

SPDA-SAM: A Self-prompted Depth-Aware Segment Anything Model for Instance Segmentation

Yihan Shang, Wei Wang, Chao Huang, *Member, IEEE*, Junyu Dong, *Member, IEEE* and Xinghui Dong, *Member, IEEE*

Abstract—Recently, Segment Anything Model (SAM) has demonstrated strong generalizability in various instance segmentation tasks. However, its performance is severely dependent on the quality of manual prompts. In addition, the RGB images that instance segmentation methods normally use inherently lack depth information. As a result, the ability of these methods to perceive spatial structures and delineate object boundaries is hindered. To address these challenges, we propose a Self-prompted Depth-Aware SAM (SPDA-SAM)¹ for instance segmentation. Specifically, we design a Semantic-Spatial Self-prompt Module (SSSPM) which extracts the semantic and spatial prompts from the image encoder and the mask decoder of SAM, respectively. Furthermore, we introduce a Coarse-to-Fine RGB-D Fusion Module (C2FFM), in which the features extracted from a monocular RGB image and the depth map estimated from it are fused. In particular, the structural information in the depth map is used to provide coarse-grained guidance to feature fusion, while local variations in depth are encoded in order to fuse fine-grained feature representations. To our knowledge, SAM has not been explored in such self-prompted and depth-aware manners. Experimental results demonstrate that our SPDA-SAM outperforms its state-of-the-art counterparts across eleven different data sets. These promising results should be due to the guidance of the self-prompts and the compensation for the spatial information loss by the coarse-to-fine RGB-D fusion operation.

Index Terms—Segment Anything Model, Instance Segmentation, Image Segmentation, Self-prompt, RGB-D Fusion.

I. INTRODUCTION

As a fundamental computer vision task, Instance Segmentation aims at accurately identifying and delineating each individual object within an image. It has been widely applied in various domains, such as autonomous driving [1], medical image analysis [2], and remote sensing detection [3], [4]. However, the existing instance segmentation models, which were trained using a relatively small data set, usually encounter challenges when applied to a different data set because of the domain-shift issue [5].

Recently, Segment Anything Model (SAM) [6] has demonstrated strong generalizability across different domains, due

This study was supported in part by the National Natural Science Foundation of China (NSFC) (No. 42576200) and in part by the Key Research and Development Program of Shandong Province, China (No. 2024ZLGX06) (Corresponding author: Xinghui Dong).

Y. Shang, J. Dong and X. Dong are with the State Key Laboratory of Physical Oceanography and the Faculty of Information Science and Engineering, Ocean University of China, Qingdao, 266100. (e-mail: shangyihan@stu.ouc.edu.cn, dongjunyu@ouc.edu.cn, xinghui.dong@ouc.edu.cn).

W. Wang and C. Huang are with the School of Cyber Science and Technology, Shenzhen Campus of Sun Yat-sen University, Shenzhen, 518107. (e-mail: wangwei29@mail.sysu.edu.cn; huangch253@mail.sysu.edu.cn).

¹The source code and models are available at <https://github.com/INDTLab/SPDA-SAM>.

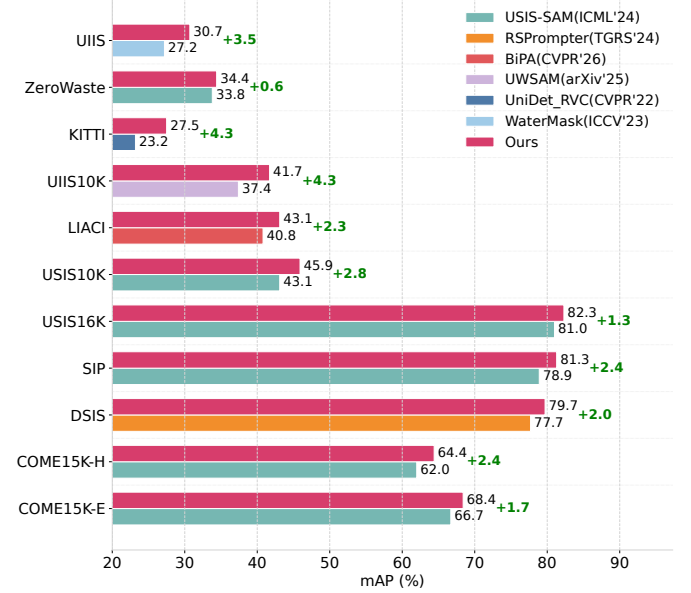


Fig. 1. Comparison between five state-of-the-art methods and our SPDA-SAM across 11 instance segmentation data sets.

to the large-scale data set, i.e., SA-1B [6] on which it was pre-trained. As a representative promptable model, SAM has shown impressive performance in various instance segmentation tasks and has been increasingly adopted in diverse downstream applications [7]–[10]. However, the performance of it is affected by the quality of prompts [11]. This dependency not only increases the cost of human intervention, but also limits the deployment and automaticity of the model [12] in practical scenarios.

In addition, depth information is useful for instance segmentation because it encodes rich geometric cues which represent object shapes, spatial positions, and relative sizes in the three-dimensional space. However, the monocular RGB images that existing instance segmentation methods normally utilize inevitably lose depth information during the imaging procedure, which impairs the ability of these methods to understand spatial structures and object boundaries.

To address the aforementioned challenges, we propose a Self-prompted Depth-Aware SAM, referred to as SPDA-SAM, for instance segmentation. This network eliminates the manual prompts provided by humans and incorporates estimated depth priors into image representations. For the purpose of adapting to a specific domain, the SPDA-SAM is fine-tuned using LoRA [13] with a relatively small data set.

To be specific, we first design a Semantic-Spatial Self-prompt Module (SSSPM). Here, "Self-prompt" means that this module generates prompts from the internal representations of the model by extracting semantic and spatial prompts from the image encoder and mask decoder of SAM, respectively, rather than relying on human intervention or independently learnable query embeddings. The semantic prompts capture semantic representations to utilize high-level contextual information, while the spatial prompts leverage the sensitivity of the mask decoder to object boundaries to enhance the detail modeling ability. As a result, the adaptive segmentation guidance and autonomous perception capabilities of the model can be obtained.

We then introduce a Coarse-to-Fine RGB-D Fusion Module (C2FFM) in which the features extracted from the geometric priors that we estimate using a pre-trained monocular depth estimation model are fused with those extracted from an RGB image. Contours extracted from the estimated depth priors are used to guide the attention distribution at the coarse level. On the other hand, local depth gradients are used to encode boundary details at the fine level, which enable the more precise instance segmentation results to be produced. The entire network can be end-to-end trained without human intervention.

To our knowledge, SAM has not been explored in such self-prompted and depth-aware manners. Our main contributions can be summarized as follows.

- We introduce a Self-prompted Depth-Aware SAM, i.e., SPDA-SAM, for the instance segmentation task. Due to the powerful generalizability of SAM, our network can adapt to a novel scenario by fine-tuning it using a small number of images. The SPDA-SAM gets rid of the user-input prompts and injects estimated depth priors into image representations. As shown in Fig. 1, our method always outperforms the state-of-the-art baselines on each of the eleven data sets.
- We design a Semantic-Spatial Self-prompt Module (SSSPM), which exploits both the semantic and spatial prompts generated automatically. Therefore, human intervention is not required.
- We propose a Coarse-to-Fine RGB-D Fusion Module (C2FFM) which progressively fuses the features extracted from the depth priors that we estimate from a single RGB image and those extracted from this image. As a result, the discriminatory power of the encoder becomes stronger.
- We conduct comprehensive evaluations on eleven publicly available data sets. The results provide the community with a set of benchmarks.

The remainder of this paper is organized as follows. In Section II, we review the related work. The proposed SPDA-SAM method is described in Section III. In Section IV, our experimental setup and results and the ablation study are reported. Finally, we draw our conclusion in Section V.

II. RELATED WORK

A. Instance Segmentation

Instance segmentation requires the model not only to distinguish between different object categories, but also to differentiate individual instances within the same category. Early approaches were usually adopted on top of a two-stage framework, such as Mask R-CNN [14], Cascade Mask R-CNN [15] and HTC [16]. These approaches typically followed a two-step paradigm, which first generates region proposals and then performs mask prediction. Although those approaches achieved a high accuracy, they often underwent heavy computational overhead, which limited their applicability in real-time or resource-constrained scenarios. Moreover, the reliance on hand-crafted region proposal generators further impaired the adaptability of the model to novel domains.

To improve efficiency and performance, one-stage instance segmentation methods were subsequently developed, for example, YOLACT [17], BlendMask [18], CondInst [19], EmbedMask [20] and SOLOv2 [21]. These methods directly generated masks without relying on region proposals, which significantly accelerated the inference process. However, they often encountered challenges in handling overlapping instances and complex object shapes, which might lead to low segmentation accuracy in densely populated or cluttered scenes. Recently, Transformer-based approaches, e.g., Mask2Former [22], were developed on top of attention mechanisms, because they leveraged global context information to better handle complex scenes and improve the discrimination between adjacent instances.

However, existing methods often depended heavily on the distributions of training data and tended to suffer performance degradation when applied to complex or domain-specific images, e.g., underwater scenarios, medical imagery and remote sensing images. This issue hinders the deployment of them in practical applications and has motivated a growing interest in foundational vision models. Therefore, we were motivated to perform instance segmentation tasks on top of such a model.

B. Segment Anything Models

As a foundation model in computer vision, Segment Anything Model (SAM) [6] has demonstrated remarkable generalizability in class-agnostic segmentation on natural images, where segmentation masks are produced through a prompt-driven mechanism. Since its release, the SAM family has continued to evolve. In particular, SAM 2 [23] extended promptable segmentation from static images to videos by introducing a streaming memory mechanism that propagates segmentation information across frames, while SAM 3 [24] further advanced this direction through Promptable Concept Segmentation (PCS), which aims to segment all instances matching a given conceptual prompt. More studies have adapted SAM to specialized downstream tasks, which span medical imaging [25], remote sensing [7], and underwater imagery [26]. In addition, parameter-efficient adaptation methods, such as SAM-Adapter [12], inserted lightweight adapters into the frozen SAM encoder for downstream tasks.

However, SAM is highly sensitive to the quality of its external prompts, as imprecise or ambiguous prompts can substantially degrade the resulting masks. To reduce such prompt dependence, some studies focused on automating prompt generation, including RSPrompter [7], USIS-SAM [26], and BiPA [27], which eliminate the need for manual prompt input. Moreover, SAM operates solely on the RGB appearance and is unable to exploit depth or geometric cues that are valuable for separating adjacent objects and delineating boundaries. To compensate for this limitation, multi-modal extensions, such as SAM-DAQ [28], have explored incorporating depth cues into SAM for geometry-aware segmentation. These efforts have substantially broadened the applicability of SAM. Nevertheless, generation of prompts in a closed loop from the representations of the model and progressive fusion of geometric priors with RGB features across encoder stages remain open questions, which motivated our work.

C. Prompt Learning

In recent years, a new learning paradigm, i.e., pre-training and prompting, has gradually become mainstream with the emergence of foundation models [29]. This paradigm eliminates the need for fine-tuning by reformulating downstream tasks through the design of input prompts, thereby aligning them more closely with the pre-training objectives. As a representative technique, Contrastive Language-Image Pre-training (CLIP) [30] leveraged textual prompts for cross-modal understanding [31]. On the other hand, Kirillov et al. [6] made the first effort on developing a foundation model for image segmentation, i.e., SAM, on top of the prompt-based mechanism. However, the prompts of SAM require human intervention.

Efforts have been made to automate the prompt generation process for SAM. RSPrompter [7] introduced learnable prompt embeddings for remote sensing instance segmentation. USIS-SAM [26] designed a Salient Feature Prompt Generator (SFPG) to automatically produce dense prompts for underwater instance segmentation. In [27], BiPA formulated dense prompt learning as a bilevel optimization problem, in which the upper level optimizes a learnable dense prompt embedding via Bayesian optimization on the validation set, while the lower level updates the model parameters on the training set. A separately generated sparse prompt further refines mask boundaries. Although these methods reduce the reliance on manual prompts, their prompts are produced by separate feed-forward modules or learnable external parameters, rather than from the predictions of the model. In other words, they lack a closed-loop mechanism in which the coarse segmentation result of the decoder is fed back to generate spatial prompts that guide a second refinement pass.

In contrast, we introduced a Semantic-Spatial Self-prompt Module (SSSPM), which utilizes both the semantic and spatial prompts automatically generated from the intermediate layers of the encoder and the coarse segmentation result of the decoder, respectively. As a result, the prompts can be dynamically generated from the internal state of the model and are inherently content-adaptive.

D. RGB-D Based Image Segmentation

In contrast to the RGB images that image segmentation methods normally utilize, the joint use of RGB images and depth maps (i.e., RGB-D) allows for exploitation of the spatial characteristics encoded in the depth maps. Previous studies have made progress by incorporating depth characteristics into feature representations [32]. However, effective feature extraction from depth maps and fusion with RGB image features remain challenges. This dilemma should be attributed to the significant difference between depth maps and RGB images in terms of data distributions, numerical range, and semantic representation.

Depth-aware convolutions, such as depth-aware CNNs [33], have been introduced. They incorporated specially designed convolutional kernels or attention mechanisms to enable the network to perceive and exploit depth information. On the other hand, dual-branch encoders [34] were proposed, in which RGB images and depth maps were processed by two individual encoders, to extract modality-specific features. The two sets of features were then fused [32]. However, these methods usually relied on the simple concatenation or shallow fusion operations. As a result, they failed to fully exploit the structural semantics embedded in the depth maps.

With the popularity of SAM, several recent studies have explored depth-aware variants of SAM. In [35], DSAM addressed camouflaged object detection by injecting depth features into RGB features via knowledge distillation and integrating them into the box prompt. SAM-DAQ [28] inserted depth-guided parallel adapters into each stage of the Hiera encoder in a skip-connection manner, performing stage-wise RGB-D feature fusion within each encoder stage.

Unlike these depth-aware SAM methods, which conduct depth fusion either at the prompt level or through stage-wise parallel adapters, we designed a C2FFM, which performs a cross-stage coarse-to-fine progressive fusion, where depth contours guide the coarse-level attention at the patch embedding stage while local depth gradients direct fine-grained feature alignment across subsequent transformer stages. Although DSAM [35] also employs estimated depth priors, it uses knowledge distillation to inject depth data. In contrast, our method directly fuses estimated depth priors with RGB features via an explicit cross-attention in the dual-path encoder.

III. METHODOLOGY

To alleviate the reliance of SAM [6] on manual prompts and address the absence of depth information in instance segmentation, we propose a Self-prompted Depth-Aware SAM, namely, SPDA-SAM. Due to the excellent generalizability of SAM, this network can be used for different instance segmentation tasks. The architecture of the SPDA-SAM is shown in Fig. 2. As can be seen, our SPDA-SAM contains a Depth-Aware Dual-Path Encoder and a Self-prompted Decoder. The former includes an RGB Encoding Path, a Depth Encoding Path and a Coarse-to-Fine RGB-D Fusion Module (C2FFM), while the latter comprises a Mask Decoder and a Semantic-Spatial Self-prompt Module (SSSPM).

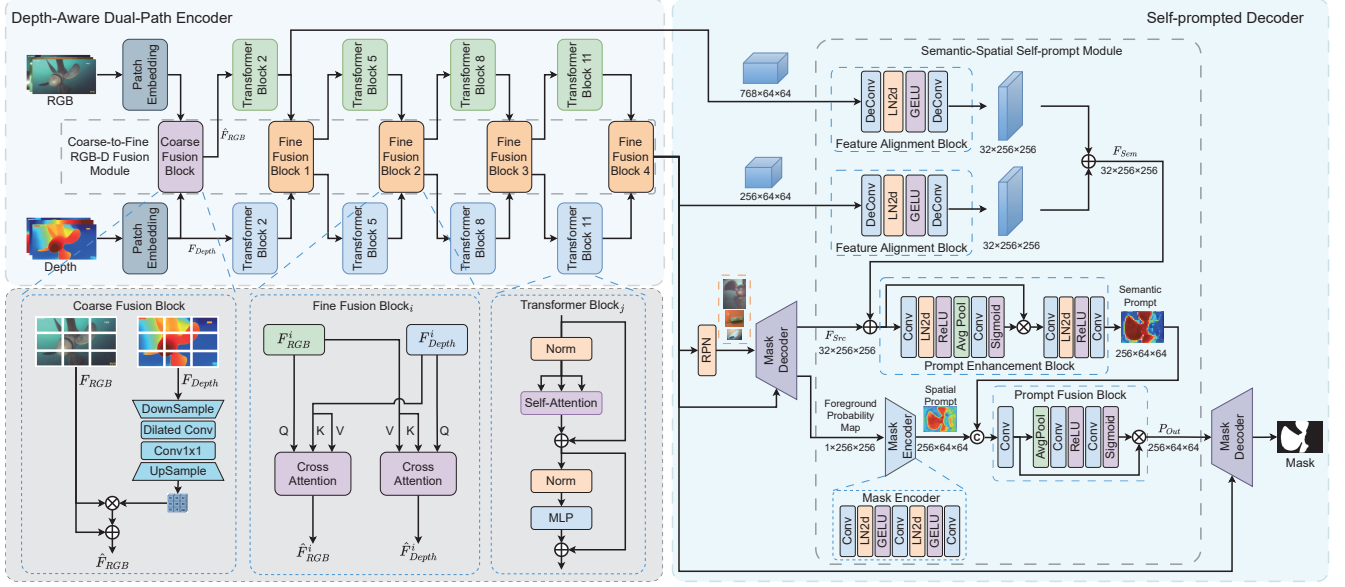


Fig. 2. The architecture of the Self-prompted Depth-Aware SAM (SPDA-SAM), which consists of a depth-aware dual-path encoder and a self-prompted decoder. The encoder contains an RGB encoding path, a depth encoding path and a Coarse-to-Fine RGB-D Fusion Module (C2FFM). The decoder includes a mask decoder and a Semantic-Spatial Self-prompt Module (SSSPM).

A. Depth-Aware Dual-Path Encoder

Within the encoder, the RGB encoding path and the depth encoding path are built on top of the backbone of SAM [6]. The C2FFM include a Coarse Fusion Block and four Fine Fusion Blocks. The former is applied to the output of the Patch Embedding Block, while the latter are applied to the output of the 2nd, 5th, 8th and 11th Transformer Blocks individually.

1) *RGB Encoding Path*: The RGB Encoding Path is initialized using the weights of the backbone of SAM. This path is used to extract features from RGB images. To adapt the RGB encoding path to a new scenario, we use LoRA [13] to fine-tune it.

2) *Depth Encoding Path*: This path is also initialized using the weights of the backbone of SAM, which aims to extract features from the estimated depth priors. LoRA [13] is also used to fine-tune the depth encoding path.

3) *C2FFM*: The C2FFM progressively performs the coarse-to-fine fusion operation on the features extracted from both the RGB image and the estimated depth priors across different stages, enabling effective utilization of the complementary information encoded in the two sets of data. A Coarse Fusion Block and four Fine Fusion Blocks are comprised of the C2FFM.

Coarse Fusion Block The Coarse Fusion Block aims to integrate depth-guided structural features into RGB features at a coarse resolution, enabling the fusion of complementary modalities to enhance global context and boundary-aware representation. To achieve this goal, we first apply a down-sampling operation, a dilated convolution and a Relu activation function to the patch embeddings of the estimated depth priors F_{Depth} , enlarging the receptive field and extracting multi-scale contextual information. Subsequently, a 1×1 convolution is used to reduce the dimensionality of channels and fuse the features, followed by an upsampling operation to restore the

resolution to match the feature maps F_{RGB} extracted from the RGB image.

The resultant feature maps are passed through a Sigmoid activation function and are then used to modulate F_{RGB} through an element-wise multiplication operation. This operation highlights the regions with strong depth cues. A residual connection further fuses the original RGB features, yielding the fused output $\hat{F}_{RGB}$. The computation of the Coarse Fusion Block can be expressed as:

$$F_{Dilated} = DilConv(Down(F_{Depth})), \quad (1)$$

$$Attn = Sigmoid(Up(Conv_{1 \times 1}(F_{Dilated}))), \quad (2)$$

$$\hat{F}_{RGB} = F_{RGB} + Attn \odot F_{RGB}, \quad (3)$$

where $\sigma(\cdot)$ is the Sigmoid activation function, $\varphi(\cdot)$ denotes a sequence of downsampling, dilated convolution, and ReLU activation applied to the depth feature maps, and $\odot$ represents the element-wise multiplication operation.

Fine Fusion Block This block is used to exploit the complementary actions of multi-modal features in local details. In total, four Fine Fusion Blocks are applied to the output of the 2nd, 5th, 8th and 11th Transformer Blocks, respectively. This hierarchical design enables the network to align multi-modal information across different levels of semantic abstraction. As a result, the network captures the complementary relationship between local fine-grained details and global contextual semantics. Regarding a Fine Fusion Block i , we first flatten the input feature maps F_{RGB}^i and F_{Depth}^i into sequential representations:

$$S_{RGB}^i = Flatten(F_{RGB}^i) \in \mathbb{R}^{B \times N \times C}, \quad (4)$$

$$S_{Depth}^i = Flatten(F_{Depth}^i) \in \mathbb{R}^{B \times N \times C}, \quad (5)$$

where $N = H \times W$ represents the length of the sequences.

Two cross-attention blocks are applied, in which S_{RGB}^i and S_{Depth}^i are used as the query, respectively. Taking the case where S_{RGB}^i is used as the query and S_{Depth}^i is utilized as the key and value, we can obtain $\hat{F}_{RGB}^i$. This process can be formulated as:

$$Q, K, V = W_Q S_{RGB}^i, W_K S_{Depth}^i, W_V S_{Depth}^i, \quad (6)$$

$$\hat{F}_{RGB}^i = Reshape(Softmax(\frac{QK^T}{\sqrt{d_k}}V)), \quad (7)$$

where $Reshape(\cdot)$ denotes the operation which transforms the sequence back to its original shape. Similarly, we obtain a set of feature maps $\hat{F}_{Depth}^i$.

B. Self-prompted Decoder

Compared with the mask decoder of the original SAM [6] which requires external prompts, we propose a Self-prompted Decoder. This decoder contains a mask decoder and an SSSPM. The self-prompted decoder leverages both the internal visual features extracted using the depth-aware dual-path encoder and the mask that the mask decoder produces to generate self-guided prompts, which are able to direct the segmentation process. The proposed self-prompt mechanism enables accurate object localization and segmentation without human intervention.

1) *Mask Decoder*: The mask decoder, which has the same structure as that of SAM but is randomly initialized, is used twice. At the first time, it receives the initial prompts generated by the Region Proposal Network (RPN) [36] and the output of the last fine fusion block. The result is a coarse segmentation mask. This process roughly localizes the target region. At the second time, the mask decoder takes the semantically and spatially enriched prompts that the proposed SSSPM produces and the output of the last fine fusion block. As a result, the mask boundaries and structural details are further refined.

2) *SSSPM*: The SSSPM aims to automatically generate high-quality prompts without human intervention. The prompts comprises semantic and spatial prompts, which are visualized in Fig. 2 by computing the dot-product similarity between the center vector and surrounding vectors. The SSSPM consists of four main components, including a Feature Alignment Block, a Mask Encoder, a Prompt Enhancement Block and a Prompt Fusion Block.

Feature Alignment Block This block facilitates the integration of multi-level semantic cues. We select two representative layers from the depth-aware dual-path encoder, including a lower-level block (i.e., Transformer Block 2) which captures local details and a higher-level block (i.e., Fine Fusion Block 4) which encodes high-level abstract semantics. The two sets of features produced by both layers differ significantly in resolution and channel dimensions. To achieve an effective fusion operation, each set of features is passed through a Feature Alignment Block, which consists of a deconvolution [37], a layer normalization layer, a GELU activation function and a second deconvolution. This block spatially upsamples and channel-aligns the two sets of features and transforms them into a unified resolution and embedding space. The two sets of resultant features are fused via an element-wise addition

operation. The result is a set of semantic features F_{Sem} , which contain rich semantic information and provide strong guidance for the subsequent prompt enhancement block.

Prompt Enhancement Block The Prompt Enhancement Block aims to enrich the features F_{Src} that the mask decoder produces, which have already been integrated with the initial prompt information. We fuse F_{Src} with the semantic features F_{Sem} via an element-wise addition operation. The fused features are then passed through the prompt enhancement block, which contains two convolutional layers, together with a layer normalization operation, a ReLU activation function and a global average pooling operation. A Sigmoid activation is used to obtain an attention map, which modulates the fused features. Finally, two convolutional layers are used to reduce the spatial size and increase the channel dimension. As a result, the semantic prompt is derived.

Mask Encoder The Mask Encoder receives a coarse foreground probability map that the mask decoder generates. The map is transformed into dense embeddings, which have been spatially aligned with the image embeddings. The mask encoder contains two strided 2×2 convolutional layers. Each layer is followed by a layer normalization operation and a GELU activation function. A 1×1 convolution is finally used to perform the dimensional projection operation. The resultant dense embeddings encode spatial priors and serve as the spatial prompt $P_{Spatial}$.

Prompt Fusion Block This block is designed to exploit the complementary relationship between the spatial prompt $P_{Spatial}$ and the semantic prompt $P_{Semantic}$. Specifically, they are first concatenated along the channel dimension, followed by a convolutional layer, to obtain a fused prompt P_{Fused} . For the sake of extracting global contextual information from P_{Fused} , we apply a global average pooling operation to it. This process summarizes the response of each channel, which facilitates the generation of an attention map in order to selectively emphasize informative channels. The pooled features are passed through two convolutional layers with a ReLU activation function in between, followed by a Sigmoid activation function. The result is an attention map $Attn$, which is then used to modulate P_{Fused} via an element-wise multiplication operation.

The final prompt that the SSSPM generates, P_{Out} , can be formulated as:

$$P_{Fused} = Conv(Concat(P_{Semantic}, P_{Spatial})), \quad (8)$$

$$Attn = \sigma(Conv(ReLU(Conv(GAP(P_{Fused}))))), \quad (9)$$

$$P_{Out} = P_{Fused} \odot Attn, \quad (10)$$

where $Concat(\cdot)$ denotes the channel-wise concatenation operation, $GAP(\cdot)$ stands for the global average pooling operation, and $\sigma(\cdot)$ represents the Sigmoid activation function. P_{Out} is fed into the mask decoder, together with the output of the last fine fusion block. Here, P_{Out} serves as a dense prompt in the form of $256 \times 64 \times 64$ feature maps, which are spatially aligned with the image embeddings. This process allows the mask decoder to more accurately distinguish between target regions with ambiguous boundaries or high semantic similar-

TABLE I

COMPARISON BETWEEN OUR METHOD AND 25 BASELINES ON THE COME15K-E [38], COME15K-H [38], DSIS [39] AND SIP [40] DATA SETS. THE BEST AND SECOND-BEST RESULTS ARE HIGHLIGHTED IN **BOLD** AND UNDERLINE FONTS, COME15K-H, RESPECTIVELY. THIS CONTINUES WITH FOLLOWING TABLES.

Method	Venue	COME15K-E			COME15K-H			DSIS			SIP		
		mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}
S4Net [41]	CVPR19	43.7	68.0	52.5	37.1	60.9	43.2	58.3	81.9	71.8	49.6	76.0	63.7
RDPNet [42]	TIP21	49.8	72.2	59.5	42.1	65.2	49.7	66.1	87.2	80.1	59.0	80.1	74.1
OQTR [43]	TMM22	48.7	71.1	58.3	42.7	65.9	50.5	63.1	85.9	77.0	59.9	83.1	76.3
OSFormer [44]	ECCV22	53.0	71.9	61.3	45.8	66.3	52.4	67.9	86.3	78.3	63.2	80.5	74.6
Mask R-CNN [14]	ICCV17	48.8	71.2	58.6	42.2	65.7	50.8	65.6	86.8	80.2	57.9	79.8	73.3
Mask Scoring R-CNN [45]	CVPR19	49.7	70.4	58.9	42.3	63.8	50.0	66.8	87.3	80.3	60.0	79.8	73.3
Cascade R-CNN [15]	TPAMI19	49.4	69.8	58.7	41.9	64.3	49.8	66.4	87.3	79.9	59.1	79.1	73.6
CenterMask [46]	CVPR20	49.6	71.6	59.0	42.5	65.2	51.0	65.7	87.6	79.7	57.6	79.8	72.3
HTC [16]	CVPR19	51.4	73.7	61.0	44.1	68.0	52.1	67.5	87.6	81.1	60.0	81.4	74.9
YOLACT [17]	ICCV19	48.1	70.7	56.2	41.4	66.0	48.5	62.4	84.1	73.3	62.0	82.3	74.7
BlendMask [18]	CVPR20	48.1	70.6	56.9	41.0	64.8	48.5	65.5	86.9	78.0	55.5	77.3	69.7
CondInst [19]	ECCV20	49.6	72.0	59.5	42.8	66.4	51.0	65.1	86.8	79.1	58.6	79.4	73.3
SOLOv2 [21]	TPAMI21	51.1	71.6	59.9	45.1	66.3	52.8	67.4	87.0	80.4	63.4	80.7	74.8
QueryInst [47]	CVPR21	51.5	73.1	61.1	43.9	67.5	51.8	67.9	87.5	80.6	61.3	81.0	75.1
SOTR [48]	ICCV21	50.7	70.4	59.0	43.5	65.1	50.3	68.2	86.9	79.3	61.6	78.4	73.0
Mask Transfomer [49]	CVPR22	48.7	71.0	56.3	40.7	64.5	47.2	67.5	87.6	80.4	57.8	78.9	70.3
SparseInst [50]	CVPR22	51.3	71.9	58.9	43.1	65.1	48.5	65.0	85.0	75.7	62.8	81.3	75.3
Mask2Former [22]	CVPR22	51.5	67.3	57.6	44.1	62.4	49.4	68.3	85.1	76.7	66.6	79.3	75.1
PolySnake [51]	TCSVT24	48.2	66.8	58.9	42.6	62.1	50.4	64.9	84.2	74.9	57.4	77.6	72.0
CalibNet [39]	TIP24	58.0	75.8	65.6	50.7	70.4	57.3	69.3	87.8	81.6	72.1	86.6	82.9
Diveseg [52]	AAAI26	61.5	75.2	63.9	55.4	70.8	57.4	76.7	88.9	83.0	76.9	87.0	81.0
BiPA [27]	CVPR26	63.3	80.5	67.7	57.2	76.3	60.7	71.5	87.8	79.5	77.9	88.1	84.1
RSPrompter [7]	TGRS24	65.0	81.6	<u>70.8</u>	59.9	78.6	65.0	<u>77.7</u>	<u>92.5</u>	86.6	78.8	<u>90.1</u>	<u>87.1</u>
USIS-SAM [26]	ICML24	<u>66.7</u>	<u>81.7</u>	70.6	62.0	<u>78.7</u>	<u>65.5</u>	68.6	84.3	74.8	<u>78.9</u>	89.3	85.0
SAMSnake [53]	NN25	58.6	70.7	61.3	53.8	64.9	54.9	64.4	74.4	62.4	72.1	84.4	80.0
SAM [6]	ICCV23	62.5	80.5	66.6	56.5	76.0	59.8	68.0	83.2	74.5	76.0	87.4	82.3
SPDA-SAM (Ours)	-	68.4	84.6	74.4	64.4	82.1	69.3	79.7	93.8	87.4	81.3	91.6	88.9

ity. Consequently, the accuracy of instance segmentation can be improved.

C. Loss Function

In addition to the loss functions of Region Proposal Network (RPN) [36], which include the object classification and bounding box regression loss functions, we apply two pixel-wise cross-entropy loss functions to supervise both the coarse and refined masks generated by applying the mask decoder twice. Finally, the combined loss function $\mathcal{L}$ can be written as

$$\mathcal{L} = \lambda_1 \cdot \mathcal{L}_{RPN} + \lambda_2 \cdot \mathcal{L}_{Coarse} + \lambda_3 \cdot \mathcal{L}_{Refined}, \quad (11)$$

where $\mathcal{L}_{RPN}$ denotes the RPN loss, and $\mathcal{L}_{Coarse}$ and $\mathcal{L}_{Refined}$ stands for the cross-entropy losses used for the coarse and refined masks, respectively. In our experiments, all the three weights were set to 1 (i.e., $\lambda_1 = \lambda_2 = \lambda_3 = 1$). This multi-stage supervision strategy ensures the consistency between coarse localization and final segmentation, which improves the quality of the final mask.

IV. EXPERIMENTS

In this section, we first describe the setup used in our experiments. Then we report the results obtained in the main experiments and ablation study.

A. Experimental Setup

Here, we introduce the data sets and evaluation metrics that we used and the implementation details of the experiments.

1) *Data Sets*: We evaluated the proposed method on eleven publicly available data sets. Specifically, six terrestrial data sets, including COME15K-E [38], COME15K-H [38], DSIS [39], SIP [40], KITTI [54] and ZeroWaste [55], and five UIIS data sets, including LIACI [56], UIIS [57], USIS10K [26], UIIS10K [58] and USIS16K [59], were used. We always followed the original splitting for each data set, in which the data set was divided into training, validation and test sets. Regarding the data sets with instance-level annotations, we directly used them as the ground-truth data. In terms of the data sets which only contained semantic annotations, we used connected component labeling to convert semantic segmentation masks to instance-level annotations.

The pre-processing strategy proposed in Depth Anything v2 [60] was used to perform monocular depth estimation on RGB images. The estimated depth maps serve as geometric priors that encode relative spatial structure and boundary cues.

2) *Evaluation Metrics*: To evaluate the performance of the proposed method on instance segmentation tasks, we utilized the widely used mean Average Precision (mAP) metric [61],

as well as AP_{50} and AP_{75} , following existing studies.

3) *Implementation Details*: We adopted the ViT-B [62] model in the pre-trained SAM [6] to initialize both the RGB and depth encoding paths from the same pre-trained checkpoint. The two paths were then updated independently via their corresponding LoRA adapters, without weight sharing between them. Specifically, LoRA was applied to the QKV projection layers of each branch with a rank of 16, a scaling factor of 32 and a dropout rate of 0.05, and without a bias term. The proposed method was implemented using PyTorch and the MMDetection framework [63].

During the training process, we used the AdamW optimizer with an initial learning rate of 0.0002 and a weight decay of 0.05. To enable efficient and stable optimization, we designed a two-stage learning rate scheduling strategy. Specifically, we employed a linear warm-up scheme for the first 50 iterations, which gradually increased the learning rate to the target value. Subsequently, we applied a cosine annealing learning rate schedule, to progressively decay the learning rate throughout the training process, which eventually reduced this rate to 0.001 times the base rate. Additionally, we used the Automatic Mixed Precision (AMP) training to accelerate convergence and reduce memory consumption. For all data sets, the batch size was set to 2 during the training process. All models were trained on two NVIDIA RTX 3080Ti GPUs or a single NVIDIA RTX 4090 GPU, while computational complexity analysis was conducted on an NVIDIA RTX 3080Ti GPU.

B. Main Experiments

We tested our method along with different baselines on the 11 data sets. In particular, the vanilla SAM (ViT-Base) was evaluated with bounding box prompts generated by the same RPN as that used in our method. This choice ensures that SAM and our method share an identical initial localization. As a result, the performance difference can be directly attributed to the proposed SSSPM and C2FFM. We will report the results as follows.

1) *COME15K-E, COME15K-H, DSIS and SIP*: We first evaluated the SPDA-SAM, together with 25 baseline approaches, on four salient instance segmentation data sets, including COME15K-E [38], COME15K-H [38], DSIS [39] and SIP [40]. As reported in Table I, the SPDA-SAM always achieved the superior performance to its counterparts. On the COME15K-E data set, our method outperformed the second-best approach by 1.7, 2.9 and 3.6 in terms of the mAP , AP_{50} and AP_{75} metrics, respectively. Regarding the COME15K-H data set, the three values were 2.4, 3.4 and 3.8, respectively. Using the DSIS data set, our method achieved the improvements of 2.0, 1.3 and 0.8 in terms of the three metrics, respectively, compared to the second-best approach. With regard to the SIP data set, the three values were 2.4, 1.5 and 1.8, respectively. These results demonstrate the robustness and generalizability of our SPDA-SAM across different salient instance segmentation scenarios.

2) *UIIS, UIIS10K and USIS16K*: In addition, we evaluated the SPDA-SAM on the UIIS [57], UIIS10K [58] and USIS16K [59] data sets. Our method was also compared with six baselines. It can be seen in Table II, our SPDA-SAM outperformed

the baseline by 3.5, 3.6 and 4.7 in mAP , AP_{50} and AP_{75} , respectively, on the UIIS data set. When the UIIS10K data set was used, the three values were 4.3, 1.7 and 3.9, respectively. In contrast, the three values were 1.3, 1.7 and 1.2 on the USIS16K data set, respectively. The above results demonstrate the better UIIS capability and the stronger robustness of our SPDA-SAM across diverse underwater scenarios, compared to its counterparts.

3) *USIS10K*: The results obtained using the proposed SPDA-SAM and ten baselines on the USIS10K [26] data set are reported in Table III. Following the original benchmark settings, we conducted both multi-class and class-agnostic instance segmentation tasks. Again, our method outperformed its counterparts with regard to each metric. Specifically, our SPDA-SAM achieved the improvements of 4.9, 3.3 and 6.4 in terms of mAP , AP_{50} and AP_{75} , respectively, compared to the best baseline in the class agnostic instance segmentation task. On the other hand, the three values were 2.8, 1.3 and 3.8, respectively, within the multi-class instance segmentation task. Compared to the methods which utilized the ViT-Huge backbone [62], our approach used the more lightweight ViT-Base backbone [62] while still achieving the superior performance.

4) *LIACI*: To comprehensively evaluate the performance of our SPDA-SAM on the LIACI [56] data set, we compared it with fourteen state-of-the-art instance segmentation approaches, including two-stage methods [14] and one-stage methods [21], [19]. In particular, we tested two recently proposed SAM-based approaches, including Efficient-SAM [65] and RSPrompter [7], and two methods specifically designed for underwater image instance segmentation, i.e., USIS-SAM [26] and BiPA [27]. As shown in Table IV, our method achieved the best result in terms of each evaluation metric. In particular, our SPDA-SAM outperformed the second-best approach by 2.3, 1.6 and 4.5 in terms of mAP , AP_{50} and AP_{75} , respectively, using the mask-level ground-truth data. Given that the box-level ground-truth data was utilized, the three values were 1.3, 2.5 and 1.3, respectively.

5) *KITTI*: We further evaluated the proposed SPDA-SAM on the KITTI [54] data set to verify its generalizability on real-world driving scenes. Since the official KITTI evaluation server only reports the mAP and AP_{50} metrics, we present the values of these metrics for comparison. The results obtained using our SPDA-SAM and nine baselines on the KITTI data set are reported in Table V. As can be seen, our SPDA-SAM achieved the best performance among the ten methods, reaching an mAP value of 27.5 and an AP_{50} value of 49.2. Compared to the second-best approach, i.e., UniDet RVC [68], our method outperformed it by 4.3 and 0.1 in terms of mAP and AP_{50} , respectively.

6) *ZeroWaste*: The ZeroWaste [55] data set was collected in extremely cluttered recycling environments. We used it to evaluate the robustness of our SPDA-SAM in handling complex object interactions and diverse material appearances. The results obtained using our SPDA-SAM and thirteen baselines on the ZeroWaste data set are shown in Table VI. It can be observed that SPDA-SAM achieved the best performance in terms of the mAP and AP_{75} metrics, which reach 34.4 and 38.2, respectively. Although USIS-SAM (ViT-Huge) [26]

TABLE II
COMPARISON BETWEEN OUR METHOD AND SIX BASELINE METHODS ON THE UIIS [57], UIIS10K [58] AND USIS16K [59] DATA SETS.

Method	Venue	UIIS			UIIS10K			USIS16K		
		mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}
Mask R-CNN [14]	ICCV17	23.4	40.9	25.3	35.8	53.6	40.2	73.6	90.0	81.7
Mask Scoring R-CNN [45]	CVPR19	24.6	41.9	26.5	37.3	<u>53.7</u>	<u>41.8</u>	71.6	88.5	79.8
PointRend [64]	CVPR20	25.9	43.4	27.6	37.3	53.0	41.3	76.3	89.4	82.7
WaterMask [57]	ICCV23	<u>27.2</u>	<u>43.7</u>	<u>29.3</u>	<u>37.4</u>	51.6	41.7	72.7	86.8	79.3
RSPrompter [7]	TGRS24	25.1	40.3	26.2	33.2	44.9	36.0	74.6	86.4	82.1
USIS-SAM [26]	ICML24	26.3	41.3	28.7	35.8	53.6	40.6	<u>81.0</u>	<u>90.8</u>	<u>87.1</u>
SAM [6]	ICCV23	27.0	42.0	28.2	36.0	52.5	40.5	71.9	85.2	81.4
SPDA-SAM (Ours)	-	30.7	47.3	34.0	41.7	55.4	45.7	82.3	92.5	88.3

TABLE III
COMPARISON BETWEEN OUR METHOD AND TEN STATE-OF-THE-ART METHODS ON THE USIS10K [26] DATA SET.

Method	Venue	Class-Agnostic			Multi-class		
		mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}
WaterMask [57]	ICCV23	58.3	80.2	66.5	37.7	54.0	42.5
S4Net [41]	CVPR19	32.8	64.1	27.3	23.9	43.5	24.4
RDPNet [42]	TIP21	53.8	77.8	61.9	37.9	55.3	42.7
OQTR [43]	TMM22	56.6	79.3	62.6	19.7	30.6	21.9
RSPrompter [7]	TGRS24	58.2	79.9	65.9	40.2	55.3	44.8
SAM+FasterRCNN [6]	ICCV23	45.9	65.9	52.1	26.4	38.9	29.0
SAM+MaskRCNN [6]	ICCV23	55.1	80.2	62.8	38.5	56.3	44.0
USIS-SAM [26]	ICML24	<u>59.7</u>	<u>81.6</u>	<u>67.7</u>	<u>43.1</u>	<u>59.0</u>	<u>48.5</u>
SAM [6]	ICCV23	59.5	80.2	66.8	41.7	55.7	47.5
SPDA-SAM (Ours)	-	64.6	84.9	74.1	45.9	60.3	52.3

TABLE IV
COMPARISON BETWEEN OUR METHOD AND FOURTEEN BASELINE METHODS ON THE LIACI [56] DATA SET.

Method	Venue	Mask			Box		
		mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}
Mask R-CNN [14]	ICCV17	36.4	58.9	40.0	36.1	60.5	38.4
SOLOv2 [21]	TPAMI21	34.8	54.7	37.5	-	-	-
CondInst [19]	ECCV20	38.6	<u>60.0</u>	39.7	<u>41.2</u>	60.9	46.0
BoxInst [66]	CVPR21	27.7	50.0	26.5	38.7	57.1	43.2
FastInst [67]	CVPR23	38.6	55.0	41.0	-	-	-
HTC [16]	CVPR19	36.4	55.2	39.2	37.8	55.4	40.5
QueryInst [47]	CVPR21	36.0	51.0	38.9	37.0	51.0	41.0
Diveseg [52]	AAAI26	38.7	55.0	41.7	-	-	-
USIS-SAM [26]	ICML24	27.8	44.6	28.8	26.3	47.4	25.3
Efficient-SAM [65]	CVPR24	33.3	51.4	33.9	35.3	54.7	39.2
RSPrompter [7]	TGRS24	40.5	58.7	42.3	40.8	<u>61.4</u>	45.9
SAMSnake [53]	NN25	34.0	52.1	35.2	-	-	-
BiPA [27]	CVPR26	<u>40.8</u>	59.2	<u>42.8</u>	41.0	60.5	<u>46.7</u>
SAM [6]	ICCV23	40.5	59.0	41.8	40.3	60.7	45.2
SPDA-SAM (Ours)	-	43.1	61.6	47.3	42.5	63.9	48.0

TABLE V
COMPARISON BETWEEN OUR METHOD AND NINE BASELINE METHODS ON THE KITTI [54] DATA SET.

Method	Venue	mAP	AP_{50}
UDeer DIS++ [69]	arXiv23	16.4	28.8
CenterPoly [70]	ICCV21	8.7	26.7
UniDet RVC [68]	CVPR22	<u>23.2</u>	<u>49.1</u>
Diveseg [52]	AAAI26	22.9	45.7
RSPrompter [7]	TGRS24	17.2	38.1
USIS-SAM [26]	ICML24	18.8	41.5
SAMSnake [53]	NN25	12.1	27.9
BiPA [27]	CVPR26	21.1	38.9
SAM [6]	ICCV23	15.3	32.9
SPDA-SAM (Ours)	-	27.5	49.2

TABLE VI
COMPARISON BETWEEN OUR SPDA-SAM AND THIRTEEN STATE-OF-THE-ART METHODS ON THE ZEROWASTE [55] DATA SET.

Method	Venue	mAP	AP_{50}	AP_{75}
Mask R-CNN [14]	ICCV17	29.0	44.9	31.8
SOLOv2 [21]	TPAMI21	19.4	35.9	17.7
CondInst [19]	ECCV20	30.4	45.2	33.2
BoxInst [66]	CVPR21	26.5	42.0	29.2
FastInst [67]	CVPR23	27.5	40.4	29.4
HTC [16]	CVPR19	29.2	46.8	31.9
QueryInst [47]	CVPR21	26.1	40.2	28.6
Diveseg [52]	AAAI26	28.7	42.1	30.5
RSPrompter [7]	TGRS24	28.7	41.7	32.0
USIS-SAM [26]	ICML24	<u>33.8</u>	49.8	<u>37.6</u>
SAMSnake [53]	NN25	18.7	28.7	20.5
BiPA [27]	CVPR26	33.5	47.6	37.3
SAM [6]	ICCV23	29.1	42.4	31.2
SPDA-SAM (Ours)	-	34.4	<u>47.8</u>	38.2

obtained the highest AP_{50} value, SPDA-SAM utilized a lighter ViT-Base backbone and outperformed the other baselines with regard to this metric. These results indicate that the proposed SPDA-SAM is able to perform robust and competitive instance segmentation in highly cluttered real-world scenarios.

TABLE VII
COMPARISON BETWEEN OUR SPDA-SAM AND TWELVE INSTANCE SEGMENTATION METHODS IN TERMS OF THE NUMBER OF PARAMETERS (M), FLOPS (G) AND INFERENCE SPEED (FPS).

Method	Venue	Params (M)	FLOPs (G)	Speed (FPS)
Mask R-CNN [14]	ICCV17	44.4	253.0	13.8
SOLOv2 [21]	TPAMI21	46.6	239.0	13.4
CondInst [19]	ECCV20	34.2	331.0	22.7
BoxInst [66]	CVPR21	35.1	372.0	22.0
FastInst [67]	CVPR23	34.1	58.2	28.9
QueryInst [47]	CVPR21	172.0	170.0	14.5
PolySnake [51]	TCSVT24	22.0	308.6	3.5
Diveseg [52]	AAAI26	390.1	2225.9	3.5
RSPrompter [7]	TGRS24	117.5	114.5	15.5
USIS-SAM [26]	ICML24	696.8	824.4	5.6
SAMSnake [53]	NN25	38.9	625.7	4.6
BiPA [27]	CVPR26	737.4	3554.0	1.7
SPDA-SAM (Ours)	-	225.4	264.5	12.2

7) *Computational Complexity Analysis*: We further compared the proposed SPDA-SAM with 12 instance segmentation methods in computational complexity, as shown in Table VII. The results were obtained on a single NVIDIA RTX 3080Ti GPU. It can be seen that our SPDA-SAM maintained a proper inference speed with a moderate number of parameters and FLOPs using the ViT-Base [62] backbone. Although the SPDA-SAM incurred the higher FLOPs value than lightweight models, such as FastInst [67], it achieved the better segmentation performance. As shown in Table IV, our method outperformed FastInst by 4.5 in terms of the mAP metric under the mask-level ground-truth setting. Compared to BiPA [27], our SPDA-SAM reduced the parameters and FLOPs by more than one half while improving the mask mAP value from 40.8 to 43.1 and the box mAP value from 41.0 to 42.5. It is suggested that our SPDA-SAM achieved a proper balance between computational complexity and segmentation accuracy.

C. Visualization

In this subsection, we will visualize the results obtained using the proposed SPDA-SAM, to intuitively convey the information that our method produced.

1) *Visualization of Instance Segmentation Masks*: We visualize the results produced by two state-of-the-art approaches, including RSPrompter [7] and USIS-SAM [26], and our method on the COME15K-E [38] and USIS10K [26] data sets in Figs. 3 and 4, respectively. As shown in Fig. 3, our SPDA-SAM generated more accurate and complete instance masks (see (c), (d) and (f)), detected a more complete set of instance objects (see (a), (e) and (j)), and better distinguished foreground objects from the background ((b), (g), (h) and (i)), compared to RSPrompter and USIS-SAM. Similarly, our method achieved the performance superior to its counterparts in handling complex scenes and object boundaries on the USIS10K [26] data set, as presented in Fig. 4. These findings should be due to the fact that our method utilized estimated

depth priors, except the RGB image characteristics that its counterparts also used.

2) *Visualization of Intermediate Results*: To intuitively understand the complementary actions between the estimated depth priors and the original image characteristics, we visualize the image embeddings produced by our depth-aware dual-path encoder in Fig. 5. It can be seen that object instances become more distinguishable and exhibit stronger contrast against the surrounding background once the depth data has been injected into the network, compared to those obtained without using geometric priors.

In addition, the foreground probability maps produced by the mask decoder of the SSSPM are visualized in Fig. 6, to intuitively demonstrate the effectiveness of the self-prompt mechanism. As can be seen, the foreground probability map predicted by the mask decoder in its second invocation with self-prompt shows higher confidences in the foreground regions, compared to that predicted in the first invocation. It is suggested that the self-prompt mechanism used by our SSSPM is useful for improving the performance of SAM.

D. Ablation Study

We performed five ablation experiments to examine the impacts of different modules of our SPDA-SAM. For simplicity, only the LIACI [56], ZeroWaste [55] and USIS10K [26] data sets were used. Regarding the LIACI data set, its mask-level ground-truth was utilized.

1) *Impact of the SSSPM*: We further examined the effect of the proposed SSSPM on our SPDA-SAM. Six variants of the SPDA-SAM were obtained by removing the entire SSSPM, using only the spatial prompt or the semantic prompt, and adopting different fusion strategies for the semantic and spatial prompts. These variants were compared with the SPDA-SAM in Table VIII. It can be observed that the use of the entire SSSPM produced the best result across the three data sets no matter which metric was used. It is suggested that the self-prompt module is useful for our SPDA-SAM.

2) *Impact of the Depth Estimator*: In addition, we investigated the effect of the depth estimator on our method. Four state-of-the-art monocular depth estimators were tested, including Dense Prediction Transformer (DPT) [71], Zero-shot Transfer by Combining Relative and Metric Depth (ZoeDepth) [72], Depth Anything (DA) [73] and Depth Anything v2 (DAv2) [60]. Given four underwater hull images, the depth maps that the four depth estimators generated were visualized in Fig. 7. It can be seen that the depth maps produced by DAv2 manifest the clearer boundaries, more consistent structures and better depth layering, compared with the other depth estimators. The results derived using our SPDA-SAM together with the four depth estimators are also compared in Table IX. As can be seen, the utilization of DAv2 normally produced the better result than those derived using the other depth estimators across the three data sets, regardless of which metric was used. This finding demonstrates that the quality of the depth map obtained using a pre-trained estimator is important to the performance of our SPDA-SAM.

Notably, the results in Table X, together with those in Tables IV and III, show that our method consistently outper-

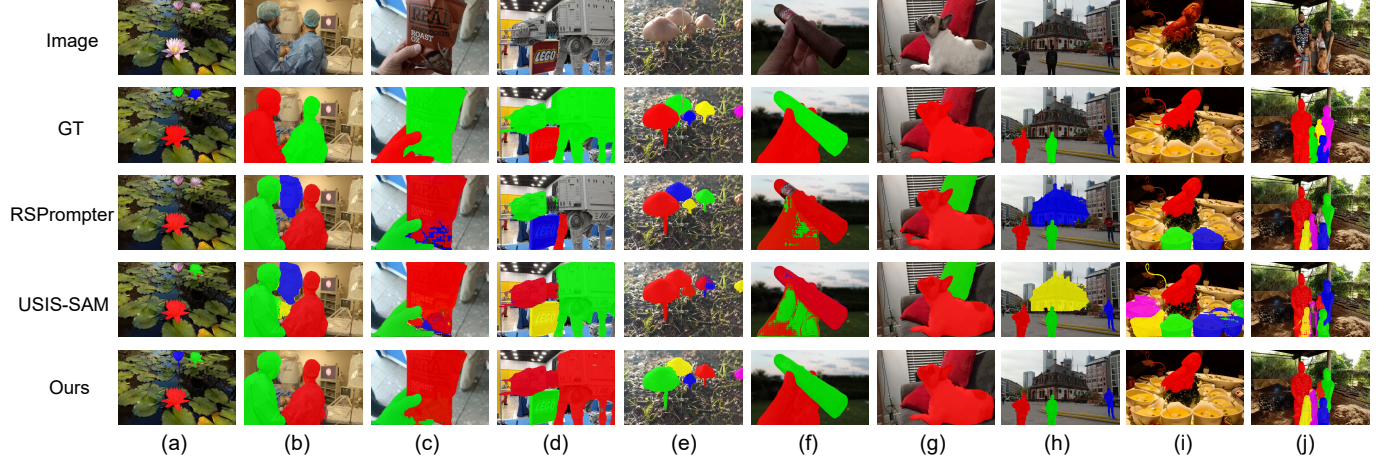


Fig. 3. Visualization of the results derived using two state-of-the-art baselines, i.e., RSPrompter [7] and USIS-SAM [26], and our method on the 10 images in the COME15K-E [38] data set.

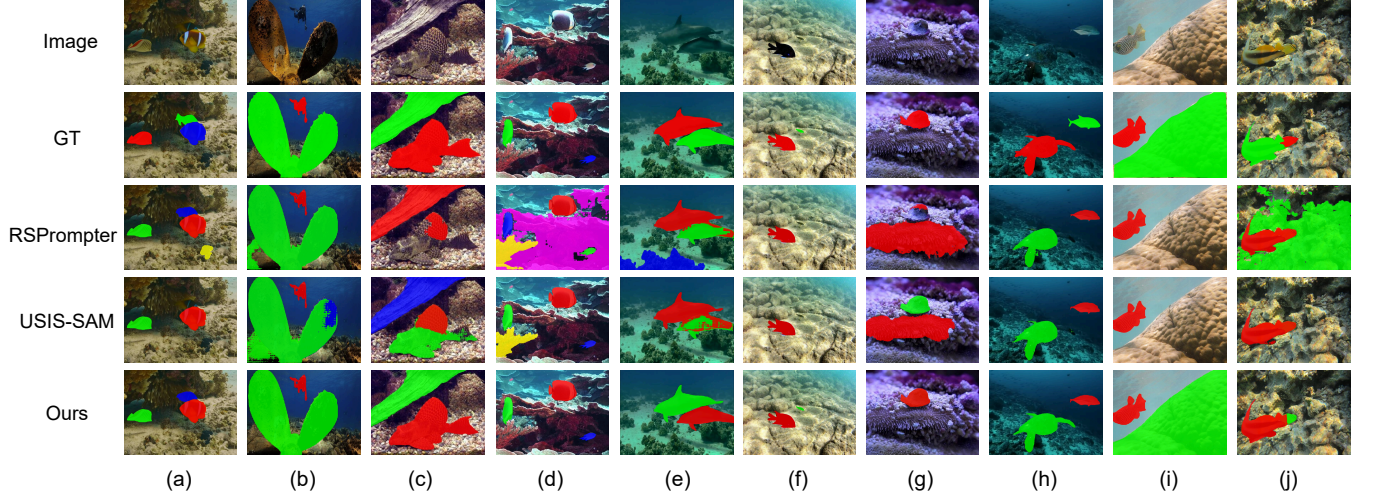


Fig. 4. Visualization of the results derived using two state-of-the-art baselines, i.e., RSPrompter [7] and USIS-SAM [26], and our method on the 10 images in the USIS10K [26] data set.

TABLE VIII
COMPARISON OF THE RESULTS OBTAINED USING THE SPDA-SAM WITH DIFFERENT PROMPT METHODS.

Prompt Method	Fusion Method	LIACI			ZeroWaste			USIS10K		
		mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}
w/o SSSPM	-	41.0	58.8	44.4	32.6	45.4	35.7	44.5	58.8	50.3
w/ Semantic	-	41.6	59.2	<u>46.0</u>	33.3	45.8	37.0	<u>45.0</u>	<u>60.0</u>	51.3
w/ Spatial	-	42.1	60.3	45.8	33.1	45.9	36.4	<u>45.0</u>	59.8	50.5
w/ Semantic & Spatial	Addition	40.8	58.0	43.9	32.2	45.0	35.3	43.7	57.0	48.7
	Cross-Attention	<u>42.3</u>	<u>61.0</u>	45.9	<u>33.5</u>	<u>46.2</u>	<u>37.2</u>	44.8	59.8	<u>51.5</u>
	Prompt Fusion Block	43.1	61.5	47.3	34.4	47.8	38.2	45.9	60.3	52.3

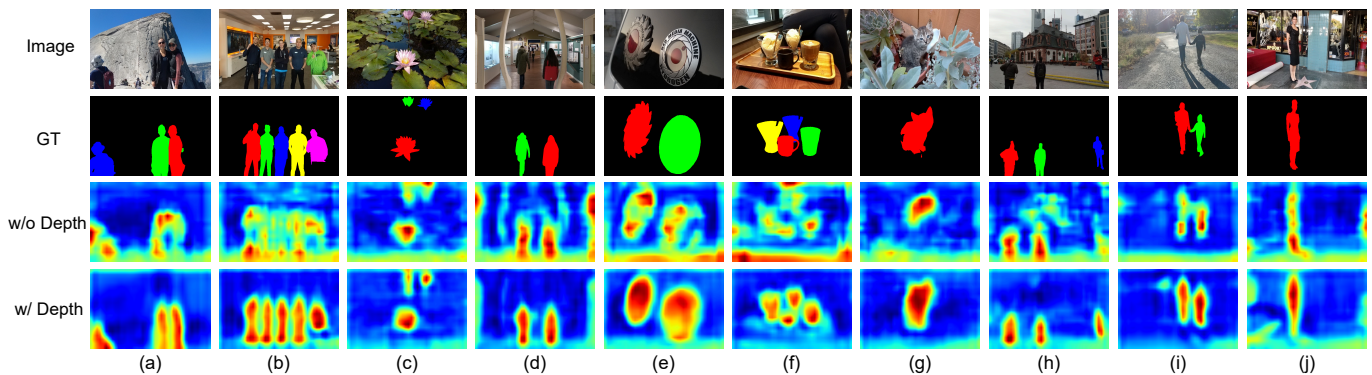


Fig. 5. Visualization of the complementary actions between the depth information and original image characteristics. With regard to an RGB image, the image embeddings generated by the depth-aware dual-path encoder are shown.

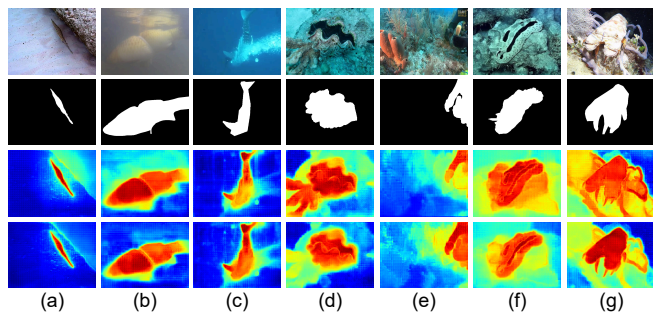


Fig. 6. Visualization of the foreground probability maps produced using the mask decoder of the SSSPM during the two-stage segmentation process. Within each group, the input image, ground-truth mask and two foreground probability maps predicted by the mask decoder in its first and second invocations are shown in turn.

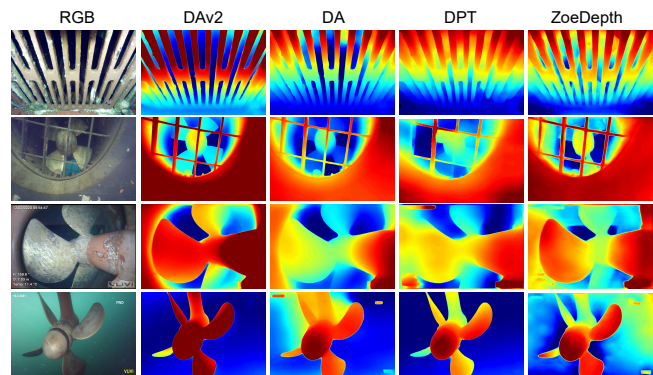


Fig. 7. Visual comparison of the depth maps obtained using four depth estimators from four underwater hull images in the LIACI [56] data set.

formed the baselines across the three datasets even when depth information was unavailable.

3) *Impact of the C2FFM*: To investigate the effectiveness of the proposed C2FFM and the contribution of its components, we compared three variants, including those without C2FFM, with only the coarse fusion block and with only the fine fusion blocks, with the full C2FFM. As reported in Table X, the full C2FFM consistently outperformed the three variants across all three data sets. The fine fusion blocks contributed a larger gain compared to the coarse fusion block alone, while

TABLE IX
COMPARISON OF THE RESULTS OBTAINED USING THE SPDA-SAM WITH DIFFERENT DEPTH ESTIMATORS.

Depth Estimator	LIACI			ZeroWaste			USIS10K		
	mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}
DPT	42.4	58.3	47.0	32.5	45.1	35.9	44.8	59.5	51.6
ZoeDepth	42.2	59.1	46.8	32.9	45.6	36.7	44.6	59.4	51.1
DepthAnything	<u>42.5</u>	<u>60.5</u>	<u>47.2</u>	<u>33.6</u>	<u>46.6</u>	<u>37.0</u>	<u>45.3</u>	<u>59.8</u>	<u>51.9</u>
DepthAnything v2	43.1	61.5	47.3	34.4	47.8	38.2	45.9	60.3	52.3

TABLE X
COMPARISON BETWEEN THE RESULTS OBTAINED USING THE SPDA-SAM WITH DIFFERENT C2FFM VARIANTS.

Method	LIACI			ZeroWaste			USIS10K		
	mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}
w/o C2FFM	42.0	59.7	46.5	33.2	45.9	37.2	44.6	59.5	51.5
w/ Coarse Fusion	42.2	60.0	46.3	33.5	46.4	37.3	44.7	59.8	51.5
w/ Fine Fusion	42.7	61.0	46.9	34.1	46.9	37.8	45.4	60.0	51.8
w/ Coarse&Fine	43.1	61.5	47.3	34.4	47.8	38.2	45.9	60.3	52.3

the combination of them achieved the best result, indicating that coarse-level contour guidance and fine-grained local refinement are complementary. These results confirm that the estimated depth priors are beneficial to instance segmentation, particularly when they are fused with the RGB features using the proposed coarse-to-fine strategy.

4) *Impact of the Mask Decoder Initialization*: We noted that the pre-trained decoder weights of SAM carry biases optimized for the standard decoding context of SAM, whereas our decoder operates under a self-prompted, iterative two-pass decoding workflow that differs from the pre-trained decoder substantially. Consequently, the pre-trained cross-attention patterns may become suboptimal and even constrain adaptation to the new context, while random initialization allows the decoder to learn attention patterns, tailored to our decoding workflow, from scratch. To empirically verify this hypothesis, we compared our SPDA-SAM with a variant that loads the mask decoder weights of the pre-trained SAM, with the other components and training settings kept identical. As reported in

TABLE XI

COMPARISON BETWEEN THE RESULTS OBTAINED USING THE SPDA-SAM WITH DIFFERENT MASK DECODER INITIALIZATION STRATEGIES.

Method	LIACI			ZeroWaste			USIS10K		
	mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}	mAP	AP_{50}	AP_{75}
SAM Init	42.5	60.7	46.8	33.5	46.7	37.6	45.1	59.5	51.6
Random Init	43.1	61.5	47.3	34.4	47.8	38.2	45.9	60.3	52.3

TABLE XII

THE mAP VALUES OBTAINED BY APPLYING THE PROPOSED SSSPM AND C2FFM TO DIFFERENT VERSIONS OF SAMs ON THE LIACI [56] DATA SET. HERE, BASE MODEL DENOTES THE VARIANT ADOPTED BASED ON THE BACKBONE OF THE CORRESPONDING VERSION OF SAM WITHOUT SSSPM AND C2FFM.

Method	Base Model	+SSSPM	+C2FFM	+SSSPM +C2FFM
SAM3	37.2	38.5 (+1.3)	37.9 (+0.7)	39.5 (+2.3)
SAM2	36.9	38.0 (+1.1)	37.7 (+0.7)	39.0 (+2.1)
SAM	40.5	42.0 (+1.5)	41.0 (+0.5)	43.1 (+2.6)

Table XI, random initialization consistently outperformed the case in which pre-trained decoder weights were loaded across the three data sets regardless of which evaluation metric was used, confirming the above finding further.

5) *Generalization to SAM2 and SAM3*: To examine the generalizability of the proposed SSSPM and C2FFM to different versions of SAMs, we applied them to SAM2 [23] and SAM3 [24]. As shown in Table XII, the use of both modules also improved SAM2 and SAM3. Specifically, SSSPM brought improvements of 1.1 and 1.3 in mAP on SAM2 and SAM3, respectively, demonstrating that the proposed self-prompt mechanism is effective across different versions of SAMs. Similarly, C2FFM improved the two SAMs by 0.7 in mAP , indicating that depth-aware coarse-to-fine fusion can consistently enhance their spatial and boundary representations. When both SSSPM and C2FFM were used together with SAM2 and SAM3 individually, the improvements become larger. In this case, the mAP values obtained using SAM2 and SAM3 were improved by 2.1 and 2.3, respectively. These findings show that the proposed SSSPM and C2FFM can be generalized to different SAM families and are complementary. However, the performance of SAM2 and SAM3 was always inferior to that of SAM regardless of whether or not these modules were utilized.

It should be noted that SAM2 and SAM3 did not achieve superior performance to that of SAM. The primary contribution of SAM2 is the extension of promptable segmentation from images to videos. However, its temporal memory mechanism cannot provide additional cross-frame information when the input consists of only a single image. In addition, SAM3 reformulates the core task as Promptable Concept Segmentation (PCS), which aims to identify and segment all instances that match a given conceptual prompt. Nevertheless, such prompts are not available in our experiment.

V. CONCLUSION

In this paper, we proposed a Self-prompted Depth-Aware Segment Anything Model (SPDA-SAM) for instance segmentation. Specifically, we designed a Semantic-Spatial Self-prompt Module (SSSPM) to alleviate the reliance of SAM on manual prompts. This module fused the high-level semantic features extracted from the encoder and the features extracted from the preliminary mask that the mask decoder generated. As a result, a prompt enriched by both semantic and spatial information was produced, which could direct the segmentation process. This design freed SAM from the reliance on human intervention. For the purpose of addressing the absence of geometric priors in instance segmentation, we introduced a Coarse-to-Fine RGB-D Fusion Module (C2FFM), which progressively performed coarse-to-fine feature fusion across different stages in our dual-path encoder. This module enabled effective exploitation of the complementary information embedded in RGB images and estimated depth priors. To the authors' knowledge, SAM has not been explored in such self-prompted and depth-aware manners. Experimental results showed that our SPDA-SAM normally outperformed its state-of-the-art counterparts across eleven instance segmentation data sets. We believe that the superior results are due to the guidance of the self-prompts and the compensation for the spatial information loss benefited from the coarse-to-fine RGB-D fusion.

REFERENCES

- [1] D. Zhou, J. Fang, X. Song, L. Liu, J. Yin, Y. Dai, H. Li, and R. Yang, "Joint 3d instance segmentation and object detection for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1839–1849.
- [2] Q. Xu, J. Li, X. He, C. Li, F. B. Tesema, W. Duan, Z. Chen, R. Qu, J. M. Garibaldi, and C. W. Chen, "De-lightsam: Modality-decoupled lightweight sam for generalizable medical segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.
- [3] H. Su, S. Wei, S. Liu, J. Liang, C. Wang, J. Shi, and X. Zhang, "Hq-isnet: High-quality instance segmentation for remote sensing imagery," *Remote Sensing*, vol. 12, no. 6, p. 989, 2020.
- [4] B. Dong, Z. Zhu, Q. Bu, M. Wu, and J. Ni, "A two-stage method with lightweight network and active contour model for remote sensing image segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.
- [5] J. Hsu, W. Chiu, and S. Yeung, "Darcnn: Domain adaptive region-based convolutional neural network for unsupervised instance segmentation in biomedical images," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 1003–1012.
- [6] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollar, and R. Girshick, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2023, pp. 4015–4026.
- [7] K. Chen, C. Liu, H. Chen, H. Zhang, W. Li, Z. Zou, and Z. Shi, "Rsprompter: Learning to prompt for remote sensing instance segmentation based on visual foundation model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–17, 2024.
- [8] Y. Wang, W. Zhou, Y. Mao, and H. Li, "Detect any shadow: Segment anything for video shadow detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 5, pp. 3782–3794, 2023.
- [9] R. Hou, X. Li, T. Ren, D. Zhou, G. Wu, and J. Cao, "Hypsam: Hybrid prompt-driven segment anything model for rgb-thermal salient object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.
- [10] S.-A. Liu, H. Xie, J. Ge, and Y. Zhang, "Refersam: Unleashing segment anything model for referring image segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 35, no. 5, pp. 4910–4922, 2025.

- [11] W. Tang, X. Liu, Y. Sun, and Z. Li, "Ssp-sam: Sam with semantic-spatial prompt for referring expression segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.
- [12] 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.
- [13] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen *et al.*, "Lora: Low-rank adaptation of large language models," *ICLR*, vol. 1, no. 2, p. 3, 2022.
- [14] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.
- [15] Z. Cai and N. Vasconcelos, "Cascade r-cnn: High quality object detection and instance segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 5, pp. 1483–1498, 2019.
- [16] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang *et al.*, "Hybrid task cascade for instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4974–4983.
- [17] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "Yolact: Real-time instance segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9157–9166.
- [18] H. Chen, K. Sun, Z. Tian, C. Shen, Y. Huang, and Y. Yan, "Blendmask: Top-down meets bottom-up for instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8573–8581.
- [19] Z. Tian, C. Shen, and H. Chen, "Conditional convolutions for instance segmentation," in *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*. Springer, 2020, pp. 282–298.
- [20] H. Ying, Z. Huang, S. Liu, T. Shao, and K. Zhou, "Embedmask: Embedding coupling for one-stage instance segmentation," *arXiv preprint arXiv:1912.01954*, 2019.
- [21] X. Wang, R. Zhang, T. Kong, L. Li, and C. Shen, "Solov2: Dynamic and fast instance segmentation," *Advances in Neural information processing systems*, vol. 33, pp. 17 721–17 732, 2020.
- [22] 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.
- [23] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson *et al.*, "Sam 2: Segment anything in images and videos," in *International Conference on Learning Representations*, vol. 2025, 2025, pp. 28 085–28 128.
- [24] N. Carion, L. Gustafson, Y.-T. Hu, S. Debnath, R. Hu, D. Suris, C. Ryali, K. V. Alwala, H. Khedr, A. Huang *et al.*, "Sam 3: Segment anything with concepts," *arXiv preprint arXiv:2511.16719*, 2025.
- [25] C. Chen, J. Miao, D. Wu, A. Zhong, Z. Yan, S. Kim, J. Hu, Z. Liu, L. Sun, X. Li *et al.*, "Ma-sam: Modality-agnostic sam adaptation for 3d medical image segmentation," *Medical Image Analysis*, vol. 98, p. 103310, 2024.
- [26] S. Lian, Z. Zhang, H. Li, W. Li, L. T. Yang, S. Kwong, and R. Cong, "Diving into underwater: Segment anything model guided underwater salient instance segmentation and a large-scale dataset," *arXiv preprint arXiv:2406.06039*, 2024.
- [27] L. Ma, H. Zheng, Y. Mao, J. Liu, C. Xu, X. Xue, Y. Wang, X. He, and W. Wang, "Bipa: Bilevel prompt adaptation for underwater instance segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026, pp. 10 731–10 740.
- [28] J. Lin, X. Zhou, J. Liu, R. Cong, G. Zhang, Z. Liu, and J. Zhang, "Sam-daq: Segment anything model with depth-guided adaptive queries for rgb-d video salient object detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, no. 9, 2026, pp. 6952–6960.
- [29] M. Jia, L. Tang, B.-C. Chen, C. Cardie, S. Belongie, B. Hariharan, and S.-N. Lim, "Visual prompt tuning," in *European conference on computer vision*. Springer, 2022, pp. 709–727.
- [30] 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.
- [31] J. Li, Y. Zhang, Y. Cui, Y. Sun, and Q. Liu, "Vision-language driven prompt learning for weakly supervised semantic segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2026.
- [32] X. Ying and M. C. Chuah, "Ucnet: Uncertainty-aware cross-modal transformer network for indoor rgb-d semantic segmentation," in *European Conference on Computer Vision*. Springer, 2022, pp. 20–37.
- [33] J. Yang, L. Bai, Y. Sun, C. Tian, M. Mao, and G. Wang, "Pixel difference convolutional network for rgb-d semantic segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 3, pp. 1481–1492, 2023.
- [34] Q. Zhang, Q. Qin, Y. Yang, Q. Jiao, and J. Han, "Feature calibrating and fusing network for rgb-d salient object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 3, pp. 1493–1507, 2023.
- [35] 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.
- [36] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.
- [37] 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.
- [38] J. Zhang, D.-P. Fan, Y. Dai, X. Yu, Y. Zhong, N. Barnes, and L. Shao, "Rgb-d saliency detection via cascaded mutual information minimization," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 4338–4347.
- [39] J. Pei, T. Jiang, H. Tang, N. Liu, Y. Jin, D.-P. Fan, and P.-A. Heng, "Calibnet: Dual-branch cross-modal calibration for rgb-d salient instance segmentation," *IEEE Transactions on Image Processing*, 2024.
- [40] D.-P. Fan, Z. Lin, Z. Zhang, M. Zhu, and M.-M. Cheng, "Rethinking rgb-d salient object detection: Models, data sets, and large-scale benchmarks," *IEEE Transactions on neural networks and learning systems*, vol. 32, no. 5, pp. 2075–2089, 2020.
- [41] R. Fan, M.-M. Cheng, Q. Hou, T.-J. Mu, J. Wang, and S.-M. Hu, "S4net: Single stage salient-instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6103–6112.
- [42] Y.-H. Wu, Y. Liu, L. Zhang, W. Gao, and M.-M. Cheng, "Regularized densely-connected pyramid network for salient instance segmentation," *IEEE Transactions on Image Processing*, vol. 30, pp. 3897–3907, 2021.
- [43] J. Pei, T. Cheng, H. Tang, and C. Chen, "Transformer-based efficient salient instance segmentation networks with orientative query," *IEEE Transactions on Multimedia*, vol. 25, pp. 1964–1978, 2022.
- [44] J. Pei, T. Cheng, D.-P. Fan, H. Tang, C. Chen, and L. Van Gool, "Osformer: One-stage camouflaged instance segmentation with transformers," in *European conference on computer vision*. Springer, 2022, pp. 19–37.
- [45] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, "Mask scoring r-cnn," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6409–6418.
- [46] Y. Lee and J. Park, "Centermask: Real-time anchor-free instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 13 906–13 915.
- [47] Y. Fang, S. Yang, X. Wang, Y. Li, C. Fang, Y. Shan, B. Feng, and W. Liu, "Instances as queries," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 6910–6919.
- [48] R. Guo, D. Niu, L. Qu, and Z. Li, "Sotr: Segmenting objects with transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 7157–7166.
- [49] L. Ke, M. Danelljan, X. Li, Y.-W. Tai, C.-K. Tang, and F. Yu, "Mask transfiner for high-quality instance segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4412–4421.
- [50] T. Cheng, X. Wang, S. Chen, W. Zhang, Q. Zhang, C. Huang, Z. Zhang, and W. Liu, "Sparse instance activation for real-time instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 4433–4442.
- [51] H. Feng, K. Zhou, W. Zhou, Y. Yin, J. Deng, Q. Sun, and H. Li, "Recurrent generic contour-based instance segmentation with progressive learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 9, pp. 7947–7961, 2024.
- [52] Z. Chen, C. Zhang, H. Fang, and R. Cong, "Empowering dino representations for underwater instance segmentation via aligner and prompt," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, no. 5, 2026, pp. 3201–3209.

- [53] Y. Wu, J. Zhan, C. Guo, and H. Zhang, "Samsnake: A generic contour-based instance segmentation network assisted by efficient segment anything model," *Neural Networks*, vol. 189, p. 107491, 2025.
- [54] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The international journal of robotics research*, vol. 32, no. 11, pp. 1231–1237, 2013.
- [55] D. Bashkirova, M. Abdelfattah, Z. Zhu, J. Akl, F. Alladkani, P. Hu, V. Ablavsky, B. Calli, S. A. Bargal, and K. Saenko, "Zerowaste dataset: Towards deformable object segmentation in cluttered scenes," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 21 147–21 157.
- [56] M. Waszak, A. Cardaillac, B. Elveaseter, F. Rødølen, and M. Ludvigsen, "Semantic segmentation in underwater ship inspections: Benchmark and data set," *IEEE Journal of Oceanic Engineering*, vol. 48, no. 2, pp. 462–473, 2022.
- [57] S. Lian, H. Li, R. Cong, S. Li, W. Zhang, and S. Kwong, "Watermask: Instance segmentation for underwater imagery," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 1305–1315.
- [58] H. Li, S. Lian, Z. Li, R. Cong, and S. Kwong, "Uwsam: Segment anything model guided underwater instance segmentation and a large-scale benchmark dataset," *arXiv preprint arXiv:2505.15581*, 2025.
- [59] L. Hong, X. Wang, Y. Li, and X. Wang, "Usis16k: High-quality dataset for underwater salient instance segmentation," *arXiv preprint arXiv:2506.19472*, 2025.
- [60] L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, and H. Zhao, "Depth anything v2," *Advances in Neural Information Processing Systems*, vol. 37, pp. 21 875–21 911, 2024.
- [61] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.
- [62] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.
- [63] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu *et al.*, "Mmdetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019.
- [64] A. Kirillov, Y. Wu, K. He, and R. Girshick, "Pointrend: Image segmentation as rendering," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9799–9808.
- [65] Y. Xiong, B. Varadarajan, L. Wu, X. Xiang, F. Xiao, C. Zhu, X. Dai, D. Wang, F. Sun, F. Iandola *et al.*, "Efficientsam: Leveraged masked image pretraining for efficient segment anything," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16 111–16 121.
- [66] Z. Tian, C. Shen, X. Wang, and H. Chen, "Boxinst: High-performance instance segmentation with box annotations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5443–5452.
- [67] J. He, P. Li, Y. Geng, and X. Xie, "Fastinst: A simple query-based model for real-time instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 23 663–23 672.
- [68] X. Zhou, V. Koltun, and P. Krähenbühl, "Simple multi-dataset detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 7571–7580.
- [69] Z. Dong, H. Ji, X. Huang, W. Zhang, X. Zhan, and J. Chen, "Pep: a point enhanced painting method for unified point cloud tasks," *arXiv preprint arXiv:2310.07591*, 2023.
- [70] H. Perreault, G.-A. Bilodeau, N. Saunier, and M. Héritier, "Centerpoly: Real-time instance segmentation using bounding polygons," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 2982–2991.
- [71] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 12 179–12 188.
- [72] S. F. Bhat, R. Birkel, D. Wofk, P. Wonka, and M. Müller, "Zoedepth: Zero-shot transfer by combining relative and metric depth," *arXiv preprint arXiv:2302.12288*, 2023.
- [73] L. Yang, B. Kang, Z. Huang, X. Xu, J. Feng, and H. Zhao, "Depth anything: Unleashing the power of large-scale unlabeled data," in *CVPR*, 2024.



Yihan Shang received the B.S. degree in Network Engineering from Harbin University of Science and Technology, Harbin, China, in 2021. He is currently pursuing the M.S. degree in Computer Technology with the Ocean University of China, Qingdao, China. His research interests include computer vision, image segmentation.



Wei Wang is currently a post-doctoral candidate at the School of Cyber Science and Technology, Shenzhen Campus of Sun Yat-sen university, Shenzhen, China. He received the Ph.D. degree at the School of Software Technology, Dalian University of Technology, Dalian, China, in 2022. He received the M.S. degree at the School of Computer Science and Technology from the Anhui University, Hefei, China, in 2018. His major research interests include transfer learning, zero-shot learning, deep learning, etc.



Chao Huang (Member, IEEE) received the Ph.D. degree in computer science and technology from Harbin Institute of Technology, Shenzhen, China, in 2022. From 2019 to 2022, he was a visiting scholar with Peng Cheng Laboratory, Shenzhen. He is currently an Assistant Professor with the School of Cyber Science and Technology, Sun Yat-sen University, Shenzhen. So far, he has published over 60 technical papers in prestigious international journals and conferences. His research interests include anomaly detection, multimedia analysis, object detection, image/video compression, and deep learning. Dr. Huang received the Distinguished Paper Award of AAAI 2023, and his dissertation was nominated for Harbin Institute of Technology's Outstanding Dissertation Award. He serves as an Associate Editor for Pattern Recognition and serves/served as the reviewer/ PC member for several top-tier journals and conferences, including IEEE TPAMI, TIP, TIFS, TNNLS, ACM CSUR, CVPR, ICCV, ECCV, ICML, NeurIPs, ICLR, AAAI, IJCAI, and ACM Multimedia.



Junyu Dong received the B.Sc. and M.Sc. degrees from the Department of Applied Mathematics, Ocean University of China, Qingdao, China, in 1993 and 1999, respectively, and the Ph.D. degree in image processing from the Department of Computer Science, Heriot-Watt University, U.K., in 2003. He joined Ocean University of China in 2004. He is currently a Professor and the Dean of the Faculty of Information Science and Engineering, Ocean University of China. His research interests include computer vision, underwater image processing, and machine learning, with more than ten research projects supported by the NSFC, MOST, and other funding agencies.



Xinghui Dong (Member, IEEE) received the Ph.D. degree from Heriot-Watt University, U.K., in 2014. He was with the Centre for Imaging Sciences, University of Manchester, U.K., from 2015 to 2021. In 2021, he joined the Ocean University of China, where he is currently a Professor. His research interests include computer vision, defect detection, texture analysis, and visual perception.