Generalized Mirror Detection

Anonymous Author(s)
Affiliation
Address
email

Abstract

Reflective imaging enables the mirror imagings and physical entities to possess identical attributes, *e.g.,* color and shape. Current mirror detection (MD) methods primarily rely on designing functional components to establish the correlation and disparities between the imagings and entities, thereby identifying the mirror regions. However, the exploration of extended scenes with dynamic content changes is rarely investigated. Therefore, we propose the MirrorSAM designed for MD based on the Segment Anything Model (SAM). Specifically, due to the varying reflections produced by mirrors in different positions and the complex visual space that interferes with localization, we design the hierarchical mixture of direction experts (HMDE) in the low-rank space to reduce biases towards entities in SAM and dynamically adjust experts based on input scene. We observe differences in depth between mirrors and adjacent areas, and propose the depth token calibration (DTC), which introduces a learnable depth token to generate depth map and serve as an error correction factor. We further formulate the selective pixel-prototype contrastive (SPPC) loss, selecting partially confusable samples to promote the decoupling of mirror and non-mirror representations. Extensive experiments conducted on four mirror benchmarks and four settings demonstrate that our approach surpasses state-of-the-art methods with few trainable parameters and FLOPs. We further extend to four transparent surface benchmarks to validate generalization.

1 Introduction

Existing segmentation and detection works have made remarkable progress in physical entity perception, but lacks exploration into virtual imaging. When objects with nearly identical attributes coexist in the same visual space without proper distinction, it can severely hinder downstream tasks, *e.g.,* path planning [1] and 3D reconstruction [2]. Mirror detection (MD) aims to differentiate between mirror and non-mirror regions, providing prior guidance for accurate scene understanding.

Existing MD methods typically rely on static designs, *e.g,* context comparison [3], mirror symmetry characteristics [4], or low-level visual differences [5] (*e.g.,* texture), which lack adaptability to dynamic scenarios. Besides, MD encounters several challenges: 1) Reflection interference, requiring strategies to handle complete/partial correspondences between entities and imagings; 2) Distinguishing mirrors from other smooth and reflective objects, *e.g.,* tiles and glasses; 3) Deformation and occlusion, where variations in mirror regions caused by shooting angles or obstructions make it difficult to rely on predefined shape priors for discrimination. A common approach to address these issues involves combining multiple specialized components in a certain manner to expand the representation space, posing computational complexity and coordination challenges among components. Furthermore, mirrors can reflect any entity, but existing datasets only capture a subset of possibilities, lacking adequate prior knowledge. Therefore, we propose the MirrorSAM based on SAM [6], which incorporates direction-aware expert groups for fine-tuning, depth calibration, and partial pixel-prototype contrastive learning. This raises three key questions: *1) Why introduce expert*

learning and direction-aware mechanism? 2) Why introduce depth calibration, and how does it differ from depth priors? 3) Why not use full-pixel contrastive learning?

We answer the first question. SAM trained on numerous images containing physical entities, excels at capturing entity representations but remains insensitive to virtual imagings. Initially, we utilize LoRA [7] with minimal learnable parameters to fine-tune the image encoder for the MD task, yet the performance is not superior (Table 4). To broaden the perceptual space, we introduce mixture of experts (MoE) [8] that dynamically select subsets of highly responsive experts based on input to uniformly choose solutions. Each expert is responsible for partial representations and scenes, collaborating with others to avoid conflicts and redundancies that arise from directly combining multiple components. This approach not only reduces computational complexity but also simplifies the structure of experts, minimizing internal uncertainties. In specific scenarios, the relationships between all corresponding entities and imagings exhibit a certain directionality. In other words, different entity-imaging regions can implicitly achieve consistency through directions acting as bridges. Unlike previous works [9, 4] that establish symmetry by fixed-angle rotations, *e.g.,* 90 degrees or multiples, we propose the direction-aware mechanism with coordinate systems and kernel learning. This allows for arbitrary-angle consistency, expanding the potential solution space.

We answer the second question. Based on the physical principle of planar mirror imaging [10], the distances from an object to the mirror surface and its reflection are equal. Consequently, the depth within the mirror region is significantly greater than that of adjacent non-mirror regions, and accurate depth differences can effectively aid in localization. Unlike multimodal segmentation frameworks [11] which directly use depth maps as additional inputs and progressively fuse them with RGB features, we propose a more efficient approach. Specifically, we introduce an extra token to generate depth map at the output end and utilize error map for calibration. Considering: 1) In real-world scenarios, obtaining $\langle$Depth, RGB$\rangle$ pairs is often impractical; 2) Dual-stream networks incur high training and deployment costs. Our single-stream framework achieves high-quality predictions without requiring paired testing data.

We answer the third question. Pixel-to-pixel contrastive learning focuses on local regions and lacks contextual understanding, which may cause local optima and high computational complexity of $\mathcal{O}(N^2)$. Prototypes capture the global characteristics of mirror and non-mirror regions, providing robust references for pixel-level differentiation with a reduced complexity of $\mathcal{O}(N)$. Simple samples contribute little to training, and hard samples may introduce noises or outliers, thus excessive focus risks overfitting. To balance these issues, we prioritize semi-hard samples, which encourage the model to refine the attention on critical regions. Additionally, our approach restricts contrastive learning to within-sample, *i.e.,* non-generalized intra-batch contrasts.

Technically, we introduce the HMDE to bridge the representation gap between physical entities and mirrored imagings. The HMDE dynamically adjusts the structure based on scene context, enabling omni-perception. Furthermore, we propose the DTC, which differs from leveraging output tokens for generating detection maps. Instead, depth tokens are utilized for generating depth maps and guiding the model to focus on error-prone areas. To disentangle representations, we formulate the SPPC loss to emphasize pixel and prototype differences, especially for confusable samples.

In summary, our main contributions are as follows:

- We formulate the MirrorSAM based on expert learning and hierarchical modeling for perceiving mirror regions. We revisit the design paradigms in MD and, to the best of our knowledge, are the first to adapt SAM to the MD task.
- We propose three customized components, the HMDE to automatically adapt to various scenes and imagings content changes, the DTC as the constraint to encourage focusing on depth error regions, refine prediction results, and the SPPC loss to selectively learn decoupled and compacted representations.
- Extensive experiments on four mirror and four transparent surface benchmarks validate the superiority of the proposed method and the effect of components.

2 Related Work

Mirror Detection. Mirrors reflect physical entities and create identical yet illusory imagings, which can seriously confuse and impact the understanding and modeling of visual space. For the fully-

supervised setting, Yang *et al.* [12] pioneered the first MD method, MirrorNet, which leverages the correlation between internal and external features of mirrors. Lin *et al.* [13] designed the PMDNet, which compares mirror features with context for correspondence and incorporates edge information. Guan *et al.* [3] constructed semantic associations among objects based on graph representation. These methods share the common motivation, *i.e.,* establishing the associations between entities and imagings. Huang *et al.* [9] developed the dual-stream network based on Transformer to exploit the symmetry property of mirrors. He *et al.* [4] presented the HetNet, which combines low-level and high-level features in the heterogeneous manner. Both aim to utilize rotation strategies to construct mirror symmetry consistency. Other works [11, 14, 15, 5] leveraged the structure differences between mirror and non-mirror regions to distinguish, *i.e.,* depth, content distribution and frequency. For the data-efficient setting, Zha *et al.* [16] introduced the first weakly supervised dataset and model based on scribble annotations, *i.e.,* WSMD, achieving performance comparable to fully supervised methods. Lin *et al.* [17] formulated the self-supervised pre-training strategy. For the video-level setting, Lin *et al.* [18] proposed the first video-level dataset and model, *i,e,* VMD. Warren *et al.* [19] improved based on inconsistent motion clues. Xu *et al.* [20] focused on extremely-weak supervision.

SAM for Downstream Tasks. SAM is trained on massive natural images, and recent efforts attempted to adapt it to various downstream tasks, *e.g.,* medical and camouflage scenarios. [21] introduced several adapters containing two-layer MLPs. [22] integrated traditional segmentation frameworks with SAM and directly encoded depth maps as cues. [23] designed a dual-stream multi-scale guidance architecture for underwater animal scenes. [24] developed the hierarchical decoding paradigm to refine prediction results. However, MD-specific SAM remains unexplored. Our MirrorSAM explores and proves the potential for the first time.

Mixtures of Experts. MoE [8, 25] follows the divide-and-conquer strategy, where the routing network are utilized to activate experts and assign weights, dynamically adjusting structure based on different inputs and ultimately aggregating predictions. Recently, some research [26, 27] focus on optimizing the internal mechanisms, *e.g.,* addressing expert load balancing during training, reusing inactive experts during inference, and achieving efficiency. However, the lack of design for experts causes focusing on homogeneous content. We construct hierarchical and heterogeneous experts.

3 Proposed Method

3.1 Overall Architecture

As illustrated in Figure 1, the MirrorSAM framework builds upon the basic structure of SAM. The HMDE is integrated into the image encoder and the DTC is incorporated into the pixel decoder. Using the last layer features of the decoder and the predicted maps, we generate prototypes and compute the SPPC loss. The primary loss function is adapted based on the specific supervision setting.

3.2 Hierarchical Mixture of Direction Experts

In previous works [23, 24], LoRA is directly applied to the image encoder to modulate features for specific tasks. Considering the complexity and ambiguity of the MD visual space, we introduce expert learning and directional correlation. We expand expert groups within each LoRA to increase representation capacity, enabling better handling of diverse scenes and imaging content variations. When imagings and corresponding entities align perfectly, we can establish global semantic awareness based on directional consistency. For partial matches, directional signals are learned from paired samples and propagated to unmatched ones to achieve complementarity.

Specifically, given input image $\mathbf{X} \in \mathbb{R}^{3\times H_0\times W_0}$, we utilize the encoder to obtain multi-scale features $\mathbf{F}^i \in \mathbb{R}^{C_i\times H_i\times W_i}$ (channel, height, and width) and optimize parameters through the HMDE. We define the angle θ of each directional convolution kernel $\mathbf{K}_\theta(m,n) \in \mathbb{R}^{C_i\times k_h\times k_w}$, and calculate the transformed direction by applying the rotation matrix to the standard convolution $\mathbf{K}(m',n')$,

$$\begin{bmatrix} m' \\ n' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} m - c_m \\ n - c_n \end{bmatrix} \tag{1}$$

where m' and n' are the rotated coordinates, c_m and c_n are the coordinates of the center point.

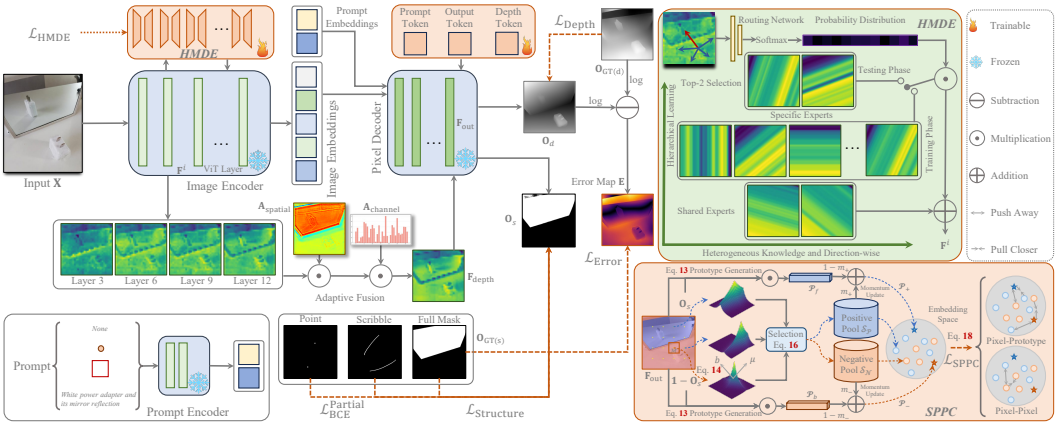


Figure 1: The framework of our MirrorSAM. Given any input image, we utilize the image encoder to extract features, where the HMDE modulates features at each stage. We adaptively fuse features from multiple scales. Subsequently, depth, prompt, and output tokens are fed to the decoder to obtain depth estimation maps and prediction maps. For the prediction maps, we apply the SPPC loss for constraint. For the depth maps, we apply two losses, one for supervision and the other for calibration. When weak (scribble or point) supervision is applied, we only adjust the output side, *i.e.,* loss function.

We then obtain feature updates specific to the direction,

$$\mathbf{F}^i_\theta(c', p, q) := \sum_{c=1}^{C_i} \sum_{m=0}^{k_h-1} \sum_{n=0}^{k_w-1} \mathbf{F}^i(c, p+m, q+n) \cdot \mathbf{K}_\theta(c', c, m, n) \tag{2}$$

where (p, q) represents the pixel position. Based on the directional kernel, we further extend to the MoE paradigm. MoE comprises two core elements: the routing network $\mathcal{G}$ and the experts $\mathcal{E}$. The routing network assigns weights to the experts, while the experts handle specific representations. Intuitively, imagings and entities features could be allocated to different experts, enabling heterogeneous processing and decoupling. For $\mathbf{F}^i$, we have,

$$\mathcal{G}(\mathbf{F}^i) = \texttt{Softmax}\left(\mathcal{F}(\texttt{GAP}(\mathbf{F}^i)) + \mathbb{N}\right) \tag{3}$$

where $\mathcal{F}$, GAP, and $\mathbb{N}$ denote convolution, global average pooling, and Gaussian distribution noise, respectively. Noise is used to mitigate biases and is only applied during the training phrase, being removed during testing phrase. We set learnable θ for each expert and integrate features from several directions. However, assigning weights and activations to all expert by $\mathcal{G}$ may weaken communication between the experts and be inefficient. We divide the expert group into shared and specific experts. The former perceive general reflection imaging and associated directions, while the latter, anchored to the former, construct consistency and customized constraints,

$$\mathbf{F}^i := \sum_{k=1}^{N_s} \mathcal{E}^k(\mathbf{F}^i) + \sum_{k=N_s+1}^{N_e} \mathcal{G}^k(\mathbf{F}^i) \cdot \mathcal{E}^k(\mathbf{F}^i; \theta^k) + \mathbf{F}^i \tag{4}$$

where N_s and N_e represent the number of shared experts and total experts, respectively. During the testing phrase, to ensure high efficiency, we activate the Top-2 weighted experts and ignore the remaining. Compared to LoRA, despite the additional learning cost introduced, the performance improvement is significant.

Ideally, the workload of all experts is $\frac{1}{N_e}$. To avoid overload of partial experts and idleness of others, we reduce the excessive reliance of the gate network based on mutual information (MI) [28] $\mathcal{I}$,

$$\mathcal{I}(\mathbf{F}; \mathcal{G}(\mathbf{F})) = \mathcal{H}(\mathcal{G}(\mathbf{F})) - \mathcal{H}(\mathcal{G}(\mathbf{F})|\mathbf{F}) \tag{5}$$

where smaller $\mathcal{I}$ indicates more balanced load, and the former and latter represent the entropy and conditional entropy of expert assignments, respectively. Our objective is to minimize $\mathcal{I}$. For $\mathcal{G}(\mathbf{x}^n) = [g^1(\mathbf{x}^n), g^2(\mathbf{x}^n), \ldots, g^{N_e-N_s}(\mathbf{x}^n)]$, we let $\sum_{k=N_s+1}^{N_e} g^k(\mathbf{x}^n) = 1$, where $g^k(\mathbf{x}^n)$ represents the probability of assigning the *n-th* sample in the batch B to the *k-th* expert. We reformulate Eq. 5 as,

$$\mathcal{I}(\mathbf{F}; \mathcal{G}(\mathbf{F})) = -\sum_{k=N_s+1}^{N_e} \mathbb{E}_{\mathbf{x}}[g^k(\mathbf{x})] \log \mathbb{E}_{\mathbf{x}}[g^k(\mathbf{x})] + \frac{1}{B} \sum_{n=1}^{B} \sum_{k=N_s+1}^{N_e} g^k(\mathbf{x}^n) \log g^k(\mathbf{x}^n) \tag{6}$$

To enable experts to learn heterogeneous knowledge, we leverage game theory [29] to cultivate the *competitive relationship* among experts, where each expert aims to maximize the uniqueness of own representations, *i.e.,* mutual opposition. We implement by $\mathcal{L}_{\text{comp}}$,

$$\mathcal{L}_{\text{comp}} = \sum_{i=1}^{N_e} \sum_{j=i+1}^{N_e} \mathbb{E}_{\mathbf{x}} \left[\texttt{sim}(\mathcal{E}^i(\mathbf{x}), \mathcal{E}^j(\mathbf{x})) \right] \tag{7}$$

where `sim` is the cosine similarity. The constraint loss $\mathcal{L}_{\text{HMDE}}$ for the HMDE can be defined as,

$$\mathcal{L}_{\text{HMDE}} = \lambda_{\text{MI}} \mathcal{I} + \lambda_{\text{comp}} \mathcal{L}_{\text{comp}} \tag{8}$$

where λ_{MI} and λ_{comp} are balance parameters.

3.3 Depth Token Calibration

Modeling the spatial geometry of the scenes through depth differences estimation facilitates discrimination. The most direct approach is incorporating paired depth maps or estimated by pretrained models as the prior and explicitly injecting the RGB modality to form the multi-branch architecture for input (or output). Considering error accumulation and computational costs, we introduce a depth token for depth map generation, formulating two heads for output-level calibration. Unlike segmentation branches that encode only the semantic-rich features of the encoder's final layer for decoding, we select features at multiple scales to provide rich details and semantics.

Technically, due to differences in high-level and low-level representations, we choose L scales (in practice, $L = \{3, 6, 9, 12\}$) and generate spatial and channel weight distributions, *i.e.,* $\mathbf{A}_{\text{spatial}}$ and $\mathbf{A}_{\text{channel}}$ by,

$$\mathbf{A}_{\text{spatial}}, \mathbf{A}_{\text{channel}} = \sigma(\mathcal{F}(\sum_{l=1}^{L} \mathcal{F}(\mathbf{F}^l))) \tag{9}$$

where σ denotes the sigmoid function. Note that the convolution parameter settings for generating $\mathbf{A}_{\text{spatial}}$ and $\mathbf{A}_{\text{channel}}$ are different. We further partition and allocate the overall weights to each sub-item for adaptive fusion,

$$\mathbf{F}_{\text{depth}} = \sum_{l=1}^{L} \mathbf{F}^l \cdot \mathbf{A}^l_{\text{spatial}} \cdot \mathbf{A}^l_{\text{channel}} \tag{10}$$

We fuse the learnable depth token and features $\mathbf{F}_{\text{out}}$ to derive estimated depth maps $\mathbf{O}_d$. We further utilize the depth loss $\mathcal{L}_{\text{Depth}}$ (*i.e.,* Huber loss, stable training and robust to noises) to optimize and generate error maps $\mathbf{E}$,

$$\mathbf{E} = \| \log \mathbf{O}_d - \log \mathbf{O}_{\text{GT(d)}} \| \tag{11}$$

where $\| \cdot \|$ denotes Min-Max normalization. To enhance the contrast of neighboring regions while ensuring convergence stability, we leverage logarithmic transformation. Directly applying noise-carrying $\mathbf{E}$ to the segmentation maps $\mathbf{O}_{\text{GT(s)}}$ may disrupt the correct regions. We formulate $\mathcal{L}_{\text{Error}}$ to promote focusing on error regions through implicit calibration,

$$\mathcal{L}_{\text{Error}} = \frac{-\sum_{i=1}^{H_0} \sum_{j=1}^{W_0} \mathbf{E}^{ij} \times \mathbf{O}^{ij}_{\text{GT(s)}} \log(\mathbf{O}^{ij}_s)}{\max(\sum_{i=1}^{H_0} \sum_{j=1}^{W_0} \mathbf{E}^{ij}, \epsilon)} \tag{12}$$

where ϵ is a extremely small value.

3.4 Selective Pixel-Prototype Contrastive Loss

Unlike prior works that establish robust semantic knowledge by contrasting across images, considering that: 1) Due to reflective imaging, even the same mirror placed in different scenes or at different positions within the same scene, or with adjusted spatial relationships of entities, presents different contents, indicating that mirrors are highly scene-dependent and sensitive to spatial arrangements; 2) The limited scale and diversity of existing mirror datasets make it challenging to learn universal representations; 3) Mirror and non-mirror regions within individual samples exhibit certain distributional distinctiveness. Considering the above, we formulate the hierarchical aggregation from pixel-to-prototype and pixel-to-pixel at the intra-sample level. The similarity of content inside and outside the mirror results cause pixels exhibiting ambiguity, thus we aim to select samples that are easily confused for particular attention.

In detail, we leverage the final layer feature $\mathbf{F}_{\text{out}}$ of the decoder along with the foreground and background segmentation map $\mathbf{O}_s$, $1 - \mathbf{O}_s$ to generate foreground and background prototypes, *i.e.,*

$\mathcal{P}_f$ and $\mathcal{P}_b$ as base reference,

$$\mathcal{P}_f = \frac{\sum_{i,j} \mathbf{O}_s^{ij} \cdot \mathbf{F}_{\text{out}}^{ij}}{\sum_{i,j} \mathbf{O}_s^{ij}},\ \mathcal{P}_b = \frac{\sum_{i,j} (1 - \mathbf{O}_s^{ij}) \cdot \mathbf{F}_{\text{out}}^{ij}}{\sum_{i,j} (1 - \mathbf{O}_s^{ij})} \tag{13}$$

For the pixel feature $\mathbf{f}^{ij}$ at position (i, j), we establish its Laplacian distribution to accelerate computation and improve noise resistance (compared to Gaussian distribution),

$$\mu^{ij} = \mathbf{f}^{ij},\ b^{ij} = \frac{1}{|\mathcal{M}(i,j)|} \sum_{(p,q) \in \mathcal{M}(i,j)} |\mathbf{f}^{pq} - \mu^{ij}| \tag{14}$$

where μ and b represent the location and scale parameters, respectively, $|\mathcal{M}(i, j)|$ represents the number of neighboring pixels (*e.g.,* 3×3) for the pixel $\mathbf{P}^{ij}$ (*i.e.,* dynamic region generation, as opposed to fixed partitioning at the patch level). We further calculate the distance between pixel distributions, *e.g,* negative pairs,

$$\mathcal{D}(\mathbf{P}^{ij}, \mathbf{P}^{kl}) = \| \sum_{c=1}^{C} \left(\log \frac{b^{kl,c}}{b^{ij,c}} + \frac{|\mathbf{f}^{ij,c} - \mathbf{f}^{kl,c}|}{b^{kl,c}} \right) \| \tag{15}$$

For each anchor pixel, confusable samples are selected according to the rule,

$$\mathcal{S}_{\mathcal{N}} = \{ \mathbf{P}^{kl} \mid \mathcal{D}^{mn} < \mathcal{D}^{kl} < (1 + \alpha) \cdot \mathcal{D}^{mn} \} \tag{16}$$

where $\mathcal{D}^{mn} = \mathcal{D}(\mathbf{P}^{ij}, \mathbf{P}^{mn}), \mathcal{D}^{kl} = \mathcal{D}(\mathbf{P}^{ij}, \mathbf{P}^{kl})$, $\mathbf{P}^{mn}$ represent positive sample pixels, and α is the interval parameter. The inherent uncertainty within pixels is difficult to measure through distance. We aim to assign higher weights to high-uncertainty pixels. We define the differential entropy $\mathcal{H}_{\mathcal{M}}$ of region $\mathcal{M}$ as,

$$\mathcal{H}_{\mathcal{M}} = 1 + \ln(2b) \tag{17}$$

Thus, the pixel weight w for $\mathcal{S}$ is generated by normalizing $\mathcal{H}$. Similarly, we can obtain the semi-hard positive sample pool $\mathcal{S}_{\mathcal{P}}$. We derive the SPPC loss $\mathcal{L}_{\text{SPPC}}$,

$$\mathcal{L}_{\text{SPPC}} = -\frac{1}{B} \sum_{i=1}^{B} \sum_{k=1}^{|\mathcal{S}_{\mathcal{P}}|} \log \left[w^k \cdot \texttt{sim}'(\mathcal{S}_{\mathcal{P}}^{k(i)}, \mathcal{P}_{+}^{(i)}) \ \frac{1}{\texttt{sim}'(\mathcal{S}_{\mathcal{P}}^{k(i)}, \mathcal{P}_{-}^{(i)}) + \sum_{j=1}^{|\mathcal{S}_{\mathcal{N}}|} w^j \cdot \texttt{sim}'(\mathcal{S}_{\mathcal{P}}^{k(i)}, \mathcal{S}_{\mathcal{N}}^{j(i)})} \right] \tag{18}$$

where $\texttt{sim}' = \exp(\texttt{sim})/\tau$, τ is a temperature coefficient. Note that $\mathcal{P}_+$ and $\mathcal{P}_-$ (not $\mathcal{P}_f$ and $\mathcal{P}_b$) represent the positive and negative prototypes, updated from $\mathcal{S}_{\mathcal{P}}^K$ and $\mathcal{S}_{\mathcal{N}}^K$ based on momentum m. For the fully-supervised setting, following [4], we leverage $\mathcal{L}_{\text{Structure}}$ as the primary loss,

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{Structure}} + \sum\nolimits_{\mathcal{L} \in \{\mathcal{L}_{\text{SPPC}}, \mathcal{L}_{\text{Depth}}, \mathcal{L}_{\text{Error}}, \mathcal{L}_{\text{HMDE}}\}} \lambda_{\mathcal{L}} \mathcal{L} \tag{19}$$

where $\lambda_{\mathcal{L}}$ denotes weight parameter. For the weakly-supervised setting, following [16], we switch the primary loss to $\mathcal{L}_{\text{BCE}}^{\text{Partial}}$. And we adjust $\mathcal{L}_{\text{SPPC}}$ to $\mathcal{L}_{\text{SPPC}}^{\text{Partial}}$.

4 Experiments

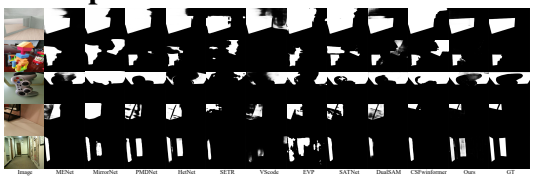


Figure 2: Qualitative comparison on MD scenarios. Best viewed by zooming in.

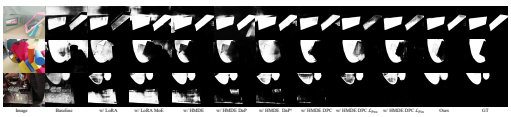


Figure 3: Qualitative ablation of proposed components. From top to bottom, the scenes include cross-imaging, complex correspondence and occlusion, and smooth tabletop reflection interference. Partial variants correspond to Table 4

Datasets. We conduct experiments on eight datasets. **Mirror datasets:** MSD [12] and PMD [13] datasets contain 3,063 and 5,096 training images, 955 and 571 testing images, respectively. MirrorD [11] contains 2,000 training images and 1,049 testing images, and is accompanied by depth maps. VMD [18] has 143 (7,835 images) and 126 (7,152 images) videos for training and testing. **Transparent surface datasets:** GDD [38] contains 2,980 training images and 936 testing images. GSD [39] consists of 3202 training pairs and 810 testing pairs, with a larger coverage of regions and contrast distributions. GlassD [40] comprises 2400 training images and 609 testing images with

Table 1: Quantitative comparison on MSD and PMD datasets. S, C, G, O, M denote salience detection, camouflage detection, general segmentation, related tasks segmentation, MD methods, respectively. Foundation models include, *e.g.,* LLMs and SAM. Best performance in **bold**, second in underline. ‡ represents data is unavailable. ↑ indicates higher values are better, while ↓ indicates the opposite.

Methods	Attr.	Transformer Encoder	Foundation Model	Prompt	MSD					PMD				
					MAE↓	S_m↑	E_m↑	F_β^w↑	IoU↑	MAE↓	S_m↑	E_m↑	F_β^w↑	IoU↑
UDUN [30] MM'23	S	✗	✗	✗	0.071	0.815	0.838	0.746	0.713	0.039	0.784	0.794	0.652	0.600
MENet [31] CVPR'23	S	✗	✗	✗	0.054	0.868	0.906	0.829	0.805	0.033	0.826	0.873	0.727	0.680
MirrorNet [12] ICCV'19	M	✗	✗	✗	0.065	0.850	0.891	0.812	0.790	0.043	0.761	0.841	0.663	0.585
PMDNet [13] CVPR'20	M	✗	✗	✗	0.047	0.875	0.908	0.845	0.815	0.032	0.810	0.859	0.716	0.660
SANet [3] CVPR'22	M	✗	✗	✗	0.054	0.862	0.898	0.829	0.798	0.071	0.808	0.839	0.721	0.668
VCNet [14] TPAMI'22	M	✗	✗	✗	0.044	‡	‡	‡	0.854	0.028	‡	‡	‡	0.694
HetNet [4] AAAI'23	M	✗	✗	✗	0.043	0.881	0.921	0.854	0.824	0.029	0.828	0.865	0.734	0.690
SETR [32] CVPR'21	S	✓	✗	✗	0.071	0.797	0.840	0.750	0.690	0.035	0.753	0.775	0.633	0.564
VST [33] ICCV'21	S	✓	✗	✗	0.054	0.861	0.901	0.818	0.791	0.036	0.783	0.814	0.639	0.591
DSAM [22] MM'24	C	✓	✓	✓	0.037	0.888	0.919	0.871	0.832	0.031	0.800	0.839	0.745	0.666
VSCode [34] CVPR'24	C	✓	✗	✓	0.077	0.800	0.820	0.721	0.687	0.042	0.787	0.816	0.656	0.607
FSEL [35] ECCV'24	C	✓	✗	✗	0.043	0.868	0.918	0.859	0.814	0.038	0.803	0.844	0.737	0.671
SAM [6] ICCV'23	G	✓	✓	✗	0.108	0.755	0.768	0.689	0.624	0.063	0.688	0.711	0.597	0.616
EVP [36] CVPR'23	G	✓	✗	✓	0.064	0.845	0.896	0.811	0.780	0.037	0.793	0.861	0.694	0.634
LISA [37] CVPR'24	G	✓	✓	✓	0.061	0.822	0.859	0.799	0.771	0.044	0.811	0.825	0.678	0.660
DualSAM [23] CVPR'24	O	✓	✓	✓	0.039	0.903	0.932	0.882	0.848	0.034	0.816	0.839	0.705	0.636
HSAM [24] CVPR'24	O	✓	✓	✓	0.042	0.878	0.909	0.865	0.833	0.032	0.808	0.833	0.721	0.619
SATNet [9] AAAI'23	M	✓	✗	✗	0.033	0.887	0.916	0.865	0.834	0.025	0.826	0.858	0.739	0.684
CSFwinformer [5] TIP'24	M	✓	✗	✗	0.045	0.875	0.905	0.846	0.821	0.024	0.831	0.864	0.756	0.700
Ours	M	✓	✓	✓	**0.025**	**0.936**	**0.958**	**0.924**	**0.913**	**0.024**	**0.867**	**0.907**	**0.788**	**0.759**

paired depth maps. Trans10K [41] consists of 10,428 images with three categories: things, stuff and background. Images are divided into 5,000, 1,000 and 4,428 images for training, validation and testing, respectively.

Table 2: Comparison on MirrorD and VMD.

Methods	Depth	MAE↓	S_m↑	E_m↑	F_β^w↑	IoU↑
VSCode [34] CVPR'24	✓	0.045	0.839	0.892	0.801	0.763
Ours	✓	**0.033**	**0.879**	**0.925**	**0.840**	**0.820**
Methods	**Video**	**MAE↓**	**S_m↑**	**E_m↑**	**F_β^w↑**	**IoU↑**
VMDNet [18] CVPR'23	✓	0.105	0.731	0.742	0.623	0.567
Ours	✗	**0.099**	**0.761**	**0.775**	**0.655**	**0.600**

Table 3: Comparison of model efficiency.

	SATNet [9] AAAI'23	CSFwinformer [5] TIP'24	DualSAM [23] CVPR'24	WSMD [16] AAAI'24	Ours
FLOPs↓ (GMAC)	153.00	139.45	39.51	21.39	4.21
Params.↓ (M)	139.36	150.54	74.30	26.16	10.23

Implementation Details. We implement the model and conduct experiments on NVIDIA A100 GPUs via Pytorch. To reduce memory consumption and align with the settings in [5, 9], we adjust the input size from the default 1024×1024 to 512×512 and initialize the parameters with pre-trained SAM-B. We utilize the AdamW optimizer to update the parameters, with the initial learning rate and the weight decay both set to 0.001. The batch size is 16, and the total epochs is 200. During the training phrase, we freeze the base components, *e.g.,* image encoder, and optimize the DTC and the HMDE. During the testing phase, we do not employ any post-processing operations, *e.g.,* CRF. If depth labels are not provided, we leverage [42] for generation.

Evaluation Metrics. We adopt seven evaluation metrics: S-measure (S_m) [43], mean E-measure (E_m) [44], weighted F-measure, (F_β^w) , maximum F-measure(F_β) [45], Intersection over union (IoU), Mean Absolute Error (MAE), and Balance Error Rate (BER). Note that the higher the better for the first five. To evaluate efficiency, we leverage model trainable parameters and computational complexity.

4.1 Comparison with SOTA Methods

Quantitative Comparison. In Table 1, our approach outperforms various types of methods, *e.g.,* DualSAM, with gains of 1.4%, 3.3%, 2.6%, 4.2%, and 6.5% respectively on the five metrics of the MSD dataset, and the average surpasses CSFwinformer by around 5.0%, which demonstrates the effectiveness of expert adaptation and differential learning. In Table 2, our method achieve comparable performance without complex modality fusion and temporal signal. In Table 3, the trainable parameters and FLOPs of MirrorSAM are about one-fifteenth and one-thirty-fifth of CSFwinformer, respectively, highlighting the efficiency.

Qualitative Comparison. We consider different scenarios and provide visualizations for comparison. In Figure 2, the first line illustrates smooth surface interference, *i.e.,* tile. The second and third rows

depict incomplete correspondence and irregular occlusion. The fourth rows pertains to situations with only imagings and no corresponding physical entities. The last row demonstrates complex multi-object perception. Our approach effectively handles intricate visual spatial variations and reflection interferences.

4.2 Ablation Study

We validate the effect of the proposed components (Table 4, Figures 3, 4, and 6) and crucial hyperparameter settings (Figure 5) on the MSD and PMD datasets.

Expert Modeling. Directly applying the vanilla SAM to MD yields unsatisfactory results, but incorporating LoRA fine-tuning (Table 4 Line 9) significantly improves performance, highlighting the necessity of domain knowledge adaptation. Introducing the HMDE to establish dynamic perceptual scene changes and perceive visual spatial content and geometric relationships in arbitrary directions further enhances effect, with the two elements (MoE and angle) proving complementary. We further boost through $\mathcal{L}_{\text{HMDE}}$-controlled diverse experts.

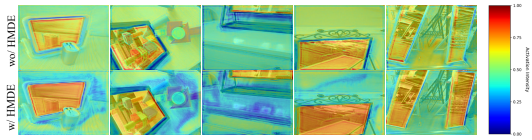


Figure 4: Heat map visualization.

Table 4: Quantitative ablation of components and strategies. DaI, DaP, DaP†, $\mathcal{L}_{\text{Pro}}$, $\mathcal{L}_{\text{Pix}}$, and $\mathcal{L}_{\text{Pix}\leftrightarrow\text{Pro}}$ denote depth as input, depth as prompt (training only), depth as prompt (full phase), prototype-to-prototype, pixel-to-pixel, and pixel-to-prototype contrastive learning, respectively.

Components/Strategies			MSD					PMD				
			MAE↓	S_m↑	E_m↑	F_β^w↑	IoU↑	MAE↓	S_m↑	E_m↑	F_β^w↑	IoU↑
HMDE	DPC	SPPC	0.108	0.755	0.768	0.689	0.624	0.063	0.688	0.711	0.597	0.616
✓	✗	✗	0.035	0.878	0.897	0.855	0.836	0.038	0.801	0.829	0.713	0.688
✓	✓	✗	0.028	0.908	0.933	0.894	0.889	0.029	0.839	0.872	0.756	0.723
✓	✗	✓	0.032	0.895	0.918	0.888	0.875	0.032	0.848	0.861	0.745	0.736
✗	✓	✓	0.055	0.803	0.811	0.744	0.685	0.051	0.746	0.766	0.651	0.645
✓	✓	✓	**0.025**	**0.936**	**0.958**	**0.924**	**0.913**	**0.024**	**0.867**	**0.907**	**0.788**	**0.759**
MoE	Angle	$\mathcal{L}_{\text{HMDE}}$	0.043	0.846	0.856	0.801	0.783	0.044	0.773	0.785	0.678	0.661
✓	✗	✗	0.039	0.857	0.874	0.824	0.801	0.042	0.784	0.796	0.689	0.672
✓	✓	✗	0.037	0.868	0.885	0.839	0.820	0.039	0.792	0.813	0.695	0.680
✓	✓	✓	**0.035**	**0.878**	**0.897**	**0.855**	**0.836**	**0.038**	**0.801**	**0.829**	**0.713**	**0.688**
DaI	DaP	DaP†	0.035	0.878	0.897	0.855	0.836	0.038	0.801	0.829	0.713	0.688
✓	✗	✗	0.033	0.885	0.883	0.866	0.848	0.035	0.815	0.838	0.725	0.695
✗	✓	✗	0.033	0.888	**0.911**	0.869	**0.860**	0.034	0.812	0.841	**0.735**	0.690
✗	✗	✓	**0.030**	**0.895**	0.904	**0.875**	0.855	**0.032**	**0.820**	**0.847**	0.729	**0.699**
$\mathcal{L}_{\text{Pro}}$	$\mathcal{L}_{\text{Pix}}$	$\mathcal{L}_{\text{Pix}\leftrightarrow\text{Pro}}$	0.028	0.908	0.933	0.894	0.889	0.029	0.839	0.872	0.756	0.723
✓	✗	✗	0.030	0.913	0.938	0.899	0.895	0.029	0.835	0.884	0.767	0.731
✗	✓	✗	0.028	0.915	**0.945**	0.906	**0.902**	0.027	**0.845**	0.880	0.769	**0.738**
✗	✗	✓	**0.027**	**0.922**	0.940	**0.908**	0.899	**0.026**	0.841	**0.888**	**0.773**	0.730
SATNet [9] AAAI'23			0.033	0.887	0.916	0.865	0.834	0.025	0.826	0.858	0.739	0.684
w/ SPPC			**0.030**	**0.901**	**0.933**	**0.880**	**0.857**	**0.024**	**0.839**	**0.873**	**0.755**	**0.701**
HSAM [24] CVPR'24			0.042	0.878	0.909	0.865	0.833	0.032	0.808	0.833	0.721	0.619
w/ SPPC			**0.038**	**0.895**	**0.926**	**0.886**	**0.861**	**0.029**	**0.826**	**0.854**	**0.732**	**0.637**
Ponit	Box	Text	0.025	0.936	0.958	0.924	0.913	0.024	0.867	0.907	0.788	0.759
✓	✗	✗	0.023	0.942	0.955	0.932	0.918	0.022	0.875	0.916	0.797	0.769
✗	✓	✗	**0.021**	**0.945**	0.961	**0.944**	**0.928**	**0.022**	**0.896**	**0.935**	**0.821**	**0.786**
✗	✗	✓	0.022	0.941	**0.964**	0.940	0.925	0.022	0.888	0.928	0.808	0.780

Depth Injection. We explore three strategies: 1) Fusing depth map and RGB image as input; 2) Using the depth map as the dense prompt during the training phase; 3) Employing for full phase, yet the performance is inferior to calibrating at the output end. We attribute this to the noise and inaccuracies in depth maps generated by pre-trained depth estimation models (or low-quality paired), which can degrade original representations when explicitly encoded and interacted with. In contrast, implicit depth learning through gradient optimization facilitates calibration and enhances feature quality. Moreover, not using depth maps during the testing phrase is more aligned with practical deployment.

Contrast Settings. Pixel-, prototype-, and pixel-prototype- level contrastive learning are all inferior to the SPPC. We analyze: 1) The insufficient number of prototypes, even for the entire batch, is not enough for contrast; 2) Pixels are easily affected by noise interference; 3) Pixel-prototype treats all samples equally. The selection and discrimination of semi-hard positive/negative samples is crucial. The SPPC can be adapted to other approaches to boost performance.

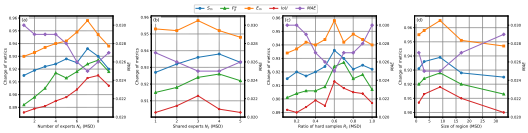


Figure 5: Hyperparameter settings.

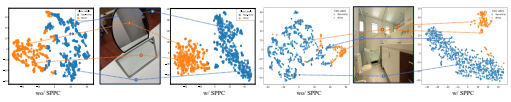


Figure 6: t-SNE visualization. The transition from loose crossover to clustered separation.

SAM Prompt Strategies. In the baseline, prompts are automatically generated. We further explore: 1) Randomly shifting several pixels based on GT to generate points and boxes prompt; 2) Leveraging Qwen 2.5 [46] to generate image descriptions, which are then encoded into embedding vectors using CLIP [47] and merged with visual features. We observe that box prompts yield better performance, possibly due to providing more visual priors and avoiding semantic conflicts with text representations.

Crucial Hyperparameters. We observe that performance gradually improves as $1 < N_e \leq 7$, but decreases when $7 < N_e$. We analyze that while each expert specializes in part of the data space, excessive experts may cause conflicts and redundancies. When $10\% < R_s \leq 60\%$, performance gradually improves, but declines when $60\% < R_s$. This indicates that the benefit from simple samples is limited, and increasing proportion of confusable samples can enhance the discriminative ability. However, too many hard samples may cause matching errors and degrade performance. See **Appendix** for further analysis.

5 Discussion

Table 5: Semi-supervised setting.

Methods	25%					35%				
	MAE	$S_m\uparrow$	$E_m\uparrow$	$F_\beta^w\uparrow$	IoU↑	MAE	$S_m\uparrow$	$E_m\uparrow$	$F_\beta^w\uparrow$	IoU↑
EVP [36] CVPR'23	0.123	0.574	0.602	0.525	0.476	0.106	0.643	0.666	0.583	0.550
HSAM [24] CVPR'24	0.110	0.618	0.592	0.561	0.552	0.098	0.677	0.657	0.625	0.612
Ours	**0.094**	**0.652**	**0.647**	**0.612**	**0.611**	**0.085**	**0.708**	**0.720**	**0.673**	**0.658**
Methods	45%					55%				
	MAE	$S_m\uparrow$	$E_m\uparrow$	$F_\beta^w\uparrow$	IoU↑	MAE	$S_m\uparrow$	$E_m\uparrow$	$F_\beta^w\uparrow$	IoU↑
EVP [36] CVPR'23	0.097	0.688	0.701	0.645	0.596	0.089	0.728	0.741	0.689	0.665
HSAM [24] CVPR'24	0.086	0.727	0.735	0.685	0.665	0.078	0.775	0.784	0.731	0.719
Ours	**0.077**	**0.759**	**0.770**	**0.718**	**0.697**	**0.065**	**0.780**	**0.803**	**0.759**	**0.748**

Table 6: Weakly-supervised setting.

Methods	Scribble-MSD					Scribble-PMD				
	MAE	$S_m\uparrow$	$E_m\uparrow$	$F_\beta^w\uparrow$	IoU↑	MAE	$S_m\uparrow$	$E_m\uparrow$	$F_\beta^w\uparrow$	IoU↑
PFA [48] ICML'24	0.075	0.815	0.886	0.771	0.762	0.057	0.760	0.805	0.642	0.585
WSMD [16] AAAI'24	0.078	0.828	0.878	0.780	0.750	**0.051**	0.773	**0.824**	0.630	0.600
Ours	**0.070**	**0.843**	**0.903**	**0.799**	**0.778**	0.053	**0.784**	0.816	**0.665**	**0.618**
Methods	Point-MSD					Point-PMD				
	MAE	$S_m\uparrow$	$E_m\uparrow$	$F_\beta^w\uparrow$	IoU↑	MAE	$S_m\uparrow$	$E_m\uparrow$	$F_\beta^w\uparrow$	IoU↑
PFA [48] ICML'24	0.155	0.673	0.755	0.573	0.573	0.095	0.689	0.751	0.563	0.486
WSMD [16] AAAI'24	0.168	0.696	0.742	0.586	0.560	0.091	0.706	0.760	0.550	0.501
Ours	**0.147**	**0.719**	**0.768**	**0.613**	**0.584**	**0.088**	**0.716**	**0.777**	**0.583**	**0.519**

Semi-supervised Setting. In Table 5, we randomly select 25%, 35%, 45%, and 55% of the training data (MSD dataset) and design the learning framework based on the plain *Mean-Teacher* [49]. As the data decreases, the performance degradation of EVP and HSAM becomes more pronounced. Our method exhibits more significant advantages compared to fully supervised setting. We analyze: 1) Insufficient prior information makes it challenging to generalize from few samples to unseen distributions, leading to overreliance on labeled data. Our approach leverages the intrinsic knowledge of fine-tuned SAM to achieve zero-shot perception and utilizes expert collaboration and depth representations to maximize the utility of limited samples; 2) Noise interference from pseudo-labels hampers efficiency in handling unlabeled data. Our contrastive strategy encourages learning consistency and generates high-quality pseudo-labels.

Table 7: Quantitative comparison on transparent surface datasets.

Methods	GDD				GSD			
	IoU↑	$F_\beta\uparrow$	MAE↓	BER↓	IoU↑	$F_\beta\uparrow$	MAE↓	BER↓
RFENet [50] IJCAI'23	0.874	0.929	0.062	5.79	0.836	0.904	0.049	6.24
CMCM [51] AAAI'25	0.883	0.933	0.059	5.65	0.849	0.912	0.050	6.02
Ours	**0.887**	**0.945**	**0.051**	**5.08**	**0.871**	**0.927**	**0.048**	**5.55**
Methods	GlassD				Trans10K			
	IoU↑	$F_\beta\uparrow$	MAE↓	BER↓	IoU↑	$F_\beta\uparrow$	MAE↓	BER↓
RFENet [50] IJCAI'23	0.699	0.825	0.046	11.42	**0.912**	‡	0.043	3.68
CMCM [51] AAAI'25	0.742	0.853	0.043	9.33	0.899	0.878	0.046	3.55
Ours	**0.764**	**0.875**	**0.038**	**8.28**	0.903	**0.895**	**0.042**	**3.13**

Weakly-supervised Setting. In Table 6, we evaluate under the weakly supervised setting based on the scribble and point annotations provided by [16] on the MSD and PMD datasets. Scribbles and points provide the basic structure and location of the foreground and background, respectively. Our method is superior and exhibits a larger gap under the point setting. Beyond the critical components, we attribute to: 1) Different from WSMD, which adopts Canny[52]-generated edge maps as supervision, SAM provides more accurate and fine-grained knowledge priors. 2) Unlike WSMD, which establishes prototype contrasts between images and prototype enhancements in PFA, we consider the association of confusable pixels and dynamic prototypes within a single sample. We argue that the representations of mirror and non-mirror regions across images do not possess strong semantic correlations, thus enforcing alignment may disrupt the original features.

Broader Impacts. In Table 7, we achieve promising performance in transparent surface segmentation as well. We attribute to the significant depth disparity between transparent and non-transparent regions, and reliance on context/orientation modeling, which is consistent with the MD.

6 Conclusion

We introduce two novel perspectives for the MD: parameter-efficient fine-tuning and expert learning. We find consistency in the directional association between imagings and entities, and design shared and specific orientation-wise MoE to adaptively adjust the image encoder. By leveraging the depth disparity between mirror and non-mirror regions and utilizing token-based perception, we avoid explicitly introducing estimated noise. We revisit the contrastive strategies and establish selective pixel-prototype learning within individual samples, rather than across samples.

References

[1] J. Tang and H. Ma, "Large-scale multi-robot coverage path planning via local search," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 16, 2024, pp. 17 567–17 574.

[2] J. Liu, X. Tang, F. Cheng, R. Yang, Z. Li, J. Liu, Y. Huang, J. Lin, S. Liu, X. Wu *et al.*, "Mirrorgaussian: Reflecting 3d gaussians for reconstructing mirror reflections," in *ECCV*, 2024.

[3] H. Guan, J. Lin, and R. W. Lau, "Learning semantic associations for mirror detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5941–5950.

[4] R. He, J. Lin, and R. W. Lau, "Efficient mirror detection via multi-level heterogeneous learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023, pp. 790–798.

[5] Z. Xie, S. Wang, Q. Yu, X. Tan, and Y. Xie, "Csfwinformer: Cross-space-frequency window transformer for mirror detection," *IEEE Transactions on Image Processing*, 2024.

[6] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4015–4026.

[7] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.

[8] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural computation*, vol. 3, no. 1, pp. 79–87, 1991.

[9] T. Huang, B. Dong, J. Lin, X. Liu, R. W. Lau, and W. Zuo, "Symmetry-aware transformer-based mirror detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023, pp. 935–943.

[10] M. Born and E. Wolf, *Principles of optics: electromagnetic theory of propagation, interference and diffraction of light*. Elsevier, 2013.

[11] H. Mei, B. Dong, W. Dong, P. Peers, X. Yang, Q. Zhang, and X. Wei, "Depth-aware mirror segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3044–3053.

[12] X. Yang, H. Mei, K. Xu, X. Wei, B. Yin, and R. W. Lau, "Where is my mirror?" in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 8809–8818.

[13] J. Lin, G. Wang, and R. W. Lau, "Progressive mirror detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3697–3705.

[14] X. Tan, J. Lin, K. Xu, P. Chen, L. Ma, and R. W. Lau, "Mirror detection with the visual chirality cue," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 3492–3504, 2022.

[15] M. Zha, F. Fu, Y. Pei, G. Wang, T. Li, X. Tang, Y. Yang, and H. T. Shen, "Dual domain perception and progressive refinement for mirror detection," *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.

[16] M. Zha, Y. Pei, G. Wang, T. Li, Y. Yang, W. Qian, and H. T. Shen, "Weakly-supervised mirror detection via scribble annotations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 7, 2024, pp. 6953–6961.

[17] J. Lin and R. W. Lau, "Self-supervised pre-training for mirror detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 12 227–12 236.

[18] J. Lin, X. Tan, and R. W. Lau, "Learning to detect mirrors from videos via dual correspondences," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9109–9118.

[19] A. Warren, K. Xu, J. Lin, G. K. Tam, and R. W. Lau, "Effective video mirror detection with inconsistent motion cues," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 17 244–17 252.

[20] K. Xu, T. W. Siu, and R. W. Lau, "Zoom: Learning video mirror detection with extremely-weak supervision," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 6, 2024, pp. 6315–6323.

[21] T. Chen, L. Zhu, C. Ding, R. Cao, Y. Wang, Z. Li, L. Sun, P. Mao, and Y. Zang, "Sam fails to segment anything?–sam-adapter: Adapting sam in underperformed scenes: Camouflage, shadow, medical image segmentation, and more," *arXiv preprint arXiv:2304.09148*, 2023.

[22] Z. Yu, X. Zhang, L. Zhao, Y. Bin, and G. Xiao, "Exploring deeper! segment anything model with depth perception for camouflaged object detection," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 4322–4330.

[23] P. Zhang, T. Yan, Y. Liu, and H. Lu, "Fantastic animals and where to find them: Segment any marine animal with dual sam," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 2578–2587.

[24] Z. Cheng, Q. Wei, H. Zhu, Y. Wang, L. Qu, W. Shao, and Y. Zhou, "Unleashing the potential of sam for medical adaptation via hierarchical decoding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 3511–3522.

[25] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," *arXiv preprint arXiv:1701.06538*, 2017.

[26] X. Wu, S. Huang, and F. Wei, "Mixture of lora experts," *arXiv preprint arXiv:2404.13628*, 2024.

[27] H. Zhao, Z. Qiu, H. Wu, Z. Wang, Z. He, and J. Fu, "Hypermoe: Towards better mixture of experts via transferring among experts," *arXiv preprint arXiv:2402.12656*, 2024.

[28] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, vol. 69, no. 6, p. 066138, 2004.

[29] D. Fudenberg, *Game theory*. MIT press, 1991.

[30] J. Pei, Z. Zhou, Y. Jin, H. Tang, and P.-A. Heng, "Unite-divide-unite: Joint boosting trunk and structure for high-accuracy dichotomous image segmentation," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 2139–2147.

[31] Y. Wang, R. Wang, X. Fan, T. Wang, and X. He, "Pixels, regions, and objects: Multiple enhancement for salient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 10 031–10 040.

[32] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. Torr *et al.*, "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6881–6890.

[33] N. Liu, N. Zhang, K. Wan, L. Shao, and J. Han, "Visual saliency transformer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 4722–4732.

[34] Z. Luo, N. Liu, W. Zhao, X. Yang, D. Zhang, D.-P. Fan, F. Khan, and J. Han, "Vscode: General visual salient and camouflaged object detection with 2d prompt learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 17 169–17 180.

[35] Y. Sun, C. Xu, J. Yang, H. Xuan, and L. Luo, "Frequency-spatial entanglement learning for camouflaged object detection," in *European Conference on Computer Vision*. Springer, 2025, pp. 343–360.

[36] W. Liu, X. Shen, C.-M. Pun, and X. Cun, "Explicit visual prompting for low-level structure segmentations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 19 434–19 445.

[37] X. Lai, Z. Tian, Y. Chen, Y. Li, Y. Yuan, S. Liu, and J. Jia, "Lisa: Reasoning segmentation via large language model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 9579–9589.

[38] H. Mei, X. Yang, Y. Wang, Y. Liu, S. He, Q. Zhang, X. Wei, and R. W. Lau, "Don't hit me! glass detection in real-world scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3687–3696.

[39] J. Lin, Z. He, and R. W. Lau, "Rich context aggregation with reflection prior for glass surface detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 13 415–13 424.

[40] J. Lin, Y. H. Yeung, and R. W. Lau, "Depth-aware glass surface detection with cross-modal context mining," *arXiv preprint arXiv:2206.11250*, 2022.

[41] E. Xie, W. Wang, W. Wang, M. Ding, C. Shen, and P. Luo, "Segmenting transparent objects in the wild," in *Proceedings of the European Conference on Computer Vision*, 2020, pp. 696–711.

[42] L. Yang, B. Kang, Z. Huang, X. Xu, J. Feng, and H. Zhao, "Depth anything: Unleashing the power of large-scale unlabeled data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 10 371–10 381.

[43] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4548–4557.

[44] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, and A. Borji, "Enhanced-alignment measure for binary foreground map evaluation," *arXiv preprint arXiv:1805.10421*, 2018.

[45] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps?" in *Proceedings of the IEEE Conference on Computer Vision and Pattern recognition*, 2014, pp. 248–255.

[46] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei *et al.*, "Qwen2. 5 technical report," *arXiv preprint arXiv:2412.15115*, 2024.

[47] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[48] G. Chan, P. Zhang, H. Dong, S. Ji, and B. Chen, "Scribble-supervised semantic segmentation with prototype-based feature augmentation," in *Forty-first International Conference on Machine Learning*.

[49] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," *Advances in neural information processing systems*, vol. 30, 2017.

[50] K. Fan, C. Wang, Y. Wang, C. Wang, R. Yi, and L. Ma, "Rfenet: Towards reciprocal feature evolution for glass segmentation," *arXiv preprint arXiv:2307.06099*, 2023.

[51] J. Lin, Y.-H. Yeung, S. Ye, and R. W. Lau, "Leveraging rgb-d data with cross-modal context mining for glass surface detection," *AAAI*, 2025.

[52] J. Canny, "A computational approach to edge detection," *IEEE Transactions on pattern analysis and machine intelligence*, no. 6, pp. 679–698, 1986.

[53] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[54] L. Li, T. Zhou, W. Wang, J. Li, and Y. Yang, "Deep hierarchical semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 1246–1257.

[55] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer, 2015, pp. 234–241.

[56] R. Azad, E. K. Aghdam, A. Rauland, Y. Jia, A. H. Avval, A. Bozorgpour, S. Karimijafarbigloo, J. P. Cohen, E. Adeli, and D. Merhof, "Medical image segmentation review: The success of u-net," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[57] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 1290–1299.

[58] C. He, K. Li, Y. Zhang, L. Tang, Y. Zhang, Z. Guo, and X. Li, "Camouflaged object detection with feature decomposition and edge reconstruction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 22 046–22 055.

[59] D.-P. Fan, G.-P. Ji, M.-M. Cheng, and L. Shao, "Concealed object detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 10, pp. 6024–6042, 2021.

[60] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.

[61] S. Yu, B. Zhang, J. Xiao, and E. G. Lim, "Structure-consistent weakly supervised salient object detection with local saliency coherence," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 4, 2021, pp. 3234–3242.

[62] Y. Pang, X. Zhao, L. Zhang, and H. Lu, "Multi-scale interactive network for salient object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9413–9422.

[63] X. Zhao, Y. Pang, L. Zhang, H. Lu, and L. Zhang, "Towards diverse binary segmentation via a simple yet general gated network," *International Journal of Computer Vision*, pp. 1–78, 2024.

[64] M.-M. Cheng*, S. Gao*, A. Borji, Y.-Q. Tan, Z. Lin, and M. Wang, "A highly efficient model to study the semantics of salient object detection," *IEEE TPAMI*, 2021.

[65] C. Xie, C. Xia, M. Ma, Z. Zhao, X. Chen, and J. Li, "Pyramid grafting network for one-stage high resolution saliency detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 717–11 726.

[66] H. Mei, X. Yang, Y. Wang, Y. Liu, S. He, Q. Zhang, X. Wei, and R. W. Lau, "Don't hit me! glass detection in real-world scenes," in *CVPR*, June 2020.

[67] H. He, X. Li, G. Cheng, J. Shi, Y. Tong, G. Meng, V. Prinet, and L. Weng, "Enhanced boundary learning for glass-like object segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 15 859–15 868.

[68] J. Lin, Z. He, and R. W. Lau, "Rich context aggregation with reflection prior for glass surface detection," in *CVPR*, 2021.

[69] P. Skurowski, H. Abdulameer, J. Błaszczyk, T. Depta, A. Kornacki, and P. Kozieł, "Animal camouflage analysis: Chameleon database," *Unpublished manuscript*, vol. 2, no. 6, p. 7, 2018.

[70] T.-N. Le, T. V. Nguyen, Z. Nie, M.-T. Tran, and A. Sugimoto, "Anabranch network for camouflaged object segmentation," *Computer vision and image understanding*, vol. 184, pp. 45–56, 2019.

[71] D.-P. Fan, G.-P. Ji, G. Sun, M.-M. Cheng, J. Shen, and L. Shao, "Camouflaged object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2777–2787.

[72] Y. Lv, J. Zhang, Y. Dai, A. Li, B. Liu, N. Barnes, and D.-P. Fan, "Simultaneously localize, segment and rank the camouflaged objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 591–11 601.

[73] J. Yan, T.-N. Le, K.-D. Nguyen, M.-T. Tran, T.-T. Do, and T. V. Nguyen, "Mirrornet: Bio-inspired camouflaged object segmentation," *IEEE Access*, vol. 9, pp. 43 290–43 300, 2021.

[74] D.-P. Fan, G.-P. Ji, T. Zhou, G. Chen, H. Fu, J. Shen, and L. Shao, "Pranet: Parallel reverse attention network for polyp segmentation," in *International conference on medical image computing and computer-assisted intervention*. Springer, 2020, pp. 263–273.

[75] Q. Zhai, X. Li, F. Yang, C. Chen, H. Cheng, and D.-P. Fan, "Mutual graph learning for camouflaged object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12 997–13 007.

[76] H. Mei, G.-P. Ji, Z. Wei, X. Yang, X. Wei, and D.-P. Fan, "Camouflaged object segmentation with distraction mining," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8772–8781.

[77] F. Yang, Q. Zhai, X. Li, R. Huang, A. Luo, H. Cheng, and D.-P. Fan, "Uncertainty-guided transformer reasoning for camouflaged object detection," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 4146–4155.

[78] M. Zhang, S. Xu, Y. Piao, D. Shi, S. Lin, and H. Lu, "Preynet: Preying on camouflaged objects," in *Proceedings of the 30th ACM international conference on multimedia*, 2022, pp. 5323–5332.

[79] Y. Pang, X. Zhao, T.-Z. Xiang, L. Zhang, and H. 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*, 2022, pp. 2160–2170.

[80] Z. Liu, Z. Zhang, Y. Tan, and W. Wu, "Boosting camouflaged object detection with dual-task interactive transformer," in *2022 26th International Conference on Pattern Recognition (ICPR)*. IEEE, 2022, pp. 140–146.

[81] L. Li, E. Rigall, J. Dong, and G. Chen, "Mas3k: An open dataset for marine animal segmentation," in *International Symposium on Benchmarking, Measuring and Optimization*. Springer, 2020, pp. 194–212.

[82] Z. Fu, R. Chen, Y. Huang, E. Cheng, X. Ding, and K.-K. Ma, "Masnet: A robust deep marine animal segmentation network," *IEEE Journal of Oceanic Engineering*, 2023.

[83] M. J. Islam, P. Luo, and J. Sattar, "Simultaneous enhancement and super-resolution of underwater imagery for improved visual perception," *arXiv preprint arXiv:2002.01155*, 2020.

[84] P. Drews-Jr, I. d. Souza, I. P. Maurell, E. V. Protas, and S. S. C. Botelho, "Underwater image segmentation in the wild using deep learning," *Journal of the Brazilian Computer Society*, vol. 27, pp. 1–14, 2021.

[85] Y. Sun, G. Chen, T. Zhou, Y. Zhang, and N. Liu, "Context-aware cross-level fusion network for camouflaged object detection," *arXiv preprint arXiv:2105.12555*, 2021.

[86] L. Li, B. Dong, E. Rigall, T. Zhou, J. Dong, and G. Chen, "Marine animal segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 4, pp. 2303–2314, 2021.

[87] J. Liu, J. Zhang, and N. Barnes, "Modeling aleatoric uncertainty for camouflaged object detection," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2022, pp. 1445–1454.

[88] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A. L. Yuille, and Y. Zhou, "Transunet: Transformers make strong encoders for medical image segmentation," *arXiv preprint arXiv:2102.04306*, 2021.

[89] A. He, K. Wang, T. Li, C. Du, S. Xia, and H. Fu, "H2former: An efficient hierarchical hybrid transformer for medical image segmentation," *IEEE Transactions on Medical Imaging*, vol. 42, no. 9, pp. 2763–2775, 2023.

[90] T. Yan, Z. Wan, X. Deng, P. Zhang, Y. Liu, and H. Lu, "Mas-sam: Segment any marine animal with aggregated features," *arXiv preprint arXiv:2404.15700*, 2024.

A Appendix

We provide prediction results on MSD, PMD and Mirror-RGBD datasets, which can be found in the `mirror_prediction_results.zip`. You can test by using the evaluation code `eval_performance.zip`. Note that IoU is calculated by `eval_IoU.py`.

A.1 Related Work

Semantic segmentation (SS) assigns each pixel to a specific category, such as mirror or non-mirror. The development of semantic segmentation based on deep learning can be roughly categorized into three groups. The first group includes convolutional paradigms represented by FCN series [53, 54], and U-Net series [55, 56]. Despite enlarging the receptive field through methods like pyramid pooling, global modeling remains challenging. The second group involves architectures based on Vision Transformers, with Mask2Former [57] as a representative, aiming to establish context awareness using self-attention. However, these methods are limited to fixed categories and may fail in open-world scenarios. Recently, some works introduced visual-language foundation models such as CLIP [47], greatly expanding zero-shot transfer capabilities. Some works are based on segmentation foundation models like SAM [6], equipped with several adapters or prompts to extend to downstream applications. Binary segmentation (BS), as a subclass of semantic segmentation, aims to distinguish between foreground and background, with specific applications such as salient object detection and camouflage object detection. Unlike SS or BS, it rarely considers interference from reflective imaging, making it difficult to directly transfer relevant methods to MD.

A.2 Evaluation Metrics

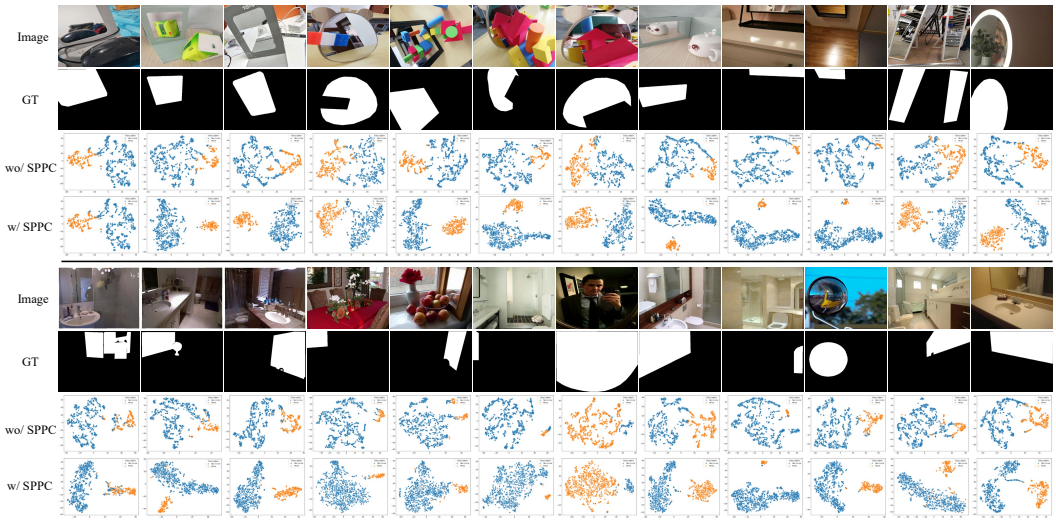
For the transparent surface benchmarks, following [50, 51], we use five metrics: Accuracy (ACC), Intersection over Union (IoU), F-measure (F_β), Mean Absolute Error (MAE), and Balance Error Rate (BER). For the camouflage and underwater benchmarks, following [58, 59], we adopt four metrics: structure-measure (S_m), enhanced-alignment measure (E_m), weighted F-measure (F_β^w), and MAE.

A.3 Implementation Details

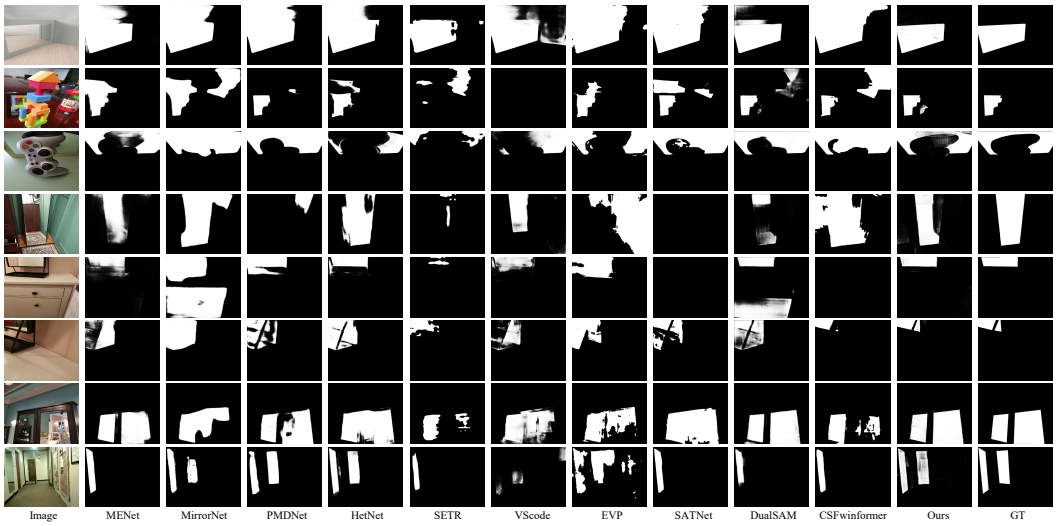
In Figure 5, we further discuss the impact of the number of shared experts and the region size centered around pixels. We find that the increment in performance is capped by the number of shared experts. We argue that too many active experts may lead to redundant and conflicting representations. While enlarging the region size continuously enhances performance, the benefits saturate, and the computational complexity significantly increases. When the region is too large, there is a noticeable decrease in performance. We attribute this to the excessive representation differences between pixels in the region, forcefully constructing statistical distributions using other information may weaken the original representation rather than enhance. To balance efficiency and performance, we set the region size to 3. For the semi-supervised setting, we adopt five random seeds to partition the dataset and report the averages. Note that for the weakly supervised settings (scribble or point), we do not utilize edge loss [16]. In Table 8, we provide the loss weight settings.

Table 8: Hyperparameter settings.

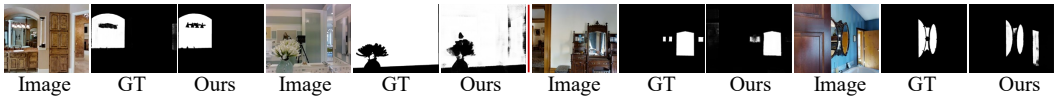
Hyperparameter	Value
λ_{MI}	0.4
λ_{comp}	0.6
λ_{SPPC}	0.1
λ_{Depth}	0.4
λ_{Error}	0.4
λ_{HMDE}	0.1



(a) t-SNE visualization of the SPPC usage before and after on MSD (upper) and PMD (lower) benchmarks.



(b) Qualitative comparison on MD scenarios. Best view by zooming in.



(c) Failure cases.

Figure 7: Qualitative analysis.

Table 9: Quantitative analysis. Best performance in bold, second in underline. ‡ represents data is unavailable. ↑ indicates higher values are better, while ↓ indicates the opposite.

(a) Performance comparison of transparent surface benchmarks.

Metric	GDD [38]					GSD [39]					Trans10K [41]				
	ACC ↑	IoU ↑	F_β ↑	MAE ↓	BER ↓	ACC ↑	IoU ↑	F_β ↑	MAE ↓	BER ↓	ACC ↑	IoU ↑	F_β ↑	MAE ↓	BER ↓
PSPNet [60]	0.904	79.23	0.892	0.110	10.00	0.845	70.53	0.816	0.104	13.22	0.863	82.38	‡	0.093	9.72
SCWSSOD [61]	0.915	81.05	0.897	0.105	9.52	0.888	76.49	0.865	0.084	9.63	‡	‡	‡	‡	‡
MINet [62]	0.925	83.80	0.910	0.083	7.94	0.908	77.41	0.860	0.080	8.77	‡	‡	‡	‡	‡
GateNet [63]	‡	81.70	0.931	0.073	8.84	‡	68.90	0.898	0.073	10.12	‡	‡	‡	‡	‡
CSNet [64]	‡	77.30	0.876	0.135	11.33	‡	66.60	0.805	0.135	14.76	‡	‡	‡	‡	‡
PGNet [65]	‡	85.70	0.930	0.074	6.82	‡	80.50	0.897	0.068	7.88	‡	‡	‡	‡	‡
GDNet [66]	0.919	81.47	0.895	0.098	8.73	0.882	76.77	0.864	0.076	9.66	‡	‡	‡	‡	‡
TransLab [41]	0.920	81.97	0.899	0.095	8.98	0.886	74.23	0.837	0.089	10.40	0.927	87.63	‡	0.063	5.46
MirrorNet [12]	‡	85.10	0.903	0.083	7.67	‡	74.20	0.828	0.090	10.76	‡	‡	‡	‡	‡
PMDNet [13]	‡	87.00	0.930	0.067	6.17	‡	81.70	0.890	0.061	6.74	‡	‡	‡	‡	‡
EBLNet [67]	0.944	87.70	0.922	0.064	6.08	0.919	81.70	0.878	0.069	6.75	0.939	89.58	‡	0.052	4.60
GSDNet [68]	0.949	88.07	0.932	0.059	5.71	**0.931**	83.67	0.903	0.055	6.12	‡	‡	‡	‡	‡
RFENet [50]	‡	87.40	0.929	0.062	5.79	‡	83.60	0.904	0.049	6.24	0.965	**91.25**	‡	0.043	3.68
CMCM [51]	‡	88.30	0.933	0.059	5.65	‡	84.90	0.912	0.050	6.02	‡	87.80	0.878	0.046	3.55
Ours	**0.955**	**88.70**	**0.945**	**0.051**	**5.08**	0.925	**87.10**	**0.927**	**0.048**	**5.55**	**0.972**	90.30	**0.895**	**0.042**	**3.13**

(b) Performance comparison of camouflage benchmarks.

Methods	CHAMELEON [69]				CAMO [70]				COD10K [71]				NC4K [72]			
	S_m↑	E_m↑	F_β^w↑	MAE↓	S_m↑	E_m↑	F_β^w↑	MAE↓	S_m↑	F_β^w↑	E_m↑	MAE↓	S_m↑	E_m↑	F_β^w↑	MAE↓
MirrorNet [73]	‡	‡	‡	‡	0.741	0.804	0.652	0.100	‡	‡	‡	‡	‡	‡	‡	‡
PraNet [74]	0.860	0.898	0.763	0.044	0.769	0.833	0.663	0.094	0.789	0.839	0.629	0.045	0.822	0.876	0.724	0.059
SINet [71]	0.872	0.946	0.806	0.034	0.751	0.771	0.606	0.100	0.771	0.806	0.551	0.051	0.810	0.873	0.772	0.057
MGL [75]	0.893	0.923	0.813	0.030	0.775	0.847	0.673	0.088	0.814	0.865	0.666	0.035	‡	‡	‡	‡
PFNet [76]	0.882	0.931	0.810	0.033	0.782	0.852	0.695	0.085	0.800	0.868	0.660	0.040	0.829	0.887	0.745	0.053
UGTR [77]	0.888	0.940	0.794	0.031	0.785	0.859	0.686	0.086	0.818	0.850	0.667	0.035	‡	‡	‡	‡
LSR [72]	0.893	0.938	0.839	0.033	0.793	0.826	0.725	0.085	0.793	0.868	0.685	0.041	0.839	0.883	0.779	0.053
PreyNet [78]	0.895	0.951	0.844	0.028	0.790	0.842	0.708	0.077	0.813	0.891	0.697	0.034	‡	‡	‡	‡
SINet-V2 [59]	0.888	0.942	0.816	0.030	0.820	0.882	0.743	0.070	0.815	0.887	0.680	0.037	0.847	0.903	0.769	0.048
ZoomNet [79]	0.902	0.958	**0.845**	**0.023**	0.820	0.892	0.752	0.066	0.838	0.888	0.729	0.029	0.853	0.896	0.784	0.043
DTINet [80]	‡	‡	‡	‡	0.856	0.916	0.796	0.050	0.824	0.896	0.695	0.034	0.863	0.917	0.792	0.041
FEDER [58]	0.907	0.964	‡	0.025	0.807	0.873	‡	0.069	0.823	0.900	‡	0.032	0.846	0.905	‡	0.045
EVP [36]	0.871	0.917	0.795	0.036	0.846	0.895	0.777	0.067	0.843	0.907	0.742	0.032	0.874	‡	‡	‡
VSCode [34]	‡	‡	‡	‡	0.836	0.892	0.768	0.060	0.847	0.913	0.744	0.028	0.874	0.920	0.813	0.038
DSAM [22]	‡	‡	‡	‡	0.832	0.913	0.794	0.061	0.846	0.921	0.760	0.033	0.871	0.932	0.826	0.040
Ours	**0.918**	**0.971**	0.840	0.024	**0.875**	**0.936**	**0.817**	**0.046**	**0.877**	**0.938**	**0.793**	**0.024**	**0.895**	**0.942**	**0.847**	**0.032**

(c) Performance comparison of underwater benchmarks.

Methods	MAS3K [81]				RMAS [82]				UFO120 [83]				RUWI [84]			
	S_m↑	E_m↑	F_β^w↑	MAE↓	S_m↑	E_m↑	F_β^w↑	MAE↓	S_m↑	E_m↑	F_β^w↑	MAE↓	S_m↑	E_m↑	F_β^w↑	MAE↓
SINet [71]	0.820	0.884	0.725	0.039	0.835	0.908	0.780	0.025	0.837	0.890	0.834	0.079	0.789	0.825	0.872	0.096
PFNet [76]	0.839	0.890	0.746	0.039	0.843	0.922	0.771	0.026	0.708	0.683	0.550	0.216	0.883	0.870	0.790	0.062
RankNet [72]	0.812	0.867	0.722	0.043	0.846	0.927	0.772	0.026	0.823	0.828	0.772	0.101	0.886	0.889	0.759	0.056
C2FNet [85]	0.851	0.894	0.761	0.038	0.858	0.923	0.788	0.026	0.826	0.878	0.806	0.083	0.830	0.883	0.924	0.060
ECDNet [86]	0.850	0.901	0.766	0.036	0.823	0.854	0.689	0.036	0.783	0.848	0.768	0.103	0.812	0.871	0.917	0.064
OCENet [87]	0.824	0.868	0.703	0.052	0.836	0.900	0.752	0.030	0.725	0.773	0.668	0.161	0.791	0.798	0.863	0.115
ZoomNet [79]	0.862	0.898	0.780	0.032	0.855	0.915	0.795	0.022	0.702	0.815	0.670	0.174	0.753	0.771	0.817	0.137
MASNet [82]	0.864	0.906	0.788	0.032	0.862	0.920	0.801	0.024	0.827	0.879	0.820	0.083	0.880	0.913	0.944	0.047
SETR [32]	0.855	0.917	0.789	0.030	0.818	0.933	0.747	0.028	0.811	0.871	0.796	0.089	0.864	0.895	0.924	0.055
TransUNet [88]	0.861	0.919	0.805	0.029	0.832	0.941	0.776	0.025	0.825	0.888	0.827	0.079	0.872	0.910	0.940	0.048
H2Former [89]	0.865	0.925	0.810	0.028	0.844	0.931	0.799	0.023	0.844	0.901	0.845	0.070	0.884	0.919	0.945	0.045
SAM [6]	0.763	0.807	0.656	0.059	0.697	0.790	0.534	0.053	0.768	0.827	0.745	0.121	0.855	0.907	0.929	0.057
SAM-Ad [21]	0.847	0.914	0.782	0.033	0.816	0.927	0.752	0.027	0.829	0.884	0.834	0.081	0.878	0.913	0.946	0.046
Dual-SAM [23]	0.884	0.933	0.838	0.023	0.860	0.944	0.812	0.022	0.856	0.914	0.864	0.064	**0.903**	0.939	0.959	0.035
MAS-SAM [90]	0.887	0.938	0.840	0.025	0.865	0.948	0.819	0.021	0.861	0.914	0.864	0.063	0.894	0.941	**0.961**	0.035
Ours	**0.912**	**0.957**	**0.868**	**0.019**	**0.884**	**0.960**	**0.839**	**0.017**	**0.875**	**0.928**	**0.880**	**0.056**	0.899	**0.947**	0.957	**0.032**

A.4 Experiments

In Figure 4, before integrating the HMDE, although the model could perceive mirror regions well, it also exhibits high responses to non-mirror areas. Upon the introduction of the HMDE, we further enhance the representations of mirror regions and attenuate non-mirror areas. We attribute this to the expert grouping and directional synergy mechanism. In Figure 7 (a), we visualize the feature distributions before and after applying the SPPC. After using the SPPC, the features become more compact internally, and the inter-class distances increase. In Figure 7 (b), we provide visual comparisons of predictions in more diverse scenarios. In Table 9, we validate the generalization of the proposed method in transparent surface, camouflage and underwater scenarios. Our method yields promising results. We analyze that 1) Depth difference between transparent and opaque areas is significant, resembling camouflage and underwater scenarios; 2) Dynamically selecting based on input is effective for modeling complex visual spaces; 3) Distribution-based sample selection and pixel-prototype collaboration can maximize the utilization of intra-sample features, especially for tasks with limited data scale.

A.5 Limitation and Future Work

In Figure 7 (c), our method still needs improvement in detecting fine-grained regions, mixed multiple and small objects. One possible reason is that the original SAM relies solely on high-level semantic feature for decoding. In the future, we will alleviate the aforementioned issues through uncertainty modeling and high-order fusion, and deploy the model to mobile devices.