a). Therefore, we choose $\alpha = 20$ in our method by default. For the hyper-parameter β, we change the value of β from 0.1 to 2, and the results as presented in Figure 6 (b). It can be seen that the model gets the best performance when $\beta = 1$. Therefore, to get the best performance, we set α as 20, and β as 1.



(a) (b)

Figure 6: Ablation Studies on Hyper-parameters of α (a) and β (b), presented in Equation (6).

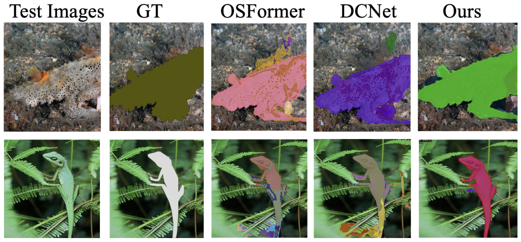


Figure 7: Visualizations of Various Methods. Different colored masks indicate different instances.

Visualization Results. To comprehensively evaluate our method, we also present some qualitative analysis, as shown in Figure 7. We visualize the segmentation masks of various methods, including OSFormer [Pei *et al.*, 2022], DCNet [Luo *et al.*, 2023], and our method, to demonstrate the performance wtih qualitative results. It can be seen that our method performs better than previous methods, which can be proved by the clear boundaries and accurate masks of our method (shown in the last row of Figure 7). In short, our method not only improves the evaluation metrics on two datasets but also gains in visual results of segmentation masks.

5 Conclusion

In this paper, we propose a novel Mixture-of-Queries Transformer (MoQT) for camouflaged instance segmentation. MoQT applies a Frequency Enhancement Feature Extractor for feature extraction in the frequency domain, with the assistance of a contour enhancement module and a color removal module. Besides, a Mixture-of-Queries Decoder uses multiple expert groups of queries as candidates and shares semantic information with transformer encoder features. Multi-scale features enable MoQT to refine prediction hierarchically and get fine-grained instance masks with collaboration of multiple groups of queries. Compared with plenty of state-of-the-art baselines, our proposed MoQT shows outstanding performance on two benchmark datasets, demonstrating the proposed method's effectiveness.

Acknowledgments

This work was supported by Ant Group Postdoctoral Programme. The authors from Ant Group are supported by the Leading Innovative and Entrepreneur Team Introduction Program of Hangzhou (Grant No.TD2022005).

References

[Bolya *et al.*, 2019] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9157–9166, 2019. 1, 3, 5

[Cai and Vasconcelos, 2019] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: High quality object detection and instance segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 43(5):1483–1498, 2019. 1, 3, 5

[Carion *et al.*, 2020] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 2, 5

[Chen *et al.*, 2019] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4974–4983, 2019. 1, 3, 5

[Chen *et al.*, 2020] Hao Chen, Kunyang Sun, Zhi Tian, Chunhua Shen, Yongming Huang, and Youliang Yan. Blendmask: Top-down meets bottom-up for instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8573–8581, 2020. 5

[Cheng *et al.*, 2021] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *Advances in Neural Information Processing Systems*, 2021. 3, 4, 5

[Cheng *et al.*, 2022] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2022. 3, 4, 5

[Dong *et al.*, 2023] Bo Dong, Jialun Pei, Rongrong Gao, Tian-Zhu Xiang, Shuo Wang, and Huan Xiong. A unified query-based paradigm for camouflaged instance segmentation, 2023. 1, 2, 3, 5, 6

[Dong *et al.*, 2024] Bo Dong, Pichao Wang, Hao Luo, and Fan Wang. Adaptive query selection for camouflaged instance segmentation. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 6598–6606, 2024. 5

[Fan *et al.*, 2020a] Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Camouflaged object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2777–2787, 2020. 3

[Fan *et al.*, 2020b] Deng-Ping Fan, Ge-Peng Ji, Tao Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao. Pranet: Parallel reverse attention network for polyp segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 263–273. Springer, 2020. 1

[Fan *et al.*, 2020c] Deng-Ping Fan, Tao Zhou, Ge-Peng Ji, Yi Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao. Inf-net: Automatic covid-19 lung infection segmentation from ct images. *IEEE transactions on medical imaging*, 39(8):2626–2637, 2020. 1

[Fang *et al.*, 2021] Yuxin Fang, Shusheng Yang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Instances as queries. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6910–6919, 2021. 5

[Fu *et al.*, 2019] Huazhu Fu, Fei Li, José Ignacio Orlando, Hrvoje Bogunović, Xu Sun, Jingan Liao, Yanwu Xu, Shaochong Zhang, and Xiulan Zhang. Refuge: Retinal fundus glaucoma challenge, 2019. 2

[Guo *et al.*, 2021] Ruohao Guo, Dantong Niu, Liao Qu, and Zhenbo Li. Sotr: Segmenting objects with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7157–7166, 2021. 5

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6, 7

[He *et al.*, 2017] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 1, 3, 5

[Huang *et al.*, 2019] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring r-cnn. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6409–6418, 2019. 5

[Ke *et al.*, 2022] Lei Ke, Martin Danelljan, Xia Li, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Mask transfiner for high-quality instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4412–4421, 2022. 5

[Le *et al.*, 2019] Trung-Nghia Le, Tam V Nguyen, Zhongliang Nie, Minh-Triet Tran, and Akihiro Sugimoto. Anabranch network for camouflaged object segmentation. *Computer vision and image understanding*, 184:45–56, 2019. 3

[Le *et al.*, 2021] Trung-Nghia Le, Yubo Cao, Tan-Cong Nguyen, Minh-Quan Le, Khanh-Duy Nguyen, Thanh-Toan Do, Minh-Triet Tran, and Tam V Nguyen. Camouflaged instance segmentation in-the-wild: Dataset, method, and benchmark suite. *IEEE Transactions on Image Processing*, 31:287–300, 2021. 1, 3

[Le *et al.*, 2023] Minh-Quan Le, Minh-Triet Tran, Trung-Nghia Le, Tam V. Nguyen, and Thanh-Toan Do. Unveiling camouflage: A learnable fourier-based augmentation for camouflaged object detection and instance segmentation, 2023. 1, 5

[Li *et al.*, 2024] Chen Li, Ge Jiao, Guowen Yue, Rong He, and Jiayu Huang. Multi-scale pooling learning for camouflaged instance segmentation. *Applied Intelligence*, pages 1–15, 2024. 1, 5

[Lin *et al.*, 2017] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, 2017. 3, 4

[Liu *et al.*, 2021] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 7

[Luo *et al.*, 2023] Naisong Luo, Yuwen Pan, Rui Sun, Tianzhu Zhang, Zhiwei Xiong, and Feng Wu. Camouflaged instance segmentation via explicit de-camouflaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17918–17927, 2023. 1, 2, 3, 5, 6, 7

[Mei *et al.*, 2021] Haiyang Mei, Ge-Peng Ji, Ziqi Wei, Xin Yang, Xiaopeng Wei, and Deng-Ping Fan. Camouflaged object segmentation with distraction mining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8772–8781, 2021. 3

[Pan *et al.*, 2011] Yuxin Pan, Yiwang Chen, Qiang Fu, Ping Zhang, Xin Xu, et al. Study on the camouflaged target detection method based on 3d convexity. *Modern Applied Science*, 5(4):152, 2011. 3

[Pang *et al.*, 2022] Youwei Pang, Xiaoqi Zhao, Tian-Zhu Xiang, Lihe Zhang, and Huchuan Lu. Zoom in and out: A mixed-scale triplet network for camouflaged object detection. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 2160–2170, 2022. 3

[Pei *et al.*, 2022] Jialun Pei, Tianyang Cheng, Deng-Ping Fan, He Tang, Chuanbo Chen, and Luc Van Gool. Osformer: One-stage camouflaged instance segmentation with transformers. In *European conference on computer vision*. Springer, 2022. 1, 2, 3, 5, 6, 7

[Ren *et al.*, 2015] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 1, 3

[Ren *et al.*, 2021] Jingjing Ren, Xiaowei Hu, Lei Zhu, Xuemiao Xu, Yangyang Xu, Weiming Wang, Zijun Deng, and Pheng-Ann Heng. Deep texture-aware features for camouflaged object detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(3):1157–1167, 2021. 3

[Sengottuvelan *et al.*, 2008] P Sengottuvelan, Amitabh Wahi, and A Shanmugam. Performance of decamouflaging through exploratory image analysis. In *2008 First International Conference on Emerging Trends in Engineering and Technology*, pages 6–10. IEEE, 2008. 3

[Tian *et al.*, 2020] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 282–298. Springer, 2020. 5

[Turić *et al.*, 2010] Hrvoje Turić, Hrvoje Dujmić, and Vladan Papić. Two-stage segmentation of aerial images for search and rescue. *Information Technology and Control*, 39(2), 2010. 1

[Wang *et al.*, 2020a] Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, and Lei Li. Solo: Segmenting objects by locations. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 649–665. Springer, 2020. 1, 3

[Wang *et al.*, 2020b] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. Solov2: Dynamic and fast instance segmentation. *Advances in Neural information processing systems*, 33:17721–17732, 2020. 1, 3, 5

[Xu and Feng, 2023] Nanqing Xu and Weiwei Feng. Metafake: Few-shot face forgery detection with meta learning. In *Proceedings of the 2023 ACM Workshop on Information Hiding and Multimedia Security*, pages 151–156, 2023. 1

[Xu *et al.*, 2021] Qinwei Xu, Ruipeng Zhang, Ya Zhang, Yanfeng Wang, and Qi Tian. A fourier-based framework for domain generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14383–14392, 2021. 3

[Yang *et al.*, 2021] Fan Yang, Qiang Zhai, Xin Li, Rui Huang, Ao Luo, Hong Cheng, and Deng-Ping Fan. Uncertainty-guided transformer reasoning for camouflaged object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4146–4155, 2021. 3

[Zhang *et al.*, 2024] Yi Zhang, Changtao Miao, Man Luo, Jianshu Li, Wenzhong Deng, Weibin Yao, Zhe Li, Bingyu Hu, Weiwei Feng, Tao Gong, et al. Mfms: Learning modality-fused and modality-specific features for deepfake detection and localization tasks. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 11365–11369, 2024. 1

[Zhong *et al.*, 2022] Yijie Zhong, Bo Li, Lv Tang, Senyun Kuang, Shuang Wu, and Shouhong Ding. Detecting camouflaged object in frequency domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4504–4513, 2022. 3