Shift the Lens: Environment-Aware Unsupervised Camouflaged Object Detection

Ji Du[1,2], Fangwei Hao[1], Mingyang Yu[1], Desheng Kong[1]
Jiesheng Wu[1], Bin Wang[1], Jing Xu[1,*], Ping Li[2,3*]
[1]College of Artificial Intelligence, Nankai University, China
[2]Department of Computing, The Hong Kong Polytechnic University, Hong Kong
[3]School of Design, The Hong Kong Polytechnic University, Hong Kong

Abstract

*Camouflaged Object Detection (COD) seeks to distinguish objects from their highly similar backgrounds. Existing work has essentially focused on isolating camouflaged objects from the environment, demonstrating ever-improving performance but at the cost of extensive annotations and complex optimizations. In this paper, we diverge from this paradigm and shift the lens to isolating the salient environment from the camouflaged object. We introduce **EASE**, an **E**nvironment-**A**ware un**S**up**E**rvised COD framework that identifies the environment by referencing an environment prototype library and detects camouflaged objects by inverting the retrieved environmental features. Specifically, our approach (DiffPro) uses large multimodal models, diffusion models, and vision-foundation models to construct the environment prototype library. To retrieve environments from the library and refrain from confusing foreground and background, we incorporate three retrieval schemes: Kernel Density Estimation-based Adaptive Threshold (KDE-AT), Global-to-Local pixel-level retrieval (G2L), and Self-Retrieval (SR). Our experiments demonstrate significant improvements over current unsupervised methods, with EASE achieving an average gain of over 10% on the COD10K dataset. When integrated with SAM, EASE surpasses prompt-based segmentation approaches and performs competitively with state-of-the-art fully-supervised methods. Code is available at* https://github.com/xiaohainku/EASE.

1. Introduction

Camouflaged Object Detection (COD) [15] focuses on identifying objects that blend in with their surroundings, making them visually challenging to detect, as indicated by the left part of Fig. 1. Existing COD methods focus on **identifying hidden cues** within the environment to detect the concealed. Fully-supervised learning approaches in COD tackle the challenge by designing architectures that capture fine camouflage details in complex settings, using bottom-up/top-down [15, 16, 53] or multi-stream [76, 82] structures to leverage dense annotations. In a similar vein, semi-supervised learning [79] seeks to maximize the learning of camouflage features with minimal annotation. To further reduce the reliance on extensive labeling, weakly-supervised learning relies on sparse annotations, like points [8], scribbles [26], boxes [79] or pseudo-labels [23], to guide models in distinguishing camouflaged objects from their backgrounds.

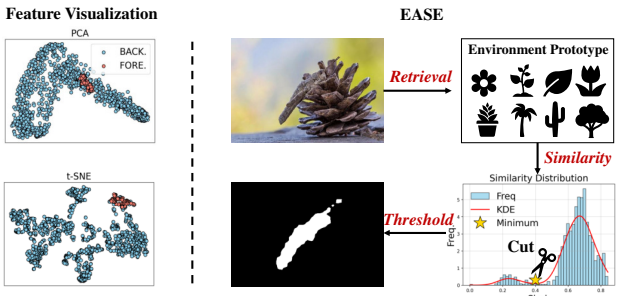

Figure 1. (Left) Feature visualization from DINOv2 [55] shows that objects blend seamlessly with their surroundings, making it extremely challenging to distinguish foreground solely based on feature similarity due to a lack of clear boundaries. (Right) EASE offers a solution by shifting focus from direct feature comparison. Instead of relying on inter-feature similarity alone, EASE identifies background regions by comparing features against predefined environment prototypes. Camouflaged objects are then isolated by inverting this background detection.

Another line of research revolves around prompt-based segmentation, devoid of reliance on annotations. A typical paradigm involves using Large Multimodal Mod-

*Corresponding authors: Jing Xu (xujing@nankai.edu.cn) and Ping Li (p.li@polyu.edu.hk).

els (LMMs) to identify categories [28, 29] or bounding boxes [34, 70] of camouflaged objects. These outputs then serve as prompts within SAM [39] to achieve accurate segmentation.

The above approaches focus on improving the detection of non-salient camouflaged objects from the salient environment—similar to discerning a glass ball within a bottle of water. Supervised methods have shown notable progress, yet striking the ideal balance between performance and annotation expense warrants further exploration. For prompt-based segmentation, the hallucination [31, 43] originating from LMMs leaves a large performance gap with supervised learning ascribed to the high similarity between the camouflaged objects and the environment.

If finding the glass ball is so complicated, we would like to ask: *why don't we just pour off the water?*

When humans cannot immediately locate camouflaged objects, they often rely on their prior knowledge about the environment, systematically filtering out extraneous details until the target emerges. Inspired by this cognitive strategy, we introduce EASE. As illustrated in Fig. 1, rather than directly targeting camouflaged objects, EASE emphasizes the surrounding environment, retrieving from a specially crafted environment prototype library. It then identifies the camouflaged object by inverting the retrieved details. To unleash the potential of EASE, we consider the key to be ❶ the prototype library and ❷ retrieval schemes.

Instead of relying on the laborious manual collection of environmental prototypes from online sources or datasets—which may also risk revealing camouflaged object information—our proposed DiffPro utilizes foundation models to automatically build an environment-focused prototype library. First, we employ Large Multimodal Models (LMMs) such as LLaVA-1.5 [50] to define a set of environment categories. Recognizing broad environmental contexts is a simpler task for LMMs than identifying camouflaged objects. To create diverse prototypes that exclude camouflaged objects, we take advantage of the powerful conditional generation abilities of diffusion models like Stable Diffusion [61] to efficiently generate a range of environmental images. Finally, self-supervised vision foundation models, such as DINOv2 [55], are used to extract both global (average-pooled) and local (pixel-level) feature vectors from these images.

The prototype library enables us to assess whether a target belongs to the environment or is a camouflaged object by retrieving the top-N prototypes and setting a similarity threshold based on their average. However, fixed thresholds are unsuitable given the variability across camouflage scenarios, and they also add extra hyperparameters, complicating the model. To resolve this, we introduce KDE-AT, which uses kernel density estimation to model the similarity distribution, allowing the minimum value to function as a flexible, adaptive threshold that reduces complexity and enhances detection reliability.

For effective and efficient retrieval, we propose Global-to-Local retrieval (G2L). G2L first identifies the top-K images using global prototypes and then performs pixel-level retrieval on these selected images. This two-step approach sharpens retrieval precision to a pixel-by-pixel level while reducing the number of prototypes needed, eliminating redundant retrievals, and enhancing overall efficiency.

To address the distribution gap between camouflaged and generated images that may impact retrieval accuracy, we introduce a Self-Retrieval (SR) approach. Unlike G2L that relies on external prototype libraries, SR uses its own features as the prototype reference. SR includes two prototypes—foreground and background—derived through mask-averaged pooling over target features, guided by G2L masks. The similarity of each pixel is determined by the contrast between these prototypes. By incorporating inter-feature similarity, SR effectively refines retrieval accuracy, reducing false positives and negatives.

Extensive experiments conducted across multiple datasets showcase that our EASE substantially outperforms the SOTA unsupervised methods. Our contributions are summarized as follows:

- We introduce a new perspective to understand COD, i.e., stripping the environment from the camouflaged rather than finding the camouflaged from the environment. The proposed approach EASE segments camouflaged objects through retrieving from the prototype library in a training-free manner, substantially reducing the expense of annotation and training.
- To construct the environment prototype library, we propose DiffPro, an automatic and efficient pipeline that mitigates issues where camouflaged objects in the environment library could obscure retrieval decisions. DiffPro addresses this by leveraging multiple foundation models in a collaborative framework.
- We devise multifaceted retrieval schemes (including KDE-AT, G2L, and SR) to effectively and efficiently isolate environments from camouflaged objects. Extensive experiments speak to the efficacy of our method.

2. Related Work

2.1. Camouflaged Object Detection

Camouflaged Object Detection [15, 77] is attracting increasing research attention because of its distinct challenges and real-world applicability. Existing efforts have primarily focused on disentangling camouflaged objects from the environment, and can be categorized into supervision-based [7, 16, 20, 21, 24, 25, 30, 32, 33, 41, 47, 48, 51, 53, 56, 67, 68, 75, 76, 82–84] and prompt-based [28, 29, 70] segmentation depending on whether supervision signals are

required or not. Supervision-based methods can be further divided into fully-supervised and weakly-supervised learning, where fully-supervised learning employs dense annotations as signals while weakly-supervised learning utilizes sparse supervision to ease the labeling burden. Another line of research, prompt-based segmentation, extracts insights on camouflaged objects from LMMs by drawing on category [28, 29] or localization [29, 70]. These insights are then processed to generate prompts compatible with SAM, which then performs the segmentation.

Similar to prompt-based segmentation, our method is also supervision-free and foundation model-dependent. However, instead of searching for camouflaged objects directly from the environment, we eliminate the environment and let the camouflaged objects reveal themselves.

2.2. Unsupervised Object Segmentation

Unsupervised Object Segmentation [1, 54, 71] aims to segment objects without any supervision signals. Current prevailing unsupervised methods revolve around segmenting objects using features from self-supervised learning models [6, 55]. LOST [64] utilizes the activation features of a pre-trained self-supervised visual Transformer, DINO [6], selecting regions with low similarity as seeds and expanding them to identify object boundaries, thereby achieving object localization. TokenCut [73] adopts features as vertices of the graph, cosine similarity between features as the weights of edges, and then employs Normalized Cut (Ncut) [63] to segment the target into foreground and background parts. Based on this, CutLER [72] proposes MaskCut, which can generate multiple masks compared to TokenCut. CuVLER [1] leverages features from multiple DINO and DINOv2 to perform Ncut and derive the final mask by clustering and voting. DiffCut [10] utilizes features from the diffusion model for Ncut instead of DINO.

The most similar philosophy to our approach is FOUND [65], which also discovers objects by looking for the background rather than the foreground. FOUND utilizes the less weighted features in the DINO self-attention maps as seed patches and obtains the coarse background mask by similarity matching. The reversed background masks are further regarded as supervision in the self-supervised refinement stage for fine segmentation. However, the high similarity between camouflaged objects and the environment leads to a remarkable decrease in self-supervised feature differentiation (see Tab. 1). Therefore, effective separation of background and foreground cannot be achieved solely based on the similarity between features (Fig. 1 left). We propose prototype library-based retrieval to address this problem. Our starting point, as illustrated in Fig. 1, is that even though two features are similar, they may not maintain consistent similarity with the third party.

2.3. Retrieval-Augmented Generation

In NLP, Retrieval-Augmented Generation (RAG) [4, 19, 44] aims to compensate for the deficiencies of language models on knowledge-intensive tasks, while illuminating their decisions and updating their world knowledge. This paradigm has recently catalyzed popularity in the vision community and spawned promising applications in a growing number of domains, such as long-tailed recognition [57], video captioning [78], continual learning [58], image retrieval [69] and object detection [38].

The most related to our method is retrieval-based segmentation [3, 36, 74]. These methods generate a series of category prototypes using diffusion models and then employ pre-trained segmentation models [74], unsupervised segmentation methods [36], or superpixel segmentation algorithms [3] to generate optimized mask proposals, which are assigned semantic class through retrieval. For these methods, generating an image of a dog may be accompanied by other undesirable ones, such as grass or trees, which would dilute the prototype semantics. However, our method focuses not on the foreground category but on the background. Our DiffPro generates target-only prototypes and excludes potential interferences. To obtain fine segmentation, we design a series of retrieval schemes instead of using existing segmentation methods.

3. Method

Given an image, unsupervised camouflaged object detection aims to generate a binary mask containing camouflaged objects without any supervision as well as training. Unlike previous approaches [15, 70] that focused on painstakingly searching for camouflaged objects from the environment, our approach focuses on detecting the obvious environment and then inverting it to obtain foreground objects. Building on a series of methods for discovering objects using self-supervised features [64, 65, 72, 73], our approach accomplishes object segmentation through retrieval, utilizing the similarity distribution of features to a third party rather than the similarity between features.

The overview of our proposed EASE is illustrated in Fig. 2. First, we propose DiffPro to craft the environment prototype library leveraging various foundation models (Sec. 3.1). Second, for effective and efficient retrieval, we propose G2L to implement global-to-local retrieval (Sec. 3.2). Third, we propose SR to supplement G2L retrieval by utilizing the similarity between features, where the features themselves serve as the prototype library (Sec. 3.3). Finally, we introduce KDE-AT, which provides adaptive similarity thresholds for each image instead of fixed thresholds via hyperparameter search (Sec. 3.4).

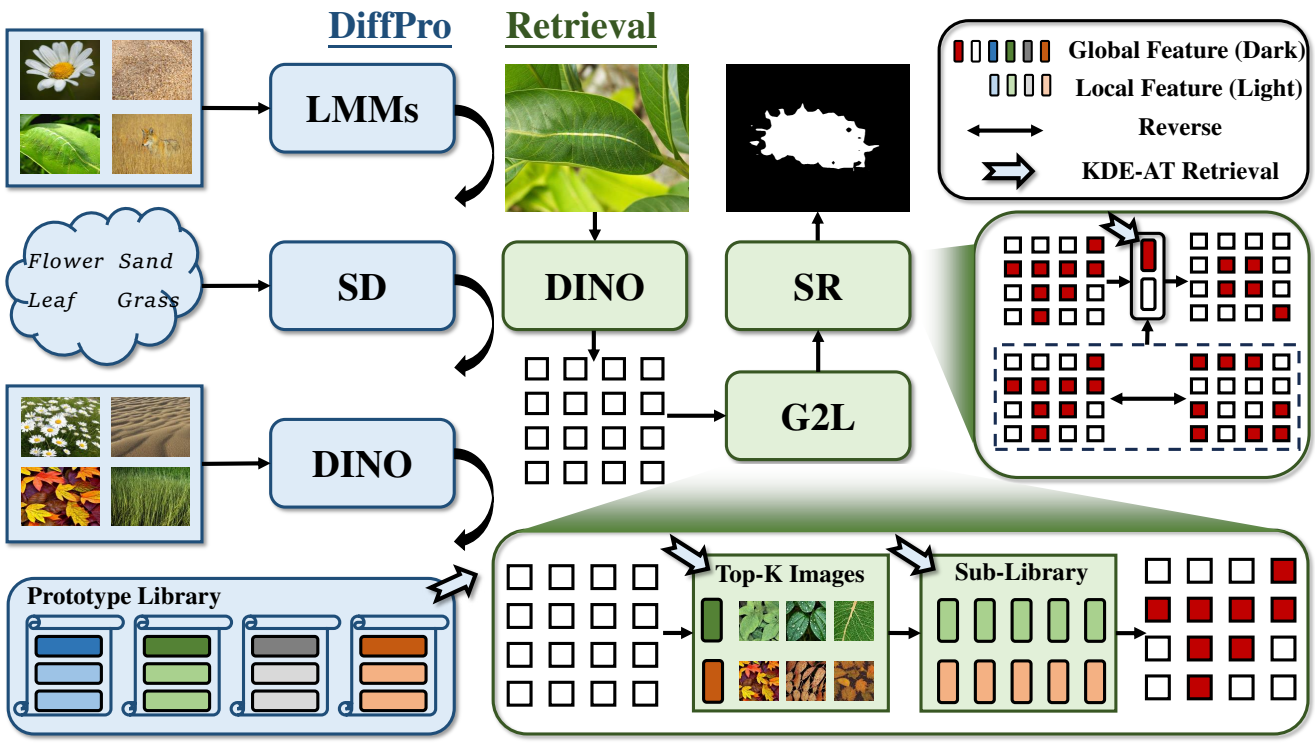

Figure 2. EASE consists of two main components: DiffPro and Retrieval. In DiffPro, LMMs establish an environmental category set based on existing datasets. With these categories, Stable Diffusion generates corresponding environmental images. Using DINO, these images are processed to extract both global and local features, forming a comprehensive environment prototype library. During Retrieval, DINO generates a set of patch embeddings from the target images. In the first stage, G2L retrieval, the top-K images in the prototype library that are most similar to the environment of the target image are retrieved. Next, a more detailed retrieval is conducted on a subset of local features from the top-K images. To enhance retrieval precision and mitigate potential noise, Self-Retrieval (SR) uses coarse masks from the G2L process. These masks are employed to create global average pooling features for both foreground and background regions, which are then retrieved again for refinement. All retrieval thresholds are adaptively set using KDE-AT, ensuring optimal results across varied scenarios.

3.1. Environment Prototype Library

We propose DiffPro (left in Fig. 2) to build a prototype library that contains only environments.

The first step is to get the environment category set. Inspired by the power of LMMs demonstrated on VQA [2, 11, 45, 49], we utilize LMMs to derive the environment set. Although previous works [29, 70] have shown that the imperceptibility of camouflaged objects can cause potential hallucinations in LMMs, our approach enables LMMs to recognize the salient environment rather than the camouflaged object and therefore bypasses the potential for erroneous output. Given the image datasets $\{\mathbf{I_i}\}_{i=1}^{n}$ where n denotes the total number of images, the environment category could be derived by $C_i = \text{LMM}(\mathbf{I_i}, P_c)$. P_c denotes the prompt. We use "*What is the environment of this photo? Answer the question using a single word or phrase*" for it. By removing repeated and irrelevant categories, all C_i constitute the set of environment categories $\{C_j\}_{j=1}^{m}$, where m denotes the total categories.

The second step is to get the environment image set. An intuitive idea is to retrieve images from the Web via the search engine or from an existing dataset such as COCO-Stuff [5]. However, retrieval via the Web takes a significant amount of time, even without labeling the images. In addition, both retrieved images and images in existing datasets suffer from serious object-environment coexistence issues. For example, flowers are often accompanied by bees or butterflies. The presence of objects in the environment library will significantly distort the similarity distribution and affect the separation of the environment.

To address these issues, we turn to the powerful conditional generation capabilities of diffusion models [27, 61] to automatically generate images containing only the environment. For each category C_j in the environment category set $\{C_j\}_{j=1}^{m}$, we leverage Stable Diffusion [61] to generate l images, $\mathbf{G_{i,C_j}} = \text{SD}(C_j, P_s)$, where P_s denotes the prompt. In this paper, we adopt "*a photo of [class]*" for P_s. All generated images make up the environment image set

$\{\mathbf{G}_{\mathbf{i},\mathbf{C_j}}\}_{i=1:l,j=1:m}$.

The third step is to obtain the final environment prototype library. We adopt self-supervised models, DINO [6, 55], as the feature extractor. Given the generated image $\mathbf{G}$ in the environment image set $\{\mathbf{G}_{\mathbf{i},\mathbf{C_j}}\}_{i=1:l,j=1:m}$, we employ DINO to extract the patch embeddings $\mathbf{E}$ from the last layers, $\mathbf{E} = \text{DINO}(\mathbf{G}_{\mathbf{i},\mathbf{C_j}})$, $\mathbf{E} \in \mathbb{R}^{h\times w\times d}$, where $h \times w$ denotes the resolution of the features and d the embedding dimension.

For each generated image, local prototypes could be derived as $\mathbf{P^l_{i,j}} = \mathbf{E_{i,j}} \in \mathbb{R}^d$. By employing global average pooling, we get the global feature as

$$\mathbf{P^g} = \frac{1}{hw}\sum_{i=1}^{h}\sum_{j=1}^{w}\mathbf{P^l_{i,j}} = \frac{1}{hw}\sum_{i=1}^{h}\sum_{j=1}^{w}\mathbf{E_{i,j}}. \tag{1}$$

We denote the final environment prototype library as $\{(\mathbf{P^g}, \mathbf{P^l})\}$.

3.2. Global-to-Local Retrieval

We propose G2L to efficiently retrieve the environment from coarse to fine. As illustrated in Fig. 2 (right), given the test image $\mathbf{T} \in \mathbb{R}^{H\times W\times 3}$, we adopt DINO to extract the patch embeddings $\mathbf{F} \in \mathbb{R}^{h\times w\times d}$. We first perform a global retrieval. For each global feature $\mathbf{P^g} \in \mathbb{R}^d$ in the prototype library $\{(\mathbf{P^g}, \mathbf{P^l})\}$, the cosine similarity between $\mathbf{F_{i,j}} \in \mathbb{R}^d$ and $\mathbf{P^g}$ could be formulated as

$$\mathbf{S_{i,j}} = \frac{\mathbf{F_{i,j}} \cdot \mathbf{P^g}}{\|\mathbf{F_{i,j}}\|\|\mathbf{P^g}\|}. \tag{2}$$

For each $\mathbf{F_{i,j}}$, we retrieve the top-N global features and adopt the average cosine similarity to replace $\mathbf{S_{i,j}}$,

$$\mathbf{S_{i,j}} = \frac{1}{N}\sum_{f=1}^{N}\frac{\mathbf{F_{i,j}} \cdot \mathbf{P^g_f}}{\|\mathbf{F_{i,j}}\|\|\mathbf{P^g_f}\|}. \tag{3}$$

By setting an adaptive threshold (see Sec. 3.4) for $\mathbf{S}$, we could get the coarse mask $\mathbf{M_c} \in \mathbb{R}^{h\times w}$.

We then perform local retrieval on local features $\mathbf{P^l}$ to get the finer mask. However, the number of local features is $h \times w$ times larger than that of global features. The retrieval time will also increase by a corresponding factor compared to the global retrieval. To enhance retrieval efficiency, we focus on local features with high similarity to the target, which are often concentrated within a limited set of images. By selecting local features from the top-K (note that top-K is different from the previously mentioned top-N) similar images, we construct a streamlined sub-library for retrieval. This approach significantly optimizes the retrieval process.

Given the coarse mask $\mathbf{M_c} \in \mathbb{R}^{h\times w}$, we could get the mask average pooling feature for $\mathbf{F}$,

$$\mathbf{F_{map}} = \frac{\sum_{i,j}\mathbf{M_c} \odot \mathbf{F}}{\sum_{i,j}\mathbf{M_c}}. \tag{4}$$

We calculate the similarity between $\mathbf{F_{map}}$ and all global features and retrieve the top-K images to form the sub-library. A similar retrieval is conducted for the sub-library. We then get the fine mask $\mathbf{M_f} \in \mathbb{R}^{h\times w}$ for local retrieval.

3.3. Self-Retrieval

The distribution gap between the camouflaged and generated images may result in some pixels being misrecognized. To mitigate this, we propose Self-Retrieval (SR), where the embedding $\mathbf{F}$ itself acts as the prototype. Given the mask $\mathbf{M_1} = \frac{\mathbf{M_c}+\mathbf{M_f}}{2}$ from the G2L stage, we obtain the background and foreground prototypes as

$$\mathbf{P^b} = \frac{\sum_{i,j}\mathbf{M_1} \odot \mathbf{F}}{\sum_{i,j}\mathbf{M_1}}, \tag{5}$$

$$\mathbf{P^f} = \frac{\sum_{i,j}(1-\mathbf{M_1}) \odot \mathbf{F}}{\sum_{i,j}(1-\mathbf{M_1})}. \tag{6}$$

It is notable that while $\mathbf{M_1}$ may contain noise, mask averaging pooling mitigates this issue and enables prototypes to roughly characterize the foreground and background.

For each $\mathbf{F_{i,j}}$, we compute its cosine similarity to the background and foreground prototypes and characterize the environment similarity using the difference between them.

$$\mathbf{S^b_{i,j}} = \frac{\mathbf{F_{i,j}} \cdot \mathbf{P^b}}{\|\mathbf{F_{i,j}}\|\|\mathbf{P^b}\|}, \tag{7}$$

$$\mathbf{S^f_{i,j}} = \frac{\mathbf{F_{i,j}} \cdot \mathbf{P^f}}{\|\mathbf{F_{i,j}}\|\|\mathbf{P^f}\|}, \tag{8}$$

$$\mathbf{S^s_{i,j}} = \mathbf{S^b_{i,j}} - \mathbf{S^f_{i,j}}. \tag{9}$$

After normalizing $\mathbf{S^s}$, we apply KDE-AT again to obtain the environment mask $\mathbf{M_2} \in \mathbb{R}^{h\times w}$ for the SR stage.

The final mask could be derived by reversing the environment mask,

$$\mathbf{M} = 1 - \frac{\mathbf{M_1} + \mathbf{M_2}}{2}. \tag{10}$$

3.4. Kernel Density Estimation-based Adaptive Threshold

In this section, we introduce the aforementioned Kernel Density Estimation-based Adaptive Threshold (KDE-AT). We propose KDE-AT to provide an adaptive threshold for each similarity distribution instead of setting a fixed threshold for all distributions via hyperparameter search.

Taking the similarity $\mathbf{S} \in \mathbb{R}^{h\times w}$ in G2L stage for example, we first reshape it to $\mathbb{R}^{hw}$. The kernel density estimator could be formulated as

$$\text{KDE}(x) = \frac{1}{hw}\sum_{i=1}^{hw}K_b(x - x_i) = \frac{1}{hwb}\sum_{i=1}^{hw}K(\frac{x - x_i}{b}), \tag{11}$$

Method	Feat. Extra.	CHAMELEON				CAMO				COD10K				NC4K			
		$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$
FreeSOLO	DenseCL-ResNet101	0.659	0.730	0.503	0.101	0.534	0.600	0.333	0.177	0.630	0.725	0.421	0.093	0.611	0.781	0.441	0.128
DiffCut	SD-XL-SSD-1B	0.537	0.592	0.374	0.245	0.630	0.683	0.491	0.166	0.591	0.628	0.379	0.160	0.671	0.727	0.531	0.134
LOST	DINO-ViT-S/16	0.581	0.732	0.406	0.141	0.573	0.701	0.431	0.173	0.631	0.742	0.438	0.095	0.634	0.755	0.505	0.128
Spectral	DINO-ViT-S/8	0.622	0.678	0.471	0.178	0.603	0.673	0.487	0.210	0.588	0.614	0.391	0.169	0.679	0.733	0.559	0.143
MaskCut	DINO-ViT-B/8	0.616	0.653	0.473	0.191	0.593	0.640	0.456	0.218	0.594	0.616	0.393	0.189	0.655	0.693	0.524	0.178
MaskCut	DINOv2-ViT-L/14	0.687	0.719	0.531	0.115	0.630	0.658	0.494	0.194	0.574	0.572	0.352	0.190	0.646	0.668	0.503	0.177
TokenCut	DINO-ViT-S/16	0.652	0.742	0.508	0.134	0.641	0.714	0.506	0.160	0.661	0.739	0.474	0.100	0.721	0.802	0.608	0.101
TokenCut	DINOv2-ViT-L/14	0.682	0.705	0.494	0.088	0.620	0.655	0.439	0.144	0.619	0.649	0.373	0.088	0.664	0.699	0.492	0.108
FOUND	DINO-ViT-S/8	0.552	0.602	0.381	0.218	0.526	0.578	0.397	0.277	0.488	0.487	0.264	0.248	0.569	0.605	0.410	0.218
FOUND	DINOv2-ViT-L/14	0.659	0.745	0.490	0.104	0.558	0.622	0.373	0.172	0.605	0.675	0.386	0.100	0.621	0.693	0.455	0.131
VoteCut	DINOv2-ViT-B/14	0.707	0.748	0.567	0.103	0.633	0.693	0.495	0.163	0.641	0.706	0.439	0.104	0.688	0.751	0.555	0.114
VoteCut	DINOv2-ViT-L/14	0.657	0.677	0.474	0.112	0.526	0.522	0.303	0.196	0.619	0.646	0.389	0.103	0.609	0.623	0.412	0.140
VoteCut	DINO(v2) Ensemble	0.674	0.735	0.546	0.145	0.637	0.702	0.522	0.170	0.690	0.782	0.534	0.092	0.739	0.818	0.649	0.097
EASE	DINO-ViT-S/8	0.676	0.765	0.550	0.105	0.653	0.737	0.563	0.166	0.673	0.732	0.514	0.109	0.728	0.790	0.633	0.108
EASE	DINOv2-ViT-S/14	0.754	0.848	0.651	0.074	0.692	0.763	0.604	0.138	0.720	0.804	0.582	0.079	0.752	0.832	0.661	0.085
EASE	DINOv2-ViT-L/14	**0.819**	**0.899**	**0.741**	**0.044**	**0.749**	**0.831**	**0.684**	**0.098**	**0.773**	**0.866**	**0.656**	**0.040**	**0.800**	**0.884**	**0.735**	**0.056**

Table 1. Quantitative comparisons of unsupervised segmentation with eight unsupervised methods on four commonly used COD datasets. "↑ / ↓": the higher/lower the better. The best and second-best results are **bolded** and underlined to highlight.

where K denotes the kernel function and b the bandwidth. In this paper, we adopt the gaussian kernel. So we have

$$\mathrm{KDE}(x) = \frac{1}{\sqrt{2\pi}} \frac{1}{hwb\sigma} \sum_{i=1}^{hw} e^{-\frac{(x-x_i)^2}{2b^2\sigma^2}}. \tag{12}$$

σ denotes the standard deviation. For the bandwidth, we follow Scott's Rule [62] and set it as

$$b = (hw)^{-\frac{1}{d+4}}, \tag{13}$$

where d denotes the data dimension.

As shown in Fig. 1, the probability density estimation of $\mathbf{S}$ has a bimodal-like shape, so we adopt the inter-peak minima as the threshold to isolate the environment from the camouflaged objects. For more visualizations, please see **Supplementary Fig. 1**.

4. Experiment

4.1. Experimental Settings

Datasets and Evaluation Metrics. We evaluate our method on four commonly used benchmarks, including CHAMELEON (76 test images) [66], CAMO (250 test images) [42], COD10K (2,026 test images) [15] and NC4K (4,121 test images) [51]. Following previous works, we adopt structure measure (S_α) [13], mean E-measure (E_ϕ) [14], weighted F-measure (F_β^ω) [52] and mean absolute error (M) [59] for evaluation.

Implementation Details. LLaVA-1.5-7b [50] and Stable Diffusion V1-5 [61] are adopted as the LMM and diffusion model, respectively. We choose DINOv2-ViT-L14 [55] as the self-supervised model and images are resized to 476×476. We generate $l = 32$ images for each environment category. For the top-N prototypes and top-K images during retrieval, we set $N = 1024$ and $K = 64$, respectively. Experiments are conducted on an L20. The FAISS library [35] is leveraged for efficient retrieval.

4.2. Comparison with the SOTAs

Comparison Methods. We conduct comparison experiments in two settings: unsupervised segmentation and prompt-based segmentation. For unsupervised segmentation, we compare EASE with six DINO-based methods, including LOST [64], TokenCut [73], Spectral [54], MaskCut [72], FOUND [65] and VoteCut [1], one DenseCL-based method FreeSOLO [71] and a diffusion-based method DiffCut [10]. We adopt the official codes to implement these methods for COD. For DINO-based methods, following CuVLER [1], images are resized to 480×480 for DINO [6] and 476×476 for DINOv2 [55]. Post-processing, such as Conditional Random Field (CRF) [40], is discarded for all methods. For prompt-based segmentation, we extract bounding boxes from masks and prompt SAM [39] or HQSAM [37] to implement segmentation. We compare our method with four prompt-based methods (MMCPF [70], GenSAM [28], WS-SAM [23] and ProMaC [29]). We also provide seven SOTA fully-supervised methods (FDNet [82], ZoomNet [56], SegMaR [33], HitNet [30], PopNet [76], FSPNet [32] and FEDER [22]) for reference.

Quantitative comparisons. For unsupervised segmentation, as shown in Tab. 1, our method essentially outperforms previous methods under different configurations. Especially, the best results of our method achieve a $\sim$10% lead on all metrics across all datasets compared to the previous SOTAs. As indicated in Tab. 2, for prompt-based segmentation, EASE incorporated with SAM substantially outperforms its SAM-based counterparts. On the more challenging COD10K and NC4K datasets, our method even compares favorably with fully-supervised methods. These results demonstrate the effectiveness of our concept of "separating the environment".

Qualitative comparisons. As shown in Fig. 3, compared to other methods, EASE enables better localization and segmentation of objects, especially for those small and imper-

Method	SAM	CHAMELEON				CAMO				COD10K				NC4K			
		$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$
Fully-Supervised																	
FDNet	-	0.895	0.951	0.849	0.027	0.840	0.896	0.782	0.063	0.838	0.921	0.747	0.030	0.832	0.895	0.759	0.052
ZoomNet	-	0.902	0.943	0.845	0.023	0.820	0.877	0.752	0.066	0.838	0.888	0.729	0.029	0.853	0.896	0.784	0.043
SegMaR	-	0.906	0.951	0.860	0.025	0.815	0.874	0.753	0.071	0.833	0.899	0.724	0.034	0.841	0.896	0.781	0.046
HitNet	-	**0.921**	**0.967**	**0.897**	**0.019**	0.849	**0.906**	**0.809**	0.055	**0.871**	**0.935**	**0.806**	**0.023**	0.875	**0.926**	**0.834**	0.037
PopNet	-	0.917	0.965	0.875	0.020	0.808	0.859	0.744	0.077	0.851	0.910	0.757	0.028	0.861	0.909	0.802	0.042
FSPNet	-	0.908	0.943	0.851	0.023	**0.856**	0.899	0.799	**0.050**	0.851	0.895	0.735	0.026	**0.879**	0.915	0.816	**0.035**
FEDER	-	0.887	0.946	0.834	0.030	0.802	0.867	0.738	0.071	0.822	0.900	0.716	0.032	0.847	0.907	0.789	0.044
Prompt-based Segmentaion																	
WS-SAM-P	SAM-ViT-H	0.805	0.868	0.700	0.056	0.718	0.757	0.602	0.102	0.791	0.856	0.663	0.039	0.813	0.859	0.734	0.057
WS-SAM-S	SAM-ViT-H	0.820	0.887	0.723	0.048	0.759	0.814	0.667	0.092	0.803	0.877	0.680	0.038	0.829	0.886	0.757	0.052
MMCPF	HQSAM-ViT-H	-	-	-	-	0.749	0.820	0.680	0.101	0.733	0.803	0.592	0.066	0.767	0.826	0.681	0.083
GenSAM	SAM-ViT-H	0.767	0.827	0.673	0.075	0.738	0.803	0.674	0.106	0.773	0.832	0.667	0.065	0.810	0.866	0.751	0.065
ProMaC*	SAM-ViT-H	0.833	0.899	-	0.044	0.767	0.846	-	0.090	0.805	0.876	-	0.042	-	-	-	-
EASE	SAM-ViT-H	0.853	0.912	0.812	0.044	0.777	0.841	0.748	0.099	0.856	0.914	0.801	0.028	0.849	0.902	0.815	0.051
EASE	HQSAM-ViT-H	**0.864**	**0.916**	**0.827**	**0.037**	**0.807**	**0.865**	**0.771**	**0.078**	**0.866**	**0.918**	**0.811**	**0.023**	**0.866**	**0.915**	**0.833**	**0.039**

Table 2. Quantitative comparisons of prompt-based segmentation. "-": Not available. "*": Results from the paper. The best and second-best results are **bolded** and underlined to highlight.

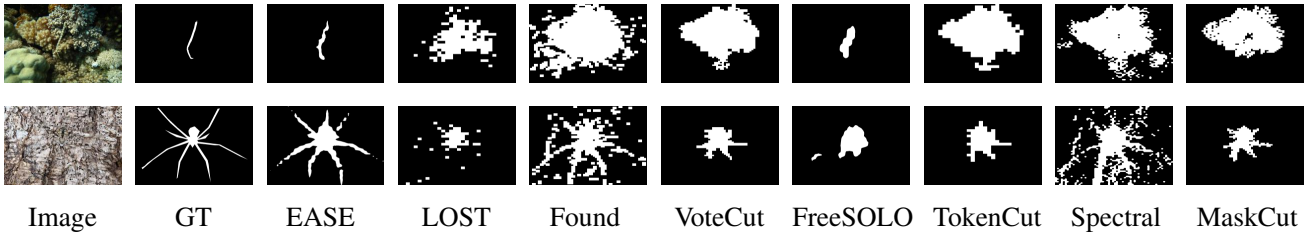

Figure 3. Qualitative comparisons with unsupervised methods.

ceptible ones.

4.3. Ablation Study

Effectiveness of DiffPro. Instead of obtaining environment prototypes from the Web or existing annotated datasets, our proposed DiffPro leverages a series of foundation models to automatically craft high-quality, environment-only prototypes, avoiding the time-consuming and laborious web collection or utilizing annotated data. To validate the effectiveness of DiffPro, we compare it with two methods: one (Web) crawling images from the Web following keywords, and the other (GT) using the COD training sets directly, i.e., the training sets of COD10K and CAMO, to obtain prototypes by masking the foreground. As indicated by Tab. 3, DiffPro keeps on par or outperforms these two methods.

Effectiveness of Retrieval Schemes. We set the baseline to be the global retrieval and its optimal threshold is determined by hyperparameter search with an interval of 0.1. Our KDE-AT assigns a distribution-related threshold to each image, avoiding additional hyperparameters while achieving a lead of $\sim$1%, as illustrated by rows a and b of Tab. 4. While global retrieval portrays the profile of the environment, it may get confused when dealing with local details. Our proposed G2L remedies this and delineates objects and regions at a finer level, bringing about a performance gain, as indicated in rows b and c. Finally, considering that there may be prototypes that are similar to both the environment and the object, SR retrieves the features themselves as prototypes, further stretching the similarity distribution to achieve the best results (row d).

4.4. Further Analysis

Sensitivity of Hyperparameters. We conduct hyperparameter sensitivity analysis on CAMO, CHAMELEON, and COD10K. For simplicity, we adopt score $= S_\alpha + E_\phi + F_\beta^\omega$ to indicate the overall performance. As shown in Fig. 4, EASE is robust to different hyper-parameter settings. We believe this stems from the fact that we discard the most influential threshold hyperparameter as well as the design of multifaceted retrieval schemes.

Different Foundation Models. For feature extractors, we experiment DINO [6], DINOv2 [55], MoCoV3 [9], MAE [59], and generative models [18, 61]. For diffusion models, we conduct experiments on SD-V1-5 [61], SD-V2-1 [61], SD-XL [60], and SD-3.5-L-Turbo [12]. For SAM, we adopt HQ-SAM [37] and the vanilla SAM [39]. Our experiments show that: a). DINOv2 outperforms other self-supervised models; b). Different diffusion models per-

Method	CHAMELEON				CAMO				COD10K				NC4K			
	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$
GT	0.816	0.893	0.740	**0.042**	**0.749**	0.822	0.683	0.109	0.759	0.846	0.641	0.058	0.793	0.873	0.724	0.066
Web	**0.821**	0.895	**0.741**	0.043	0.747	0.822	0.679	0.104	0.770	0.862	0.654	0.044	**0.806**	**0.886**	**0.741**	**0.055**
DiffPro	0.819	**0.899**	**0.741**	0.044	**0.749**	**0.831**	**0.684**	**0.098**	**0.773**	**0.866**	**0.656**	**0.040**	0.800	0.884	0.735	0.056

Table 3. Ablation experiments of DiffPro.

index	baseline	KDE-AT	G2L	SR	CHAMELEON				CAMO				COD10K				NC4K			
					$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$
a	✔	✘	✘	✘	0.793	0.877	0.704	0.056	0.720	0.808	0.638	0.099	0.737	0.830	0.600	0.048	0.749	0.832	0.666	0.074
b	✔	✔	✘	✘	0.806	0.891	0.722	0.048	0.729	0.807	0.648	0.096	0.749	0.838	0.615	0.045	0.763	0.842	0.679	0.067
c	✔	✔	✔	✘	0.812	0.894	0.729	0.047	0.741	0.826	0.672	0.103	0.763	0.857	0.639	0.043	0.788	0.875	0.716	0.062
d	✔	✔	✔	✔	**0.819**	**0.899**	**0.741**	**0.044**	**0.749**	**0.831**	**0.684**	**0.098**	**0.773**	**0.866**	**0.656**	**0.040**	**0.800**	**0.884**	**0.735**	**0.056**

Table 4. Ablation experiments of retrieval schemes.

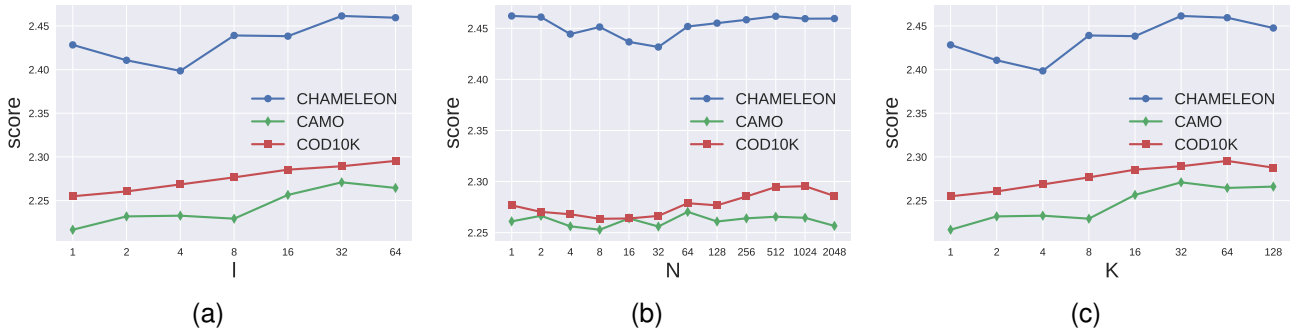

Figure 4. Hyperparameter sensitivity analysis. (a) Number of generated images for each environment category; (b) Average similarity of Top-N prototypes; (c)Top-K environment images for local retrieval.

Method	MAS3K				RMAS500			
	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$
LOST	0.646	0.753	0.490	0.107	0.566	0.672	0.358	0.132
FreeSOLO	0.613	0.693	0.403	0.111	0.552	0.670	0.283	0.102
TokenCut	0.714	0.790	0.570	0.089	0.575	0.655	0.366	0.152
Spectral	0.653	0.687	0.497	0.143	0.548	0.571	0.354	0.212
MaskCut	0.684	0.714	0.537	0.145	0.543	0.563	0.347	0.231
FOUND	0.557	0.580	0.370	0.205	0.483	0.491	0.261	0.253
DiffCut	0.513	0.628	0.314	0.218	0.436	0.537	0.200	0.265
VoteCut	0.633	0.685	0.450	0.127	0.550	0.567	0.288	0.140
EASE	**0.780**	**0.868**	**0.682**	**0.048**	**0.720**	**0.822**	**0.585**	**0.049**

Table 5. Quantitative comparisons on marine animal segmentation.

form comparably; c). HQ-SAM outperforms SAM, and the segmentation performance does not tend to saturate as the model parameters increase. Please see **Supplementary Tab. 1, 2, and 3** for more details.

Different Layers. For the best-performing feature extractor DINOv2, we experiment with features from different layers. Our experiments show that features from the last layer capture the most valuable information. Please refer to **Supplementary Fig. 2**, **Tab. 4** and **5**.

Generalization. We apply our method to marine animal segmentation [80], which is also quite challenging due to varying environmental conditions [81]. Two commonly used datasets, MAS3K [46] (1,141 test images) and RMAS [17] (500 test images), are adopted for evaluation. As indicated by Tab. 5, our approach delivers consistent performance advancements, which further demonstrates the effectiveness of EASE in isolating the environment.

5. Conclusion

In this paper, unlike previous COD work that focuses on mining camouflaged objects from the environment, we shift the paradigm by removing the salient environment and thus obtaining the target. We isolate environments and highlight camouflaged objects by retrieving from a library of environment prototypes. This is accomplished by the "prototype production line" DiffPro and multifaceted retrieval schemes, including kernel density estimation-based adaptive threshold, global-to-local retrieval and inter-feature similarity-based self-retrieval. We conduct extensive experiments to validate the effectiveness of our method.

Acknowledgments. This work was supported in part by The Hong Kong Polytechnic University under Grants P0048387, P0044520, P0050657, and P0049586, and in part by the Tianjin Science and Technology Major Project under Grant 24ZXZSSS00420.

References

[1] Shahaf Arica, Or Rubin, Sapir Gershov, and Shlomi Laufer. CuVLER: Enhanced unsupervised object discoveries through exhaustive self-supervised transformers. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 23105–23114, 2024. 3, 6

[2] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. OpenFlamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, pages 1–20, 2023. 4

[3] Luca Barsellotti, Roberto Amoroso, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. Training-free open-vocabulary segmentation with offline diffusion-augmented prototype generation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3689–3698, 2024. 3

[4] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego De Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. In *International Conference on Machine Learning*, pages 2206–2240, 2022. 3

[5] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. COCO-Stuff: Thing and stuff classes in context. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1209–1218, 2018. 4

[6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *IEEE International Conference on Computer Vision*, pages 9650–9660, 2021. 3, 5, 6, 7

[7] Chunyuan Chen, Weiyun Liang, Donglin Wang, Bin Wang, and Jing Xu. Vision-inspired boundary perception network for lightweight camouflaged object detection. *IEEE Signal Processing Letters*, pages 1–5, 2025. 2

[8] Huafeng Chen, Dian Shao, Guangqian Guo, and Shan Gao. Just a hint: Point-supervised camouflaged object detection. In *European Conference on Computer Vision*, pages 332–348, 2025. 1

[9] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *IEEE International Conference on Computer Vision*, pages 9640–9649, 2021. 7

[10] Paul Couairon, Mustafa Shukor, Jean-Emmanuel Haugeard, Matthieu Cord, and Nicolas Thome. Diffcut: Catalyzing zero-shot semantic segmentation with diffusion features and recursive normalized cut. In *Advances in Neural Information Processing Systems*, pages 1–24, 2024. 3, 6

[11] Wenliang Dai, Junnan Li, DONGXU LI, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *Advances in Neural Information Processing Systems*, pages 49250–49267, 2023. 4

[12] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning*, pages 1–28, 2024. 7

[13] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-Measure: A new way to evaluate foreground maps. In *IEEE International Conference on Computer Vision*, pages 4558–4567, 2017. 6

[14] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment measure for binary foreground map evaluation. In *International Joint Conference on Artificial Intelligence*, pages 698–704, 2018. 6

[15] Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Camouflaged object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2774–2784, 2020. 1, 2, 3, 6

[16] Deng-Ping Fan, Ge-Peng Ji, Ming-Ming Cheng, and Ling Shao. Concealed object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6024–6042, 2022. 1, 2

[17] Zhenqi Fu, Ruizhe Chen, Yue Huang, En Cheng, Xinghao Ding, and Kai-Kuang Ma. MASNet: A robust deep marine animal segmentation network. *IEEE Journal of Oceanic Engineering*, 49(3):1104–1115, 2024. 8

[18] Yatharth Gupta, Vishnu V. Jaddipal, Harish Prabhala, Sayak Paul, and Patrick Von Platen. Progressive knowledge distillation of stable diffusion xl using layer level loss. *arXiv preprint arXiv:2401.02677*, pages 1–9, 2024. 7

[19] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In *International Conference on Machine Learning*, pages 3929–3938, 2020. 3

[20] Chao Hao, Zitong Yu, Xin Liu, Yuhao Wang, Weicheng Xie, Jingang Shi, Huanjing Yue, and Jingyu Yang. Distribution-specific learning for joint salient and camouflaged object detection. *Available at SSRN 5089840*, pages 1–33, 2024. 2

[21] Chao Hao, Zitong Yu, Xin Liu, Jun Xu, Huanjing Yue, and Jingyu Yang. A simple yet effective network based on vision transformer for camouflaged object and salient object detection. *IEEE Transactions on Image Processing*, 34:608–622, 2025. 2

[22] Chunming He, Kai Li, Yachao Zhang, Longxiang Tang, Yulun Zhang, Zhenhua Guo, and Xiu Li. Camouflaged object

detection with feature decomposition and edge reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 22046–22055, 2023. 6

[23] Chunming He, Kai Li, Yachao Zhang, Guoxia Xu, Longxiang Tang, Yulun Zhang, Zhenhua Guo, and Xiu Li. Weakly-supervised concealed object segmentation with SAM-based pseudo labeling and multi-scale feature grouping. In *Advances in Neural Information Processing Systems*, pages 30726–30737, 2023. 1, 6

[24] Chunming He, Kai Li, Yachao Zhang, Yulun Zhang, Zhenhua Guo, Xiu Li, Martin Danelljan, and Fisher Yu. Strategic preys make acute predators: Enhancing camouflaged object detectors by generating camouflaged objects. In *International Conference on Learning Representations*, pages 1–10, 2024. 2

[25] Chunming He, Rihan Zhang, Fengyang Xiao, Chenyu Fang, Longxiang Tang, Yulun Zhang, Linghe Kong, Deng-Ping Fan, Kai Li, and Sina Farsiu. RUN: Reversible unfolding network for concealed object segmentation. *arXiv preprint arXiv:2501.18783*, pages 1–14, 2025. 2

[26] Ruozhen He, Qihua Dong, Jiaying Lin, and Rynson W.H. Lau. Weakly-supervised camouflaged object detection with scribble annotations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 781–789, 2023. 1

[27] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, pages 6840–6851, 2020. 4

[28] Jian Hu, Jiayi Lin, Shaogang Gong, and Weitong Cai. Relax image-specific prompt requirement in SAM: A single generic prompt for segmenting camouflaged objects. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12511–12518, 2024. 2, 3, 6

[29] Jian Hu, Jiayi Lin, Junchi Yan, and Shaogang Gong. Leveraging hallucinations to reduce manual prompt dependency in promptable segmentation. In *Advances in Neural Information Processing Systems*, pages 1–13, 2024. 2, 3, 4, 6

[30] Xiaobin Hu, Shuo Wang, Xuebin Qin, Hang Dai, Wenqi Ren, Donghao Luo, Ying Tai, and Ling Shao. High-resolution iterative feedback network for camouflaged object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 881–889, 2023. 2, 6

[31] Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. OPERA: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13418–13427, 2024. 2

[32] Zhou Huang, Hang Dai, Tian-Zhu Xiang, Shuo Wang, Huai-Xin Chen, Jie Qin, and Huan Xiong. Feature shrinkage pyramid for camouflaged object detection with transformers. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5557–5566, 2023. 2, 6

[33] Qi Jia, Shuilian Yao, Yu Liu, Xin Fan, Risheng Liu, and Zhongxuan Luo. Segment, magnify and reiterate: Detecting camouflaged objects the hard way. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4703–4712, 2022. 2, 6

[34] Yao Jiang, Xinyu Yan, Ge-Peng Ji, Keren Fu, Meijun Sun, Huan Xiong, Deng-Ping Fan, and Fahad Shahbaz Khan. Effectiveness assessment of recent large vision-language models. *Visual Intelligence*, 2(1):1–17, 2024. 2

[35] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547, 2019. 6

[36] Laurynas Karazija, Iro Laina, Andrea Vedaldi, and Christian Rupprecht. Diffusion Models for Open-Vocabulary Segmentation. In *European Conference on Computer Vision*, pages 299–317, 2024. 3

[37] Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Segment anything in high quality. In *Advances in Neural Information Processing Systems*, pages 29914–29934, 2023. 6, 7

[38] Jooyeon Kim, Eulrang Cho, Sehyung Kim, and Hyunwoo J. Kim. Retrieval-augmented open-vocabulary object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 17427–17436, 2024. 3

[39] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *IEEE International Conference on Computer Vision*, pages 3992–4003, 2023. 2, 6, 7

[40] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in Neural Information Processing Systems*, pages 1–9, 2011. 6

[41] Hala Lamdouar, Weidi Xie, and Andrew Zisserman. The making and breaking of camouflage. In *IEEE International Conference on Computer Vision*, pages 832–842, 2023. 2

[42] 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. 6

[43] Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13872–13882, 2024. 2

[44] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, pages 9459–9474, 2020. 3

[45] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning*, pages 19730–19742, 2023. 4

[46] Lin Li, Eric Rigall, Junyu Dong, and Geng Chen. MAS3K: An open dataset for marine animal segmentation. In *Benchmarking, Measuring, and Optimizing*, pages 194–212, 2021. 8

[47] Weiyun Liang, Jiesheng Wu, Xinyue Mu, Fangwei Hao, Ji Du, Jing Xu, and Ping Li. Weighted dense semantic aggregation and explicit boundary modeling for camouflaged object detection. *IEEE Sensors Journal*, 24(13):21108–21122, 2024. 2

[48] Weiyun Liang, Jiesheng Wu, Yanfeng Wu, Xinyue Mu, and Jing Xu. FINet: Frequency injection network for lightweight camouflaged object detection. *IEEE Signal Processing Letters*, 31:526–530, 2024. 2

[49] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, pages 34892–34916, 2023. 4

[50] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024. 2, 6

[51] Yunqiu Lv, Jing Zhang, Yuchao Dai, Aixuan Li, Bowen Liu, Nick Barnes, and Deng-Ping Fan. Simultaneously localize, segment and rank the camouflaged objects. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 11586–11596, 2021. 2, 6

[52] Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2014. 6

[53] Haiyang Mei, Ge-Peng Ji, Ziqi Wei, Xin Yang, Xiaopeng Wei, and Deng-Ping Fan. Camouflaged object segmentation with distraction mining. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8768–8777, 2021. 1, 2

[54] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Deep spectral methods: A surprisingly strong baseline for unsupervised semantic segmentation and localization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8364–8375, 2022. 3, 6

[55] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024:1–32, 2024. 1, 2, 3, 5, 6, 7

[56] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2150–2160, 2022. 2, 6

[57] Shubham Parashar, Zhiqiu Lin, Tian Liu, Xiangjue Dong, Yanan Li, Deva Ramanan, James Caverlee, and Shu Kong. The neglected tails in vision-language models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 12988–12997, 2024. 3

[58] Bohao PENG, Zhuotao Tian, Shu Liu, Ming-Chang Yang, and Jiaya Jia. Scalable language model with generalized continual learning. In *International Conference on Learning Representations*, pages 1–23, 2024. 3

[59] Federico Perazzi, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. Saliency filters: Contrast based filtering for salient region detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 733–740, 2012. 6, 7

[60] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, pages 1–21, 2023. 7

[61] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 2, 4, 6, 7

[62] David W Scott. *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, 2015. 6

[63] Jianbo Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000. 3

[64] Oriane Siméoni, Gilles Puy, Huy V. Vo, Simon Roburin, Spyros Gidaris, Andrei Bursuc, Patrick Pérez, Renaud Marlet, and Jean Ponce. Localizing objects with self-supervised transformers and no labels. In *Proceedings of the British Machine Vision Conference*, pages 1–25, 2021. 3, 6

[65] Oriane Siméoni, Chloé Sekkat, Gilles Puy, Antonín Vobecký, Éloi Zablocki, and Patrick Pérez. Unsupervised object localization: Observing the background to discover objects. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3176–3186, 2023. 3, 6

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

[67] Ze Song, Xudong Kang, Xiaohui Wei, Haibo Liu, Renwei Dian, and Shutao Li. FSNet: Focus scanning network for camouflaged object detection. *IEEE Transactions on Image Processing*, 32:2267–2278, 2023. 2

[68] Yujia Sun, Geng Chen, Tao Zhou, Yi Zhang, and Nian Liu. Context-aware cross-level fusion network for camouflaged object detection. In *International Joint Conference on Artificial Intelligence*, pages 1025–1031, 2021. 2

[69] Yucheng Suo, Fan Ma, Linchao Zhu, and Yi Yang. Knowledge-enhanced dual-stream zero-shot composed image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 26951–26962, 2024. 3

[70] Lv Tang, Peng-Tao Jiang, Zhi-Hao Shen, Hao Zhang, Jin-Wei Chen, and Bo Li. Chain of Visual Perception: Harnessing multimodal large language models for zero-shot camouflaged object detection. In *ACM International Conference on Multimedia*, page 8805–8814, 2024. 2, 3, 4, 6

[71] Xinlong Wang, Zhiding Yu, Shalini De Mello, Jan Kautz, Anima Anandkumar, Chunhua Shen, and Jose M. Alvarez. FreeSOLO: Learning to segment objects without annotations. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 14176–14186, 2022. 3, 6

[72] Xudong Wang, Rohit Girdhar, Stella X. Yu, and Ishan Misra. Cut and learn for unsupervised object detection and instance

segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3124–3134, 2023. 3, 6

[73] Yangtao Wang, Xi Shen, Shell Xu Hu, Yuan Yuan, James L. Crowley, and Dominique Vaufreydaz. Self-supervised transformers for unsupervised object discovery using normalized cut. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 14543–14553, 2022. 3, 6

[74] Yuan Wang, Rui Sun, Naisong Luo, Yuwen Pan, and Tianzhu Zhang. Image-to-image matching via foundation models: A new perspective for open-vocabulary semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3952–3963, 2024. 3

[75] Jiesheng Wu, Weiyun Liang, Fangwei Hao, and Jing Xu. Mask-and-edge co-guided separable network for camouflaged object detection. *IEEE Signal Processing Letters*, 30: 748–752, 2023. 2

[76] Zongwei Wu, Danda Pani Paudel, Deng-Ping Fan, Jingjing Wang, Shuo Wang, Cédric Demonceaux, Radu Timofte, and Luc Van Gool. Source-free depth for object pop-out. In *IEEE International Conference on Computer Vision*, pages 1032–1042, 2023. 1, 2, 6

[77] Fengyang Xiao, Sujie Hu, Yuqi Shen, Chengyu Fang, Jinfa Huang, Chunming He, Longxiang Tang, Ziyun Yang, and Xiu Li. A survey of camouflaged object detection and beyond. *arXiv preprint arXiv:2408.14562*, pages 1–26, 2024. 2

[78] Jilan Xu, Yifei Huang, Junlin Hou, Guo Chen, Yuejie Zhang, Rui Feng, and Weidi Xie. Retrieval-augmented egocentric video captioning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13525–13536, 2024. 3

[79] Jin Zhang, Ruiheng Zhang, Yanjiao Shi, Zhe Cao, Nian Liu, and Fahad Shahbaz Khan. Learning camouflaged object detection from noisy pseudo label. In *European Conference on Computer Vision*, pages 158–174, 2025. 1

[80] Pingping Zhang, Tianyu Yan, Yang Liu, and Huchuan Lu. Fantastic animals and where to find them: Segment any marine animal with dual SAM. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2578–2587, 2024. 8

[81] Chen Zhao, Weiling Cai, Chenyu Dong, and Chengwei Hu. Wavelet-based fourier information interaction with frequency diffusion adjustment for underwater image restoration. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8281–8291, 2024. 8

[82] Yijie Zhong, Bo Li, Lv Tang, Senyun Kuang, Shuang Wu, and Shouhong Ding. Detecting camouflaged object in frequency domain. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4504–4513, 2022. 1, 2, 6

[83] Tao Zhou, Yi Zhou, Chen Gong, Jian Yang, and Yu Zhang. Feature aggregation and propagation network for camouflaged object detection. *IEEE Transactions on Image Processing*, 31:7036–7047, 2022.

[84] Hongwei Zhu, Peng Li, Haoran Xie, Xuefeng Yan, Dong Liang, Dapeng Chen, Mingqiang Wei, and Jing Qin. I can find you! boundary-guided separated attention network for camouflaged object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3608–3616, 2022. 2