Shift the Lens: Environment-Aware Unsupervised Camouflaged Object Detection –Supplementary Material–

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,*]
[1]College of AI, Nankai University, China
[2]Department of Computing, The Hong Kong Polytechnic University, Hong Kong

Abstract

*In this supplementary material, we provide additional analysis across **7 key aspects**, including similarity distribution visualizations, generative models, SAMs, feature extractors, features from different positions, layers of DINOv2, and failure cases.*

1. Similarity distribution visualizations

Our method, EASE, separates foreground and background by comparing the target to a library of environmental prototypes. EASE generates a similarity distribution through retrieval, which typically shows a bimodal shape. The higher similarity values correspond to the environment, while the lower values indicate camouflaged objects. Therefore, the left peak of the distribution generally represents camouflaged objects, and the right peak represents the environment. To segment these peaks and identify camouflaged objects, we apply adaptive thresholds using the KDE-AT method. For a clearer understanding of our approach, please refer to Fig. 1.

2. Different generative models

At the DiffPro stage, we adopt generative models to generate prototype images of the environment. Diffusion models, as a branch of generative models, have dominated current research with their powerful conditional generation capabilities. We experiment different versions of diffusion models, including SD-V1-5 [14], SD-V2-1 [14], SD-XL [13], and SD-3.5-L-Turbo [6]. Although different versions of models generate images with different details, resolutions, and richness, our experiments show that different generative models have a limited impact on the results, as shown in Tab. 1. We believe this stems from our unique retrieval design, especially the global-to-local G2L retrieval, which allows the model to focus on a small portion of high-quality prototypes for retrieval rather than the entire prototype library. **Therefore, even if the overall quality of the prototype images generated by diffusion models, such as SD-V1-5, is not as good as that of the latest models, our method can always find a small portion of high-quality prototypes from this large number of prototypes to complete the retrieval**.

3. Different SAMs

For prompt-based segmentation, we experiment with different SAMs, as shown in Tab. 2. With the same ViT architecture, HQ-SAM [10] could capture more details, especially boundary information, than vanilla SAM [11]. In addition, the segmentation performance of both HQ-SAM and SAM is enhanced with increasing ViT parameters.

*Corresponding authors: Jing Xu and Ping Li.

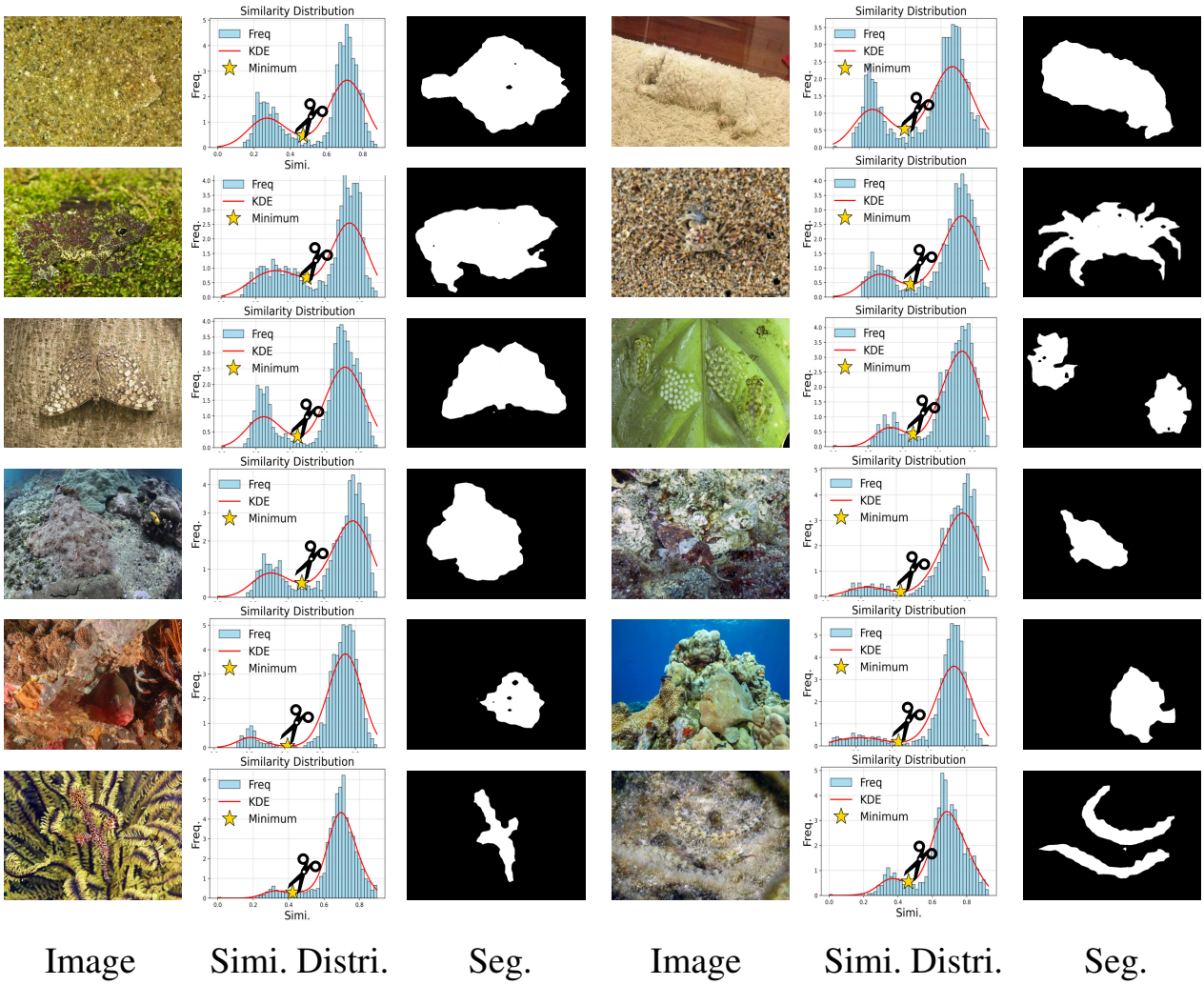

Figure 1. More visualizations of the bimodal-like similarity distribution.

4. Different feature extractors

During the DiffPro stage, we use the self-supervised model to generate environment prototypes. In the Retrieval phase, we use the same model to generate patch embeddings. We experiment with 15 feature extractors, ranging from optimization methods to model architectures, as shown in Tab. 3. These feature extractors include DINO series [3], DINOv2 series [12], MOCOv3 series [4], MAE series [8], and generative models [7, 14]. For DINO-based, MOCOV3-based, and MAE-based models, following previous practice [15, 16], we extract the **KEY** features from the last layer. For DINOv2-based models, we extract features from the last layer. Following DiffCut [5], we extract features from the last attention block of the encoder for SD-XL-SSD-1B [7] and features from the last attention block of the second phase of the encoder for SD-V1-5 [14]. Our experiments demonstrate that DINOv2 yields the best performance, a result that aligns with previous studies [1, 2, 9, 17], which highlight the robust feature extraction capabilities of DINOv2.

5. Features from different positions

For the best-performing feature extractor, DINOv2, we utilize features from the last layer. We also experiment with **Q/K/V** features from the last layer, as shown in Tab. 4. Unlike the DINO-based methods [15, 16] that use **K** features, the experiments show that directly using features from the last layer performs best.

Stable Diffusion	CHAMELEON				CAMO				COD10K				NC4K			
	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$
SD-V1-5	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**
SD-V2-1	0.815	0.898	0.739	0.046	0.749	**0.834**	0.686	0.100	0.767	0.859	0.651	0.044	0.795	0.879	0.730	0.059
SD-XL	0.815	0.893	0.735	0.045	0.746	0.825	0.678	0.101	0.772	0.867	0.657	0.041	0.801	0.885	0.736	0.057
SD-3.5-L-Turbo	**0.827**	**0.906**	**0.752**	**0.041**	**0.754**	0.832	**0.692**	0.101	**0.774**	**0.868**	**0.661**	0.042	**0.806**	**0.890**	**0.745**	**0.056**

Table 1. Ablation experiments of different generative models. "↑ / ↓": the higher/lower the better. The best results are **bolded** to highlight.

SAM	CHAMELEON				CAMO				COD10K				NC4K			
	S	E_mean	F_weight	M	S	E_mean	F_weight	M	S	E_mean	F_weight	M	S	E_mean	F_weight	M
SAM-ViT-B	0.821	0.892	0.776	0.054	0.734	0.807	0.695	0.114	0.826	0.892	0.763	0.034	0.817	0.881	0.778	0.061
SAM-ViT-L	0.849	0.912	0.808	0.044	0.769	0.834	0.736	0.102	0.851	0.912	0.795	0.028	0.845	0.902	0.811	0.049
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
HQSAM-ViT-B	0.833	0.882	0.786	0.046	0.775	0.838	0.733	0.089	0.846	0.905	0.786	0.027	0.845	0.899	0.805	0.046
HQSAM-ViT-L	0.848	0.904	0.810	0.044	0.795	0.858	0.756	0.083	0.861	**0.919**	0.805	0.024	0.860	0.913	0.826	0.040
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. Ablation experiments of SAM. "↑ / ↓": the higher/lower the better. The best results are **bolded** to highlight.

Feat. Extra.	CHAMELEON				CAMO				COD10K				NC4K			
	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$
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
DINO-ViT-S/16	0.657	0.726	0.512	0.130	0.630	0.696	0.522	0.187	0.641	0.694	0.459	0.129	0.698	0.756	0.587	0.131
DINO-ViT-B/8	0.642	0.727	0.511	0.127	0.591	0.668	0.484	0.208	0.616	0.671	0.437	0.138	0.675	0.742	0.566	0.137
DINO-ViT-B/16	0.652	0.733	0.509	0.123	0.609	0.674	0.503	0.208	0.618	0.665	0.436	0.150	0.686	0.745	0.577	0.140
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
DINOv2-ViT-B/14	0.796	0.879	0.713	0.061	0.734	0.815	0.667	0.113	0.744	0.828	0.615	0.064	0.776	0.858	0.700	0.072
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**
DINOv2-ViT-G/14	0.813	**0.904**	0.734	0.050	0.748	0.828	**0.691**	0.113	0.764	0.850	0.649	0.054	0.797	0.877	0.732	0.063
MOCOV3-ViT-S/16	0.534	0.595	0.300	0.139	0.548	0.618	0.381	0.195	0.624	0.705	0.422	0.107	0.624	0.705	0.422	0.107
MOCOV3-ViT-B/16	0.565	0.668	0.379	0.154	0.546	0.634	0.400	0.211	0.584	0.645	0.383	0.146	0.633	0.703	0.506	0.151
MAE-ViT-B/16	0.585	0.651	0.405	0.172	0.537	0.602	0.395	0.248	0.554	0.586	0.354	0.205	0.623	0.665	0.487	0.190
MAE-ViT-L/16	0.481	0.590	0.231	0.183	0.493	0.581	0.302	0.234	0.536	0.595	0.303	0.174	0.586	0.651	0.418	0.177
MAE-ViT-H/14	0.537	0.620	0.342	0.192	0.530	0.614	0.387	0.250	0.536	0.569	0.331	0.221	0.599	0.647	0.456	0.205
SD-XL-SSD-1B	0.577	0.603	0.348	0.122	0.594	0.639	0.419	0.147	0.605	0.659	0.374	0.090	0.608	0.665	0.417	0.124
SD-V1-5	0.523	0.644	0.302	0.169	0.473	0.593	0.284	0.250	0.483	0.588	0.213	0.194	0.493	0.619	0.286	0.219

Table 3. Ablation experiments of different feature extractors. "↑ / ↓": the higher/lower the better. The best results are **bolded** to highlight.

POS.	CHAMELEON				CAMO				COD10K				NC4K			
	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$
Q	0.472	0.361	0.102	0.136	0.483	0.404	0.173	0.172	0.498	0.383	0.113	0.091	0.456	0.335	0.086	0.149
K	0.794	0.895	0.715	0.055	0.702	0.772	0.629	0.142	0.693	0.766	0.543	0.093	0.725	0.801	0.637	0.104
V	0.754	0.847	0.652	0.082	0.669	0.741	0.596	0.177	0.652	0.717	0.501	0.141	0.703	0.774	0.611	0.134
Last Layer	**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 DINOv2 features from different positions of the last layer. We experiment with the features from Q/K/V/Last Layer features. The performance is optimal for features directly from the last layer.

6. Different layers of DINOv2

We also explore features from various layers of DINOv2, as shown in Tab. 5. The results reveal that features from the final layer perform the best. We hypothesize that this is because the shallower layers of DINOv2 focus on capturing low-level details, while the deeper layers encode higher-level semantic information, which is essential for distinguishing the environment from camouflaged objects.

To provide a more visual comparison of how features from different layers affect the experimental results, we visualize the similarity distributions across layers, as shown in Fig. 2. As the layer depth increases, the similarity distribution transitions

Layer	CHAMELEON				CAMO				COD10K				NC4K			
	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_\phi \uparrow$	$F_\beta^\omega \uparrow$	$M \downarrow$
1	0.383	0.477	0.172	0.344	0.359	0.467	0.184	0.371	0.398	0.443	0.150	0.322	0.434	0.517	0.239	0.305
3	0.393	0.522	0.167	0.317	0.368	0.486	0.187	0.357	0.414	0.466	0.161	0.302	0.446	0.532	0.248	0.289
5	0.379	0.521	0.124	0.280	0.358	0.493	0.170	0.353	0.401	0.454	0.139	0.302	0.434	0.534	0.228	0.292
7	0.388	0.509	0.104	0.238	0.370	0.491	0.171	0.329	0.417	0.482	0.141	0.266	0.443	0.540	0.224	0.270
9	0.386	0.525	0.102	0.236	0.387	0.522	0.166	0.294	0.429	0.520	0.133	0.225	0.441	0.550	0.120	0.247
11	0.411	0.570	0.163	0.243	0.415	0.528	0.231	0.317	0.433	0.490	0.176	0.267	0.463	0.549	0.264	0.275
13	0.454	0.578	0.219	0.225	0.450	0.548	0.277	0.291	0.468	0.523	0.222	0.238	0.495	0.578	0.308	0.249
15	0.523	0.622	0.327	0.199	0.486	0.582	0.325	0.273	0.503	0.543	0.273	0.218	0.538	0.610	0.368	0.225
17	0.427	0.258	0.000	0.140	0.406	0.253	0.001	0.181	0.455	0.264	0.001	0.091	0.422	0.256	0.002	0.153
18	0.427	0.261	0.001	0.140	0.407	0.256	0.004	0.181	0.455	0.266	0.004	0.091	0.423	0.260	0.006	0.153
19	0.426	0.256	0.001	0.140	0.408	0.259	0.009	0.182	0.460	0.277	0.015	0.091	0.428	0.270	0.017	0.153
20	0.457	0.319	0.068	0.137	0.432	0.303	0.062	0.177	0.498	0.361	0.101	0.090	0.459	0.332	0.086	0.150
21	0.687	0.711	0.518	0.090	0.593	0.588	0.396	0.151	0.685	0.722	0.499	0.069	0.665	0.685	0.504	0.108
22	0.802	0.886	0.723	0.053	0.724	0.796	0.647	0.114	0.761	0.858	0.645	0.052	0.779	0.860	0.710	0.073
23	**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 5. Ablation experiments of different layers in DINOv2. The features from the last layer contain higher-level semantics and are capable of distinguishing between the environment and camouflaged objects, thus achieving the best performance. For some layers and some images, there may be no inter-peak minima, we set the threshold to 0.5.

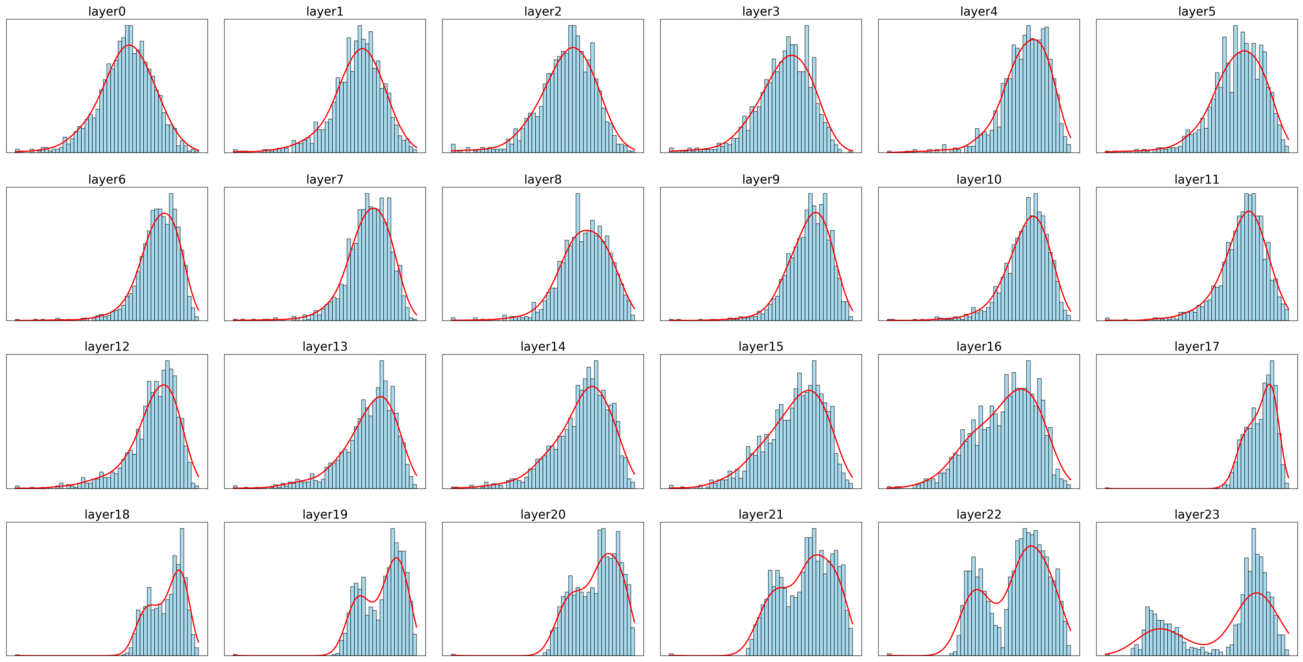

Figure 2. Similarity distributions for features from different layers of DINOv2. The image is randomly selected.

from a single peak to a bimodal shape.

7. Failure cases

Considering the effect of outliers, we only consider similarity thresholds in the range of 0.2 to 0.8 in this paper. If the threshold is not detected in this range, 0.5 is adopted by default. We present three types of failure cases in Fig. 3: (a) When the distribution around the minima is relatively flat, using this value as the threshold may overlook some details, as shown in

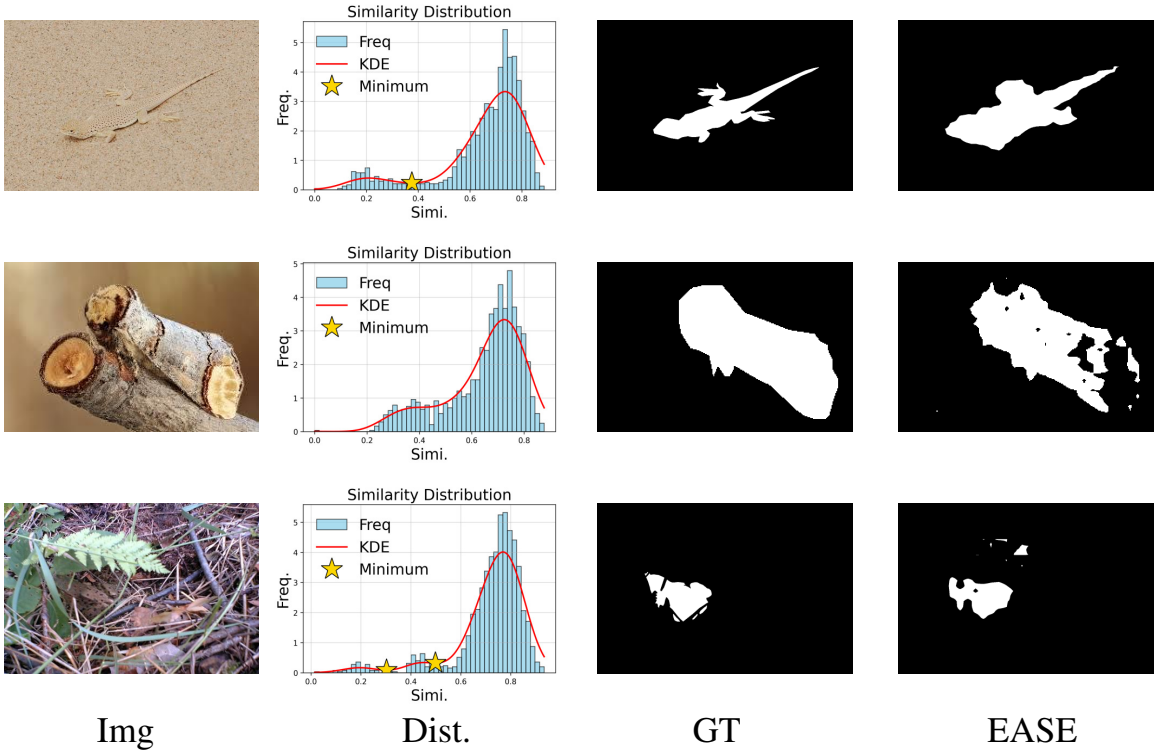

Figure 3. Failure cases.

the first row. (b) When the similarity distribution does not exhibit a clear bimodal shape, the inter-peak minima are absent, as seen in row 2. In this case, we set the threshold to 0.5. While this fixed threshold can localize objects, it struggles to capture finer details. (c) When multiple minima are present, as shown in row 3, we select the first minimum as the threshold, which similarly leads to missing information.

These failures primarily result from the non-discriminatory nature of the similarity distribution between the environment and the camouflaged object. To address this issue, we propose introducing additional prototype categories, such as specific camouflaged object categories, as a potential solution. We plan to continue exploring this challenge in future research.

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. 2

[2] 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. 2

[3] 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. 2

[4] 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. 2

[5] 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. 2

[6] 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. 1

[7] 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. 2

[8] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 2

[9] 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. 2

[10] 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. 1

[11] 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. 1

[12] 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. 2

[13] 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. 1

[14] 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. 1, 2

[15] 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. 2

[16] 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. 2

[17] 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. 2