Weakly-Supervised RGB-D Salient Object Detection via SAM-driven Pseudo Annotation and State Space Interaction-based Diffusion

Wenqi Si, Gongyang Li, *Member, IEEE*, Shixiang Shi, and Weisi Lin, *Fellow, IEEE*

***Abstract*—Weakly-supervised RGB-D Salient Object Detection (SOD) is explored to reduce the heavy burden of pixel-level annotations. But scribble annotations lack the structure and details of objects, resulting in inaccurate saliency maps. In this paper, we propose a novel scribble-supervised RGB-D SOD method, consisting of a Segment Anything Model (SAM)-driven pseudo annotation generation method (*SAM-PAG*) and a state space interaction-based conditional diffusion model (*S^2Diff*). Specifically, SAM-PAG is tailored to address the issue of sparse supervision information. In SAM-PAG, we adopt the advanced SAM to expand sparse scribbles to dense pixel-level pseudo annotations through the dual-branch structure and the consistency of segmentation masks. In S^2Diff, we adopt the diffusion model to iteratively refine the noisy saliency maps with the guidance of conditional information, generating accurate saliency maps. Naturally, the core of our S^2Diff lies in the acquisition of conditional features and the denoising of saliency maps. For the former, we employ a cross-modal conditional generation module to interweave cross-modal features through frequency integration and implicit-explicit state space interaction, effectively achieving global conditional features. For the latter, we employ a context injection module to mitigate noise interference and to enhance object information with the conditional context. With the close cooperation of SAM-PAG and S^2Diff, our method outperforms relevant scribble-supervised methods and achieves competitive performance compared to fully-supervised methods on seven datasets. The code and results of our method are available at https://github.com/Switch457/WeakS2Diff_SOD.**

***Index Terms*—RGB-D salient object detection, weakly supervision, segment anything, conditional diffusion model.**

I. Introduction

SALIENT Object Detection (SOD), a fundamental task in the field of computer vision, aims to localize and segment the regions and objects that attract human attention most in images. Fully-supervised RGB-D SOD [1]–[7], has made great progress in recent years. However, their superior performance comes at the cost of extensive manual annotation, because these models are trained with pixel-level annotations. Therefore, weakly-supervised RGB-D SOD, aiming to achieve a better trade-off between annotation efficiency and model performance, has attracted increasing attention.

Wenqi Si, Gongyang Li, and Shixiang Shi are with the School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China (e-mail: {swq; ligongyang; shishixiang}@shu.edu.cn).

Weisi Lin is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (e-mail: wslin@ntu.edu.sg).

Corresponding author: Gongyang Li.

Weakly-supervised RGB-D SOD adopts sparse annotations, *e.g.*, points [8], image-level labels [8], [9], and scribbles [10]–[14]. Among various sparse annotations, scribble annotations need just some lines on salient objects and background regions, which can provide basic distinguishable information while reducing annotation costs. Liu *et al*. [11] directly used scribbles as supervision, and performed feature clustering to regularize the latent space to salient and non-salient objects. But as a kind of sparse annotation, scribbles inevitably encounter the lack of supervision information, resulting in a limitation on model training. Moreover, some methods are dedicated to generating dense pseudo labels from sparse scribbles. For example, Li *et al*. [13] enriched the pseudo label sample set through a visual information-based selection mechanism. Some methods [10], [14] generated pseudo annotations via iteratively updating model prediction, gradually optimizing pseudo annotations as model capability improves. While the other methods [12] directly trained a model to produce dense pseudo labels in order to train a new model from scratch. However, the dense pseudo labels generated by the above methods are relatively rough, as the methods for generating pseudo labels are based on insufficient supervision.

To obtain high-quality dense pseudo labels, we introduce the powerful Segment Anything Model (SAM) [15] into the weakly-supervised RGB-D SOD. We propose a novel scribble-supervised RGB-D SOD method, including a SAM-driven Pseudo Annotation Generation method (*SAM-PAG*) and a state space interaction-based conditional diffusion model (*S^2Diff*). SAM-PAG leverages the strong segmentation ability of the visual foundation model SAM to produce reliable pixel-level pseudo annotations, S^2Diff is a strong diffusion-based saliency detector, consisting of a conditional feature generation network and a denoising network. It generates accurate saliency maps via iterative refinement with the guidance of cross-modal information. Moreover, we embed state space models into our S^2Diff to generate the conditional context and reconstruct noise features effectively.

Specifically, to provide exact dense supervision, our SAM-PAG deploys a dual-branch structure and a consistency-based fusion method. Based on our testing and validation of SAM, we found that SAM exhibits high sensitivity to image transformation. Therefore, we propose an image transformation branch in SAM-PAG to generate complete masks through complementarity of various image transformation masks. We integrate the prompt extension branch into SAM-PAG to expand the potential regions of prompts from scribbles via

the superpixel propagation. And the consistency-based fusion method fuses the segmentation masks from two branches, improving the quality of pseudo pixel-level annotations. With the pseudo pixel-level annotations and scribble-level annotations, our S^2Diff can be trained with sufficient supervision. In the conditional feature generation network of S^2Diff, the encoder extracts RGB and depth features first. Then, we adopt the Cross-modal Conditional Generation Module (CCGM) to obtain the conditional features with long-range dependencies. CCGM makes the amplitude-phase exchange in the frequency domain and then explores the implicit-explicit state space interaction of cross-modal features. In the denoising network, we employ the Context Injection Module (CIM), where the conditional context modulated by noisy mask channel information regulates the recovery of noise features, eliminating noise impact gradually. With the guidance of global conditional information, noise is removed from the noisy masks to produce accurate saliency maps iteratively.

The contributions of our work are summarized as follows:

- We propose a novel scribble-supervised RGB-D SOD method, which adopts SAM to expand scribbles to dense pixel-level pseudo annotations and constructs a powerful conditional diffusion model as saliency detector. With the supervision of scribbles and pixel-level pseudo annotations, our method achieves remarkable performance.
- We propose a SAM-driven pseudo annotation generation method (*SAM-PAG*) based on the dual-branch structure. One branch achieves complementary segmentation according to the sensitivity of SAM to image transformations, while the other branch improves object integrity by expanding the potential regions of prompts. A consistency-based fusion method integrates the segmentation masks of two branches to generate comprehensive dense pseudo annotations.
- We propose a state space interaction-based conditional diffusion model (S^2*Diff*) to generate accurate saliency maps. In S^2Diff, CCGMs in the conditional feature generation network interweave cross-modal features for global conditional information. CIMs in the denoising network regulate the reconstruction of noise features into semantically explicit features, mitigating noise interference during the conditional context-aware denoising.

II. Related Work

A. Fully-supervised RGB-D Salient Object Detection

With the development of deep learning, RGB SOD [16], [17] has achieved promising performance. Due to the susceptibility of RGB images to lighting conditions, many works introduce depth maps rich in spatial information to complement RGB images, *i.e*., RGB-D SOD [1]–[7]. Existing RGB-D SOD methods are mainly trained in a fully-supervised manner and have obtained great achievements. For example, Fan *et al*. [2] utilized a bifurcated backbone strategy to explore complementary information between multi-level features and excavate depth information to enhance RGB features. Li *et al*. [3] proposed cross-modal weighting modules based on the different natures of features from different levels, using multi-scale features for weighting. Zeng *et al*. [5] designed a parallel attention-shift convolution to capture contextual and local information simultaneously.

These methods have shown remarkable performance, but the superiority of the models heavily relies on pixel-level annotations, which are expensive and time-consuming. In this paper, we focus on the weakly-supervised RGB-D SOD to balance the annotation cost and model performance.

B. Weakly-supervised RGB-D Salient Object Detection

For weakly-supervised RGB-D SOD, sparse annotations, such as bounding boxes, points, and scribbles, serve as supervision. Compared with other sparse annotations, scribbles can provide more direct position information about objects and backgrounds without introducing negative samples simultaneously. Many works focus on scribble-supervised RGB-D SOD. Liu *et al*. [11] performed feature clustering guided by scribbles, regularizing the latent space to discriminate salient and non-salient objects. This method directly used scribbles as supervision.

To compensate for the lack of supervision information, many methods produced pixel-level annotations with scribbles. For example, Li *et al*. [13] selected reliable partial pseudo labels through consistency and confidence to expand the online pseudo annotation set. Liu *et al*. [18] aggregated multi-modal prediction via the methods of averaging and smoothing to obtain the pseudo labels in scribble-supervised multi-modal SOD. Compared to the methods that generate pseudo labels based on traditional visual information, some methods update the pseudo annotations with iterative optimization of model prediction. Xu *et al*. [10] imposed teacher-student framework, supervising the training of student model with the prediction of teacher models and updating teacher model with an ensemble of the student model at different training iterations. Wang *et al*. [14] adopted a moving average strategy to merge previous pseudo labels with current post-processed predictions. Differently, Li *et al*. [12] took the outputs from the first training as pseudo labels to train a new multi-modal VAE network for prediction refinement.

Most of the above methods generated dense pseudo labels to provide pixel-level supervision. But their dense pseudo labels were often constructed from model predictions [10], [12]–[14] and their post-processing strategies [13], [14], [18]. As a result, the quality of pseudo annotations may be limited by model performance and tends to be relatively coarse. Therefore, we leverage the strong segmentation ability of SAM and exploit its characteristics to design a pseudo annotation generation method to produce more accurate dense pseudo labels, providing supervision information similar to manually annotated dense labels.

C. Segment Anything Model

SAM [15] is a visual foundation model that implements segmentation of any objects with provided prompts. It can transfer zero-shot to new data distributions and tasks using prompt engineering. Due to the powerful segmentation capability and generalization ability of SAM, many researchers apply

SAM to various visual tasks. To generalize SAM to specific tasks, some works adopted fine-tuning techniques [19]–[21]. Ma *et al*. [19] adapted SAM to medical images by directly finetuning it on a large-scale multi-modal dataset, building a promptable foundation model for general medical image segmentation. Wu *et al*. [20] integrated SAM into medical image segmentation with a lightweight adaptation technique. Some works adopted prompt-based strategies to realize the downstream tasks [22]–[25]. Tang *et al*. [22] generated point prompts by interactively matching the features of prompt images and input images, which further guided SAM to achieve flexible open-world segmentation. Yuan *et al*. [24] utilized a text-driven approach, extracting coarse saliency maps with CLIP [26], then through random point sampling and connected component analysis, providing points and bounding box prompts for SAM, subsequently generating refined saliency maps. Hu *et al*. [25] combined self-training and SAM, fusing pseudo annotations from pre-trained model and SAM to effectively address the issues of SAM lacking camouflage domain knowledge and the insufficient segmentation capability of the self-training model in its early stages.

Given the aforementioned works and the powerful segmentation capability of SAM, we introduce SAM into our weakly-supervised RGB-D SOD method, expanding scribbles to pixel-level pseudo annotations for dense supervision.

D. Conditional Diffusion Model

Diffusion model [27] is a type of probability generation model that includes diffusion and denoising processes. In the forward diffusion process, Gaussian noise is gradually added to the ground truth until pure noise is obtained. In the reverse denoising process, the model learns denoising, iteratively refines from random noise, and restores the ground truth. On this basis, conditional diffusion models introduce conditional information, constraining the generation of results that meet the conditions. Due to the superior performance, conditional diffusion models have attracted great attention and have shown great potential in computer vision, such as underwater image enhancement [28], camouflaged object detection [29], [30], semantic segmentation [31], and remote sensing image salient object detection [32], [33]. Qing *et al*. [28] established an enhanced diffusion model, using the representation of underwater image degradation as a prior condition and regulating feature weights by frequency. Yang *et al*. [29] explicitly modeled uncertainty and leveraged it to guide the denoising process and feature aggregation, developing the accuracy of prediction in areas of high uncertainty. Wu *et al*. [31] utilized an existing stable diffusion model to generate image-semantic mask pairs with text-guided cross-attention map and domain gap between synthetic and real data. Han *et al*. [32] reformulated ORSI-SOD as a conditional guided mask generation task based on the diffusion model, introducing a consistency assessment strategy in the denoising process to mitigate the issue of overconfidence. Hou *et al*. [33] conducted frequency-domain perception and multi-level feature coordination in the denoising network of the conditional diffusion model for effective denoising.

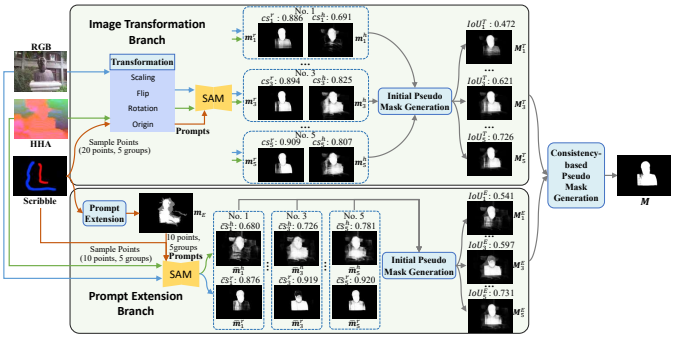

Fig. 1. Illustration of the SAM-driven pseudo annotation method. Please zoom in for details.

Inspired by the above works, we introduce the conditional diffusion model into RGB-D SOD to generate accurate saliency map with progressive refinement. Moreover, we adopt the state space interaction-based method, producing the conditional feature and injecting conditional information into denoising process effectively and efficiently.

III. SAM-driven Pseudo Annotation Generation

Our training dataset is $D = \{(X_i, \boldsymbol{y}_i)\}_{i=1}^{N}$, where $X_i = \{\boldsymbol{r}_i, \boldsymbol{d}_i\}$ is input RGB-D pair including RGB image $\boldsymbol{r}_i$ and depth map $\boldsymbol{d}_i$, and $\boldsymbol{y}_i$ is the scribble annotation. The index i will be omitted when it is unambiguous.

A. Motivation and Overview

SAM [15] is an advanced visual foundation model focused on the segmentation task. There are two kinds of prompts for SAM, including the sparse prompts (*i.e*., points, boxes, and text) and the dense prompt (*i.e*., masks). It is pre-trained on the broad SA-1B dataset, and shows great generalizability and applicability for solving general downstream segmentation tasks via prompt engineering. Due to the strong segmentation ability of SAM, we introduce it into the weakly-supervised RGB-D SOD. We aim to use SAM with the scribble annotations to generate pixel-level pseudo annotations, alleviating the problem of insufficient supervision information in model training. Therefore, we propose a SAM-driven pseudo annotation framework, as shown in Fig. 1.

Our SAM-driven pseudo annotation framework consists of an image transformation branch and a prompt extension branch. The former one focuses on generating complementary segmentation masks through various image transformations, while the latter one focuses on extending the potential sampling area. Considering the input prompt form of SAM and the existing weak annotation (*i.e*., scribble), we naturally sample points from scribble as prompts in our SAM-driven pseudo annotation framework. Moreover, in our framework, we propose a consistency-based pseudo mask generation method to fuse the initial pseudo masks generated from the above branches, producing a reliable final pixel-level pseudo mask. In the following, we elaborate on our SAM-driven pseudo annotation framework in three parts. Notably, to facilitate SAM to segment depth maps and to enhance the expression of

deep information, we encode the raw depth maps $\boldsymbol{d}$ to HHA maps $\boldsymbol{h}$ [34].

B. Image Transformation Branch

Although SAM has strong generalizability, it is sensitive to image transformations, that is, SAM has the transformation variability. As depicted in the first row of Fig. 2, we input the same RGB image (or HHA map) with the same point prompt into SAM, but perform different image transformations (such as scaling, flip, and rotation) on them, outputting different segmentation masks. For example, in the mask of HHA map with *Rotation* $90\,^{\circ}$, the top of the left object (*i.e.*, the yellow dashed circle) is indistinct, while other masks of the same HHA map are clear. In the mask of RGB image with *Flip*, the prediction of the right object (*i.e.*, the yellow dashed circle) is less complete than other masks of the same RGB image. Moreover, as depicted in the second row of Fig. 2, we input the same RGB image (or HHA map) but with different point prompts sampled from scribbles into SAM, outputting different segmentation masks. We observe that the quality of segmentation masks varied greatly under different sample points. Inspired by the above observations, we propose the image transformation branch to integrate masks of different image transformations and different point prompts, overcoming the transformation variability of SAM and enhancing the completeness of pseudo masks.

As the image transformation branch shown in Fig. 1, we first sample 20 points from the scribble annotation $\boldsymbol{y}$ as prompts $\boldsymbol{p}^T$. Then, we perform three image transformations on $\{\boldsymbol{r}, \boldsymbol{h}, \boldsymbol{p}^T\}$, including scaling ($\times 0.5$), rotation ($90\,^{\circ}$ counter-clockwise), and flip (horizontal direction). Combined with the original RGB-HHA pair, we get four sets of RGB-HHA pairs. The above four sets of RGB-HHA pairs are fed into SAM with corresponding transformed or original point prompts, respectively. Taking the RGB-HHA pair with scaling as an example, SAM respectively processes the scaled {RGB, prompts} and {HHA, prompts}, producing the segmentation mask and confidence score of RGB image and HHA map. The confidence score is estimated by the IoU prediction head in the mask decoder of SAM to evaluate the mask quality. Next, we perform the inverse image transformations of scaling ($\times 0.5$) on the output segmentation masks of the RGB-HHA pair, getting the inverse segmentation masks. After processing these four sets of RGB-HHA pairs in the above way, we get four segmentation masks and confidence scores (*i.e.*, scaling, rotation, flip, and original) for the RGB image and HHA map, respectively. We then average these four segmentation masks and confidence scores of the RGB image, obtaining $\boldsymbol{m}^r$ and cs^r. We perform the same operation on these four segmentation masks and confidence scores of the HHA map, obtaining $\boldsymbol{m}^h$ and cs^h. In addition, in this branch, we sample 20 points from the scribble annotation five times to ensure the diversity of the point prompts. According to each group of point prompts, we adopt SAM to segment RGB-HHA pair using the above operations. Therefore, as shown in Fig. 1, we obtain five sets of segmentation masks and confidence scores for RGB image and HHA map, *i.e.*, $\left\{\boldsymbol{m}_l^r, cs_l^r, \boldsymbol{m}_l^h, cs_l^h\right\}_{l=1}^{5}$.

Then, for each group of point prompts, we propose an initial pseudo mask generation method to fuse its segmentation masks based on its confidence scores to enhance the completeness of masks. Specifically, the core of our initial pseudo mask generation method is weighted fusion, which can be formulated as follows:

$$\boldsymbol{M}_l^T = \frac{cs_l^r \cdot \boldsymbol{m}_l^r + cs_l^h \cdot \boldsymbol{m}_l^h}{cs_l^r + cs_l^h}, \tag{1}$$

where $\boldsymbol{M}_l^T$ is initial pseudo mask of $l\text{-}th$ group of point prompts. Here, we also calculate the consistency score IoU_l^T between $\boldsymbol{m}_l^r$ and $\boldsymbol{m}_l^h$ to represent the cross-modal consistency via intersection over union for subsequent processing as follows:

$$IoU_l^T = \frac{|\boldsymbol{m}_l^r| \cap |\boldsymbol{m}_l^h|}{|\boldsymbol{m}_l^r| \cup |\boldsymbol{m}_l^h| + \varepsilon}, \tag{2}$$

where $|\cdot|$ means the image binarization operation with a threshold of 0.5 and $\varepsilon = 10^{-6}$. In this way, our image transformation branch can produce relatively complete initial pseudo masks.

C. Prompt Extension Branch

Since the scribble annotation only occupies a very small number of pixels, sampling point prompts from it has significant limitations in providing comprehensive object and background information. Therefore, we propose the prompt extension branch to reasonably extend the potential area of sampling point prompts. As shown in Fig. 1, the prompt extension branch is different from the image transformation branch in the former part, while the latter part is basically the same as the image transformation branch.

As the core of the former part of the prompt extension branch, we first introduce the prompt extension method, which reasonably extends the area of scribble annotation via superpixel clustering. As shown in Fig. 3, in our prompt extension method, we first adopt the SLIC [35], a classic superpixel segmentation algorithm, to segment RGB image and HHA map, respectively, into two sets of 70 superpixels. Superpixel segmentation groups pixels into perceptually similar regions via clustering [35]. Thus, combining with the scribble annotation, we label superpixels that intersect with the foreground scribble as foreground, and superpixels that intersect with the background scribble as background. We can obtain foreground and background masks, *i.e.*, $\{\boldsymbol{m}_f^r, \boldsymbol{m}_b^r\}$ for RGB image and $\{\boldsymbol{m}_f^h, \boldsymbol{m}_b^h\}$ for HHA map. Then, we perform the conflict optimization on the foreground mask with the background mask. Taking $\{\boldsymbol{m}_f^h, \boldsymbol{m}_b^h\}$ for HHA map as an example, if the pixel in $\boldsymbol{m}_f^h$ is labeled as foreground, but the pixel at the same position in $\boldsymbol{m}_b^h$ is labeled as background, we believe the pixel at this position is conflicting and meaningless. We discard the pixel at this position in $\boldsymbol{m}_f^h$, that is, we will not sample points in these conflict areas in the subsequent processing. We mark these conflict areas in gray. In this way, we obtain the extended mask $\hat{\boldsymbol{m}}_f^h$ with conflict areas. By performing the same operations on $\{\boldsymbol{m}_f^r, \boldsymbol{m}_b^r\}$, we obtain $\hat{\boldsymbol{m}}_f^r$ for RGB image. To make the extended mask more precise, we perform the coexistence optimization on $\hat{\boldsymbol{m}}_f^r$ and $\hat{\boldsymbol{m}}_f^h$. Simply put, we

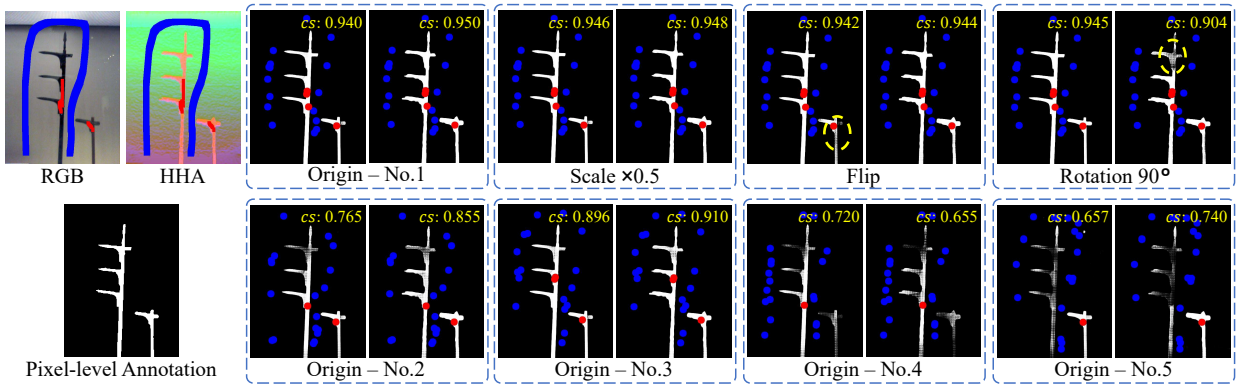

Fig. 2. Output segmentation masks and confidence scores (cs) of SAM with different transformed images (the first row) and different sample point prompts (No.1 to No.5). In each blue dashed box, the left and right segmentation masks correspond to RGB image and HHA map, respectively. The blue points are the background points, while the red ones are the foreground points.

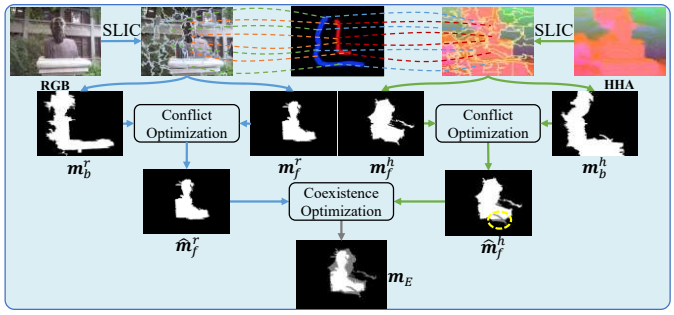

Fig. 3. Illustration of the prompt extension. Please zoom in for details.

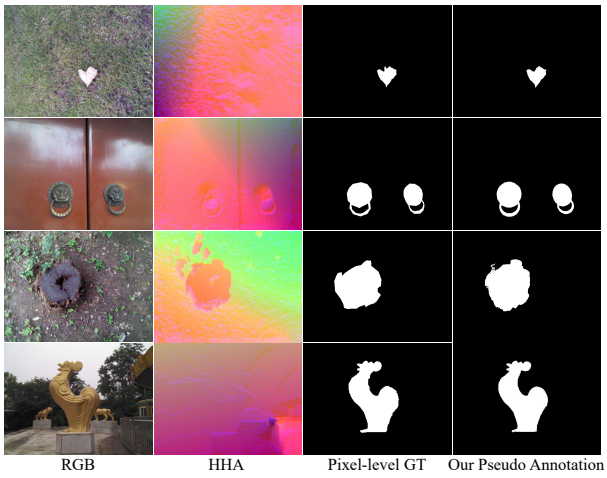

Fig. 4. Visualization of our pseudo annotations.

only keep the pixels with the same labels in two extended masks $\{\hat{\boldsymbol{m}}_f^r, \hat{\boldsymbol{m}}_f^h\}$ (*i.e.*, keeping the intersection of these two masks), while treating inconsistent areas as conflict areas. The inconsistent areas are marked in gray, and combine with the conflict areas of $\hat{\boldsymbol{m}}_f^r$ and $\hat{\boldsymbol{m}}_f^h$ to form comprehensive gray areas. In this way, we obtain the final extended annotation $\boldsymbol{m}_E$, making the distribution of sampling point areas wider.

As the prompt extension branch shown in Fig. 1, after getting $\boldsymbol{m}_E$, we sample point prompts from $\boldsymbol{m}_E$ except the gray areas and the original scribble annotation $\boldsymbol{y}$ simultaneously, *i.e.*, 10 points from $\boldsymbol{m}_E$ and 10 points from $\boldsymbol{y}$, obtaining prompts $\boldsymbol{p}^E$. In this branch, we directly adopt SAM to respectively process original {RGB, prompts} and {HHA, prompts}, producing $\{\bar{\boldsymbol{m}}^r, \bar{cs}^r\}$ for RGB image and $\{\bar{\boldsymbol{m}}^h, \bar{cs}^h\}$ for HHA map. Similar to the image transformation branch, we sample points from $\boldsymbol{m}_E$ and $\boldsymbol{y}$ five times to ensure the diversity of the point prompts. In this way, as shown in Fig. 1, we obtain five sets of segmentation masks and confidence scores for RGB image and HHA map, *i.e.*, $\{\bar{\boldsymbol{m}}_l^r, \bar{cs}_l^r, \bar{\boldsymbol{m}}_l^h, \bar{cs}_l^h\}_{l=1}^5$. We adopt the initial pseudo mask generation method to fuse them and calculate consistency scores, getting $\{\boldsymbol{M}_l^E\}_{l=1}^5$ and $\{IoU_l^E\}_{l=1}^5$. In this way, our prompt extension branch can produce diverse initial pseudo masks.

D. Consistency-based Pseudo Mask Generation

As initial pseudo masks and their consistency scores shown in Fig. 1, we observe that the initial pseudo mask with a higher consistency score usually exhibits better quality and provides more accurate supervision information. For example, the No.5 mask pair in both branches in Fig. 1 has the higher consistency score, and its initial pseudo mask is more accurate. Therefore, we propose the consistency-based pseudo mask generation method. We rank these ten initial pseudo masks ($\{\boldsymbol{M}_l^T\}_{l=1}^5$ and $\{\boldsymbol{M}_l^E\}_{l=1}^5$) according to their consistency scores ($\{IoU_l^T\}_{l=1}^5$ and $\{IoU_l^E\}_{l=1}^5$) in descending order, generating $\{\boldsymbol{M}_i^{rank}, IoU_i^{rank}\}_{i=1}^{10}$. Then, we only select four initial pseudo masks with the top four consistency scores, and perform weighted fusion on them. Finally, a denseCRF [36] post-processing method is applied to obtain the final pseudo mask $\boldsymbol{M}$ as follows:

$$\boldsymbol{M} = CRF\Big(\Big|\frac{\sum_{i=1}^{4} \boldsymbol{M}_i^{rank} \cdot IoU_i^{rank}}{\sum_{i=1}^{4} IoU_i^{rank}}\Big|\Big). \tag{3}$$

In this way, the final pseudo mask $\boldsymbol{M}$ can provide rich supervision information of salient objects and background, alleviating the problem of insufficient supervision information of scribble annotation. We also provide some representative generated pseudo annotations in Fig. 4, covering some challenging cases, such as small objects (1st row), multiple objects

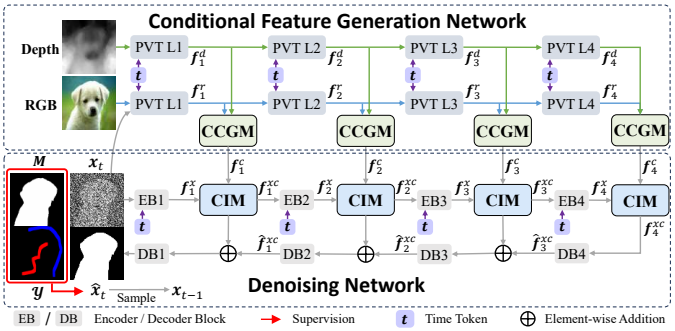

Fig. 5. Overview of the our S^2*Diff*. Please zoom in for details.

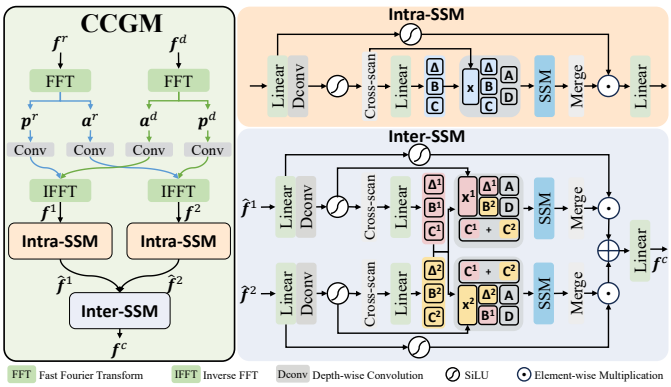

Fig. 6. Illustration of Cross-modal Conditional Generation Module.

(2nd row), low contrast (3rd row), and low-quality depth map (4th row). These generated pseudo annotations are overall highly consistent with the pixel-level ground truth, and can provide sufficient pixel-level supervision information.

IV. State Space Interaction-based Diffusion

A. Network Overview

To generate accurate saliency maps, we propose S^2Diff as illustrated in Fig. 5, consisting of a conditional feature generation network and a denoising network. Specifically, in the conditional feature generation network, we adopt Pyramid Vision Transformer (PVT) [37] to extract features from the RGB image and the depth map, respectively, and the noisy mask $\boldsymbol{x}_t$ is injected in RGB branch. The RGB features and depth features are denoted as $\boldsymbol{f}_i^r$ and $\boldsymbol{f}_i^d$ ($i \in \{1,2,3,4\}$). Then the Cross-modal Conditional Generation Module (CCGM) integrates corresponding RGB and depth features at each layer, forming informative cross-modal conditional features $\boldsymbol{f}_i^c$. Denoising network is designed to remove the noise from $\boldsymbol{x}_t$ and output the denoised mask $\hat{\boldsymbol{x}}_t$ with a four-level encoder-decoder architecture and Context Injection Modules (CIMs). Next, we describe the effect and structure of CCGM and CIM.

B. Cross-modal Conditional Generation Module

Existing methods usually aggregate cross-modal features through CNN or Transformer-based modules. Obviously, CNN-based modules [2], [3] are hard to model long-range dependencies, while Transformer-based modules [8], [11] suffer from huge computational overhead. Mamba [38], [39] achieves a good balance between global modeling capability and computational complexity. It selectively processes information by parameterizing state space model (SSM) to focus on or ignore particular inputs and implements a global receptive field with linear complexity. Therefore, we propose a Cross-modal Conditional Generation Module, which adopts SSM to produce global conditional features effectively and efficiently, capturing the global information from cross-modal features. Moreover, considering that RGB features usually provide rich appearance cues while depth features contain more structural information, they exhibit different frequency-domain characteristics. So frequency-domain exchange is introduced in CCGM to align features of different modalities, allowing RGB and depth features to mutually enhance each other and produce more robust representations. As shown in Fig. 6, our CCGM consists of three stages of frequency-domain exchange, intra-SSM, and inter-SSM. Specifically, we categorize frequency-domain exchange and intra-SSM as implicit state space interaction, and inter-SSM as explicit state space interaction.

1) Implicit State Space Interaction: Implicit state space interaction is composed of frequency-domain exchange and intra-SSM. As for frequency-domain exchange, we first perform the Fast Fourier Transform (FFT) on $\boldsymbol{f}^r$ and $\boldsymbol{f}^d$, separately, then exchange and recombine their amplitudes (*i.e.*, $\boldsymbol{a}^r$ and $\boldsymbol{a}^d$) and phases (*i.e.*, $\boldsymbol{p}^r$ and $\boldsymbol{p}^d$), forming integrated features of spatial domain through inverse FFT (IFFT). The frequency-domain exchange can be formulated as:

$$\boldsymbol{a}^{r/d} = \left|\mathcal{F}(\boldsymbol{f}^{r/d})\right|,\ \boldsymbol{p}^{r/d} = \angle\mathcal{F}(\boldsymbol{f}^{r/d}), \tag{4}$$

$$\begin{aligned} \boldsymbol{f}^1 &= \mathcal{F}^{-1}(Conv(\boldsymbol{a}^d), Conv(\boldsymbol{p}^r))), \\ \boldsymbol{f}^2 &= \mathcal{F}^{-1}(Conv(\boldsymbol{a}^r), Conv(\boldsymbol{p}^d))), \end{aligned} \tag{5}$$

where $\mathcal{F}(\cdot)$ is FFT, $\mathcal{F}^{-1}(\cdot)$ is IFFT, and $Conv(\cdot)$ is the convolutional layer. In this way, we align the cross-modal features in the frequency domain. Then, we perform intra-SSM on $\boldsymbol{f}^1$ and $\boldsymbol{f}^2$ to achieve implicit state space interaction.

As shown in the upper right corner of Fig. 6, our intra-SSM follows the structure of Mamba block [38]. Taking $\boldsymbol{f}^1$ as an example, $\boldsymbol{f}^1$ first passes through linear projection (Linear), depth-wise convolution (Dconv), and SiLU activation function. Then, an SS2D module is used, which includes four directional traversal (Cross-scan), selective SSM (Linear and SSM), and four directional fusion (Merge). In selective SSM, the parameters depend on $\boldsymbol{f}^1$, guaranteeing that the output contains information about each feature value of $\boldsymbol{f}^1$. Moreover, $\boldsymbol{f}^1$ passes a Linear and SiLU function, and then is multiplied by the output of SS2D as a gating signal. Finally, a linear layer aligns features to their original dimensions, generating $\hat{\boldsymbol{f}}^1$. In the same way, we can get $\hat{\boldsymbol{f}}^2$.

2) Explicit State Space Interaction: Implicit state space interaction alone is insufficient, and we therefore perform explicit state space interaction on $\hat{\boldsymbol{f}}^1$ and $\hat{\boldsymbol{f}}^2$ in inter-SSM. As illustrated in the lower right corner of Fig. 6, there are two symmetric branches processing each input respectively. In each branch, the operations are basically consistent with those in intra-SSM, and the main difference lies in the control

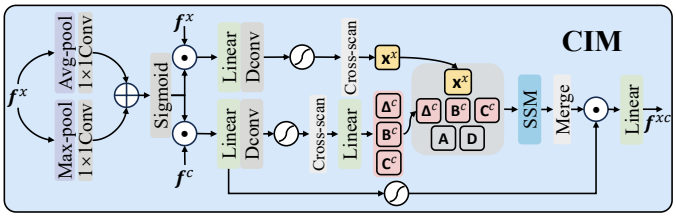

Fig. 7. Illustration of Context Injection Module.

of the parameters. In particular, we exchange $\boldsymbol{B}^1$ and $\boldsymbol{B}^2$, add $\boldsymbol{C}^1$ and $\boldsymbol{C}^2$ as parameter $\boldsymbol{C}$, keep $\boldsymbol{\Delta}^1$ and $\boldsymbol{\Delta}^2$ unchanged, and $\boldsymbol{A}$ and $\boldsymbol{D}$ are shared in two branches. Taking the first branch as an example, the process can be formulated as:

$$\bar{\boldsymbol{A}}^1 = \exp(\boldsymbol{\Delta}^1 \boldsymbol{A}),\ \bar{\boldsymbol{B}}^1 = \boldsymbol{\Delta}^1 \boldsymbol{B}^2, \tag{6}$$

$$\begin{aligned} \boldsymbol{h}_k &= \bar{\boldsymbol{A}}^1 \boldsymbol{h}_{k-1} + \bar{\boldsymbol{B}}^1 \boldsymbol{x}_k, \\ \boldsymbol{y}_k &= (\boldsymbol{C}^1 + \boldsymbol{C}^2)\boldsymbol{h}_k + \boldsymbol{D}\boldsymbol{x}_k. \end{aligned} \tag{7}$$

Eq. 6 is the discretization process, and Eq. 7 is the modeling process in SSM, where the superscript denotes the parameter is controlled by which feature. By interactively or collectively manipulating SSM parameters, inter-SSM stimulates complementarity and fusion between features. After the process of two branches, their outputs are added together and linearly mapped to the original feature dimension, generating $\boldsymbol{f}^c$.

Through implicit and explicit interactions, we can comprehensively model the global information in cross-modal features, providing conditional context to the denoising network.

C. Context Injection Module

To take full advantage of conditional context, we introduce the SSM-based CIM into the denoising network to regulate the reconstruction of noise features into semantically explicit features. The structure of CIM is shown in Fig. 7. We first modulate the noise feature $\boldsymbol{f}^x$ and the condition feature $\boldsymbol{f}^c$ with the channel attention map [40] generated from $\boldsymbol{f}^x$, generating $\hat{\boldsymbol{f}}^x$ and $\hat{\boldsymbol{f}}^c$. The channel attention map determines important channels about objects from the noise feature, allowing $\hat{\boldsymbol{f}}^x$ and $\hat{\boldsymbol{f}}^c$ to focus more on objects. The modulation process is shown as follows:

$$\hat{\boldsymbol{f}}^{x/c} = CA(\boldsymbol{f}^x) \odot \boldsymbol{f}^{x/c}, \tag{8}$$

where $\odot$ is element-wise multiplication and $CA(\cdot)$ is the channel attention operation.

!t]

Then, $\hat{\boldsymbol{f}}^x$ and $\hat{\boldsymbol{f}}^c$ are input into our renovated selective SSM, where $\hat{\boldsymbol{f}}^c$ is responsible for parameter controlling and $\hat{\boldsymbol{f}}^x$ are modeled by SSM, which can be described as follows:

$$\bar{\boldsymbol{A}}^C = \exp(\boldsymbol{\Delta}^C \boldsymbol{A}), \bar{\boldsymbol{B}}^C = \boldsymbol{\Delta}^C \boldsymbol{B}^C. \tag{9}$$

$$\begin{aligned} \boldsymbol{h}_k &= \bar{\boldsymbol{A}}^C \boldsymbol{h}_{k-1} + \bar{\boldsymbol{B}}^C \boldsymbol{x}_k^x, \\ \boldsymbol{y}_k &= \boldsymbol{C}^C \boldsymbol{h}_k + \boldsymbol{D}\boldsymbol{x}_k, \end{aligned} \tag{10}$$

where the superscripts of parameters are C because they depend on $\hat{\boldsymbol{f}}^c$, and only $\boldsymbol{x}^x$ is from $\hat{\boldsymbol{f}}^x$. In this way, we utilize $\boldsymbol{f}^c$ to guide the modeling of $\boldsymbol{f}^x$ to achieve the injection of conditional context into the denoising process.

D. Loss Function

We train our S^2Diff with the supervision of scribble annotation $\boldsymbol{y}$ and generated pixel-level pseudo annotation $\boldsymbol{M}$ as:

$$L = \underbrace{\ell_{pce}(\hat{\boldsymbol{x}}_t, \boldsymbol{y})}_{\text{sparse supervision}} + \underbrace{\ell_{bce}^w(\hat{\boldsymbol{x}}_t, \boldsymbol{M}) + \ell_{iou}^w(\hat{\boldsymbol{x}}_t, \boldsymbol{M})}_{\text{dense supervision}}, \tag{11}$$

where ℓ_{pce} is the partial cross-entropy loss [65] with $\boldsymbol{y}$ as supervision, and ℓ_{bce}^w and ℓ_{iou}^w are the weighted binary cross-entropy loss and the weighted intersection-over-union loss, respectively, with $\boldsymbol{M}$ as supervision.

V. Experiments

A. Experimental Protocol

1) Datasets. We use the scribble-annotated RGB-D datasets re-annotated by Xu *et al*. [10], containing 1485 images from NJU2K [64] and 700 images from NLPR [42] as the training dataset. For the test dataset, we use the seven RGB-D SOD datasets, including DUT [43], LFSD [44], NJU2K [64], NLPR [42], SIP [62], SSD [63], and STERE [41]. In NJU2K and NLPR, images except the raining data are used for testing.

2) Evaluation Metrics. We perform the quantitative evaluation of the model performance on five commonly used metrics, *i.e*., mean absolute error (M), average F-measure (F_m) [66], average E-measure (E_m) [67], S-measure (S_m) [68] and weighted F-measure (F_β^ω) [69]. In addition, we adopt the parameter count and the computational cost to evaluate the model complexity, and the lower they are, the better.

3) Implementation Details. We implement our method with PyTorch on an NVIDIA GTX 4090 GPU. For our SAM-PAG, we use ViT-H SAM to generate masks. Since SAM-PAG involves random point sampling, we fix the random seed to 3 for all stochastic operations. For prompt sampling, points are randomly sampled from the scribble annotations. For extended annotation, we additionally impose a constraint, where the number of foreground points is not smaller than the number of background points, to avoid prompts dominated by the background. When the pixels of scribbles or extended annotations are fewer than the required sampling points, repeated sampling is allowed. We follow the standard HHA processing pipeline and the default parameter settings [34] to convert depth maps into HHA representations. DenseCRF is applied to refine the fused pseudo annotations. The unary term is constructed from the fused binary pseudo annotations with a confidence of 0.7. We use a two-label foreground/background CRF and perform 10 mean field iterations. The spatial Gaussian term uses a spatial standard deviation of 3 and a compatibility weight of 3. The RGB-guided bilateral term uses spatial/color standard deviations of 30/5 and a compatibility weight of 5, while the HHA-guided bilateral term uses spatial/HHA-value standard deviations of 15/5 and a compatibility weight of 3.

For our S^2Diff, the backbone of the conditional feature generation network utilizes PVT-v2-B4 for initialization. In the training and testing phases, we resize the input RGB-D pair to 352×352. We adopt AdamW optimizer with an initial learning rate of $1e^{-4}$, and set the batch size to 8 and the training epoch to 150. We adopt several data augmentation strategies, such as random flipping, rotation, and color jittering. We set the

TABLE I
QUANTITATIVE COMPARISONS ON STERE, NLPR, DUT, AND LFSD DATASETS. THE BEST RESULTS OF FULLY-SUPERVISED METHODS AND WEAKLY-SUPERVISED METHODS IN EACH COLUMN ARE HIGHLIGHTED IN **RED**.

Method	Publication	Input Size	Param (M)↓	FLOPs (G)↓	STERE [41] $M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_\beta^\omega\uparrow$	NLPR [42] $M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_\beta^\omega\uparrow$	DUT [43] $M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_\beta^\omega\uparrow$	LFSD [44] $M\downarrow$	$F_m\uparrow$
Fully-supervised Methods																					
DSNet [45]	2021 TIP	288^2	-	-	.036	.894	.939	.915	.882	.024	.907	.943	.926	.886	.079	.807	.857	.841	.774	.069	.848
DCFNet [46]	2021 CVPR	352^2	108.5	107.8	.039	.886	.939	.901	.875	.021	.898	.957	.923	.892	.071	.812	.888	.836	.766	.075	.835
CIRNet [47]	2022 TIP	352^2	103.1	22.5	.039	.890	.932	.915	.872	.023	.900	.952	.933	.889	.031	.923	.949	.932	.904	.068	.867
CFIDNet [48]	2022 NCA	320^2	53.9	42.9	.043	.881	.933	.901	.867	.026	.891	.947	.922	.881	.039	.903	.940	.916	.887	.071	.849
SwinNet [49]	2022 TCSVT	384^2	198.7	124.3	.033	.895	.947	.919	.889	.018	.919	.966	.941	.913	.021	.942	.969	.948	.933	.059	.874
DIGRNet [50]	2023 TMM	352^2	201.8	68.2	.038	.891	.940	.916	.877	.023	.904	.954	.935	.895	.033	.920	.946	.926	.898	.067	.851
C2DFNet [51]	2023 TMM	256^2	47.5	22.0	.038	.881	.937	.902	.871	.021	.905	.955	.927	.897	.026	.932	.958	.933	.918	.065	.859
HINet [52]	2023 PR	352^2	98.9	389.7	.049	.859	.918	.892	.839	.026	.887	.945	.922	.876	.054	.854	.903	.884	.826	.076	.829
CAVER [53]	2023 TIP	256^2	93.8	63.9	.033	.896	.947	.913	.889	.020	.906	.961	.928	.901	.042	.892	.932	.903	.874	.063	.864
PICRNet [54]	2023 MM	224^2	106.8	121.3	.031	.905	.951	.920	.898	.019	.916	.965	.935	.911	.021	.943	.967	.943	.933	.053	**.884**
CATNet [55]	2024 TMM	384^2	262.6	341.8	.030	.904	.952	.921	.900	.018	.922	.966	.940	.916	.020	.950	.971	**.953**	.942	.051	.878
RD3D+ [56]	2024 TNNLS	352^2	28.9	43.3	.039	.880	.932	.914	.867	.022	.898	.953	.933	.889	.031	.923	.952	.936	.908	.076	.831
LAFB [57]	2024 TCSVT	352^2	451.8	137.6	.040	.886	.936	.899	.870	.024	.901	.955	.924	.894	.032	.920	.953	.926	.906	.065	.857
MAGNet [54]	2024 KBS	384^2	16.1	9.9	.030	.904	.951	.922	.892	.018	.918	.965	.939	.908	.021	.944	.967	.943	.935	.054	.878
CPNet [58]	2024 IJCV	384^2	216.5	129.3	**.029**	.903	**.954**	.920	.901	**.016**	**.925**	.969	.940	**.922**	**.019**	**.953**	**.972**	.951	**.948**	**.050**	**.884**
FasterSal [59]	2025 TMM	256^2	3.4	0.9	.040	.873	.937	.888	.866	.022	.900	.957	.920	.898	.031	.921	.954	.920	.909	.063	.850
EM-Trans [60]	2025 TNNLS	352^2	-	-	**.029**	**.913**	.953	.925	**.905**	.017	.920	.965	.940	.917	-	-	-	-	-	-	-
HENet [61]	2025 TCSVT	384^2	11.9	10.7	**.029**	**.913**	.952	**.926**	.903	**.016**	.922	**.970**	**.942**	.918	.020	.945	.971	.949	.938	**.050**	.883
Weakly-supervised Methods																					
DENet [10]	2022 TIP	352^2	37.1	95.3	.048	.857	.927	.880	.839	.031	.868	.942	.900	.856	.072	.814	.880	.846	.768	.085	.817
DHFR [11]	2023 TIP	224^2	54.5	-	.043	.879	.932	.884	.861	.027	.882	.946	.904	.871	.052	.870	.915	.864	.839	.066	.858
MIRV [12]	2024 TCSVT	352^2	63.6	37.1	.042	.872	.934	.891	.859	.025	.894	**.952**	**.914**	.883	.054	.862	.908	.877	.833	.072	.843
RPPS [13]	2024 TPAMI	256^2	53.4	23.1	.059	.835	.894	.881	.801	.035	.846	.916	.899	.814	.067	.832	.887	.877	.784	.095	.807
Ours	2026 TMM	352^2	169.0	559.0	**.036**	**.906**	**.943**	**.905**	**.891**	**.024**	**.895**	**.952**	**.914**	**.887**	**.049**	**.891**	**.926**	**.893**	**.870**	**.064**	**.872**

TABLE II
QUANTITATIVE COMPARISONS ON LFSD, SIP, SSD, AND NJU2K DATASETS.

Method	LFSD [44] $E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	SIP [62] $M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	SSD [63] $M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	NJU2K [64] $M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	**Average** $M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$
Fully-supervised Methods																							
DSNet [45]	.889	.868	.826	.052	.863	.910	.876	.840	.045	.859	.906	.885	.838	.034	.907	.943	.921	.898	.048	.868	.913	.890	.849
DCFNet [46]	.878	.841	.805	.051	.875	.916	.875	.848	.049	.836	.903	.864	.814	.035	.903	.944	.911	.893	.049	.864	.918	.879	.842
CIRNet [47]	.890	.875	.838	.052	.875	.911	.888	.848	.049	.840	.897	.878	.816	.035	.908	.940	.925	.895	.042	.886	.924	.907	.866
CFIDNet [48]	.894	.870	.828	.060	.856	.899	.864	.825	.050	.850	.914	.879	.829	.038	.898	.937	.914	.886	.047	.875	.923	.895	.858
SwinNet [49]	.912	.886	.854	.035	.912	.942	.911	.896	.040	.865	.917	.892	.851	.027	.923	.955	.934	.917	.033	.904	.944	.919	.893
DIGRNet [50]	.892	.873	.828	.053	.879	.913	.885	.849	.053	.830	.889	.866	.804	.028	.918	.952	.933	.909	.042	.885	.927	.905	.866
C2DFNet [51]	.897	.863	.835	.052	.865	.912	.871	.841	.047	.847	.911	.872	.827	.038	.898	.936	.907	.885	.041	.884	.929	.896	.868
HINet [52]	.877	.852	.802	.066	.839	.886	.856	.805	.049	.836	.899	.865	.808	.039	.895	.933	.915	.881	.051	.857	.909	.884	.834
CAVER [53]	.907	.873	.844	.042	.889	.931	.892	.872	.041	.850	.919	.878	.834	.031	.914	.950	.920	.906	.039	.887	.935	.901	.874
PICRNet [54]	.917	.888	.864	.040	.901	.934	.898	.883	.047	.847	.917	.874	.832	.029	.919	.952	.927	.912	.034	.902	.943	.912	.890
CATNet [55]	.921	.894	.863	.035	.914	.944	.911	.897	-	-	-	-	-	.026	.927	.956	.932	.922	-	-	-	-	-
RD3D+ [56]	.876	.861	.807	.047	.881	.917	.891	.857	.044	.841	.900	.882	.820	.033	.909	.943	.927	.899	.042	.880	.925	.906	.864
LAFB [57]	.899	.864	.867	.051	.877	.918	.876	.850	.041	.864	.916	.882	.842	.033	.906	.945	.916	.897	.041	.887	.932	.898	.870
MAGNet [54]	.918	.889	.859	.037	.912	.943	.908	.888	.043	.858	.924	.885	.836	.028	.923	.956	.929	.911	.032	.905	.946	.916	.896
CPNet [58]	.921	.893	.869	.035	**.916**	.941	.907	**.900**	**.035**	**.876**	.930	**.894**	**.864**	**.025**	**.931**	**.959**	**.935**	**.926**	**.030**	**.913**	**.949**	**.920**	**.904**
FasterSal [59]	.901	.859	.835	.049	.868	.926	.870	.852	.044	.844	.929	.866	.831	.034	.903	.946	.908	.899	.040	.880	.936	.890	.870
EM-Trans [60]	-	-	-	.039	.910	.937	.903	.892	.039	.862	.931	.886	.847	.027	.925	.955	.930	.918	-	-	-	-	-
HENet [61]	**.924**	**.900**	**.870**	**.032**	**.916**	**.946**	**.914**	**.900**	.036	.863	**.932**	.893	.851	.027	.922	.955	.934	.916	.031	.902	.947	.918	.892
Weakly-supervised Methods																							
DENet [10]	.873	.832	.788	.061	.834	.907	.852	.809	.068	.796	.876	.831	.765	.049	.867	.923	.882	.849	.059	.836	.904	.861	.811
DHFR [11]	.901	.855	.835	.064	.851	.899	.842	.816	.051	.846	.905	.858	.821	**.040**	.888	**.936**	**.893**	**.875**	.049	.868	.919	.871	.845
MIRV [12]	.889	.849	.814	**.049**	.862	**.923**	**.876**	.844	.055	.826	.900	.855	.801	.046	.879	.929	.890	.864	.049	.863	.919	.879	.843
RPPS [13]	.839	.835	.763	.060	.843	.898	**.876**	.809	.061	.812	.874	**.864**	.774	.065	.840	.896	.871	.805	.063	.831	.886	.872	.793
Ours	**.902**	**.869**	**.849**	.050	**.892**	.918	.871	**.863**	**.047**	**.854**	**.917**	.856	**.829**	.046	**.889**	.928	.887	.872	**.045**	**.886**	**.927**	**.885**	**.866**

training random seed to 42 for network initialization and data augmentation. An SNR-based noise schedule [30] is used to generate noise in the forward diffusion process, and the model is trained to directly predict the clean saliency map using Eq. 11. In the sampling process, we set the time steps to 10.

B. Comparison with State-of-the-arts

We compare our method with 18 fully-supervised RGB-D SOD methods, including DSNet [45], DCFNet [46], CIRNet [47], CFIDNet [48], SwinNet [49], DIGRNet [50], C2DFNet [51], HINet [52], CAVER [53], PICRNet [54], CATNet [55], RD3D+ [56], LAFB [57], MAGNet [70], CPNet [58], FasterSal [59], EM-Trans [60], and HENet [61], and 4 weakly-supervised RGB-D SOD methods, including DENet [10], DHFR [11], MIRV [12], and RPPS [13]. For most fully-supervised methods and all weakly-supervised methods, we directly use their released saliency maps or generate saliency maps using the released models and weights. For other methods without released saliency maps or weights, we retrain them according to the official implementations. Then, all saliency maps are evaluated with the same evaluation code

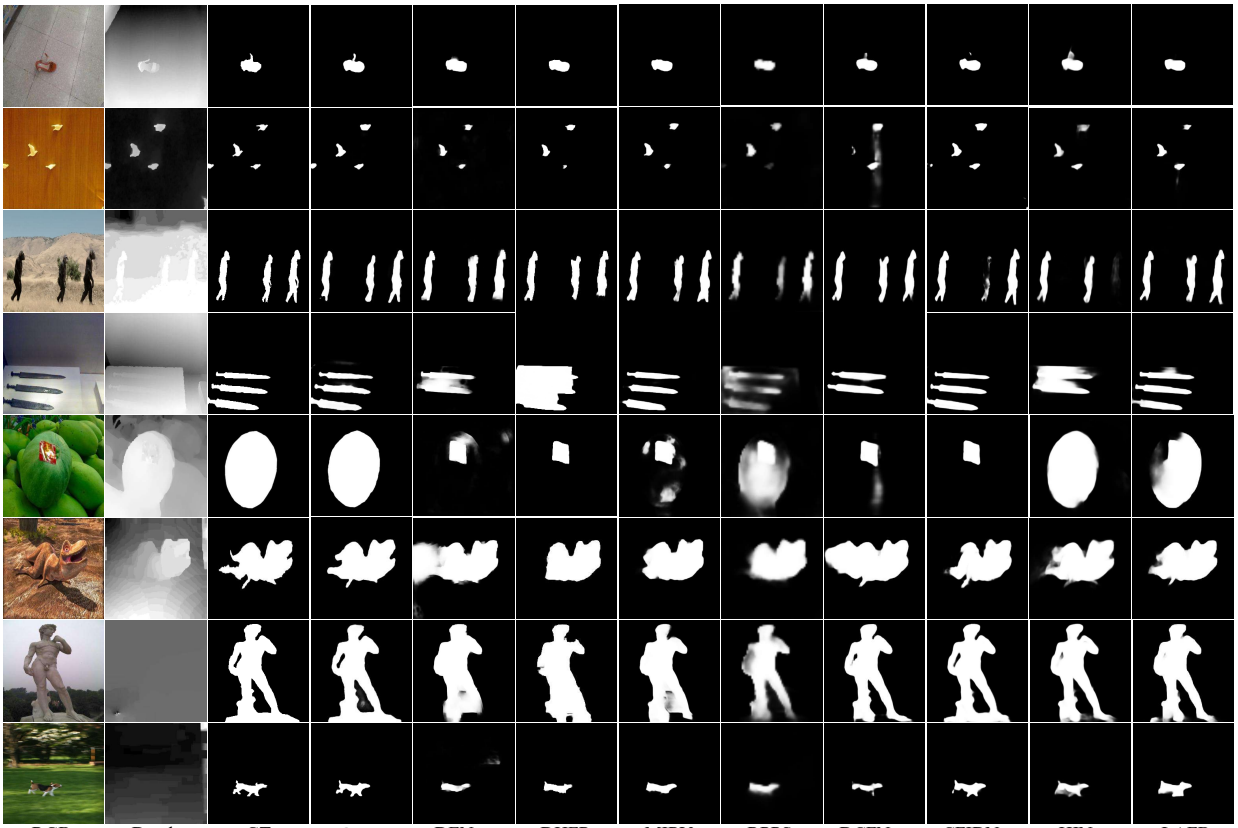

Fig. 8. Visual comparisons of our method and eight state-of-the-art methods in challenging RGB-D SOD scenes.

and metrics on the same benchmark test sets.

1) Quantitative and Model Complexity Comparison with Fully-supervised Methods: We present the quantitative and model complexity comparison results in Tab. I and Tab. II. We first compare our method with fully-supervised RGB-D SOD methods to show the performance gap between weak and dense supervision. Although our method does not rely on dense ground truths, it achieves competitive performance compared to some fully-supervised methods on several datasets and metrics. This indicates that the proposed SAM-PAG and *S^2Diff* can effectively reduce the supervision gap. In terms of model complexity, our model requires higher FLOPs, mainly because the diffusion-based detector performs iterative denoising during inference. This also motivates us to further reduce computational costs in the future and strike a balance between performance and efficiency.

2) Quantitative and Model Complexity Comparison with Weakly-supervised Methods: Compared with weakly-supervised RGB-D SOD methods, our method achieves superior performance on all datasets, outperforming all weakly-supervised methods on five datasets, *i.e.*, STERE, NLPR, DUT, LFSD, and SSD, in all metrics. Regarding the model complexity, the parameter count and full inference cost of our *S^2Diff* are 169.0M and 559.0G FLOPs, respectively, with a 352×352 input. The FLOPs are calculated over the complete 10 step denoising process during inference. Compared with existing weakly-supervised methods, our model has relatively high parameter numbers and computational cost, which need to be further optimized in future work.

TABLE III
PRACTICAL EFFICIENCY ANALYSIS OF THE PROPOSED METHOD.

Procedure	Time Cost	GPU Memory
SAM-PAG		
EAG	0.166 ± 0.001 s/image	–
ITB	3.484 ± 0.007 s/image	–
PEB	1.310 ± 0.002 s/image	–
CPMG	3.493 ± 0.026 ms/image	–
DenseCRF	1.075 ± 0.005 s/image	–
Total	4.797 ± 0.009 s/image	5742 MB
S^2Diff		
Training	7.458 ± 0.006 h	20572 MB
Inference	0.891 ± 0.011 s/image	986 MB

3) Visual Comparison: We perform a visual comparison of our method with eight RGB-D SOD methods, including four weakly-supervised methods (DENet, DHFR, MIRV, and RPPS) and four fully-supervised methods (DCFNet, CFIDNet, HINet, and LAFB). We present the visual results in Fig. 8, including various challenging scenarios, such as small objects (1st and 2nd rows), multiple objects (3rd and 4th rows), low contrast (5th and 6th rows), and low-quality depth map (7th and 8th rows). It can demonstrate that our method performs better in these complex scenes, illustrating the effectiveness and robustness of our method.

4) Practical Efficiency Analysis of the Proposed Method: As shown in Tab. III, we provide the practical efficiency analysis of both SAM-PAG and *S^2Diff*. We run our method three times,

TABLE IV
ABLATION STUDIES ON THE CONTRIBUTION OF SAM-PAG AND S^2DIFF. **BOLD** INDICATES THE BEST RESULT IN EACH COLUMN.

Variant	SSD				LFSD			
	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$
Base	0.080	0.862	0.792	0.744	0.097	0.851	0.810	0.773
Basic SAM	0.064	0.903	0.822	0.771	0.082	0.881	0.838	0.810
Non-Diff	0.073	0.880	0.807	0.760	0.092	0.859	0.817	0.785
Ours	**0.047**	**0.917**	**0.856**	**0.829**	**0.064**	**0.902**	**0.869**	**0.849**

TABLE V
ABLATION STUDIES ON THE EFFECTIVENESS OF EACH COMPONENT OF SAM-PAG. **BOLD** INDICATES THE BEST RESULT IN EACH COLUMN.

Variant	SSD				LFSD			
	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$
w/o ITB	0.061	0.899	0.840	0.797	0.077	0.876	0.842	0.815
w/o PEB	0.059	0.904	0.837	0.794	0.071	0.891	0.855	0.836
w/o CPMG	0.057	0.903	0.849	0.809	0.080	0.882	0.845	0.820
Ours	**0.047**	**0.917**	**0.856**	**0.829**	**0.064**	**0.902**	**0.869**	**0.849**

TABLE VI
ABLATION STUDIES ON THE SUPERPIXEL NUMBER IN THE PROMPT EXTENSION BRANCH.

Number	SSD				LFSD			
	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$
50	0.068	0.889	0.818	0.772	0.073	0.889	0.846	0.823
60	0.060	0.899	0.839	0.804	0.075	0.888	0.853	0.834
65	0.053	0.920	0.851	0.813	0.068	0.897	0.859	0.838
70 (Ours)	**0.047**	**0.917**	**0.856**	**0.829**	**0.064**	**0.902**	**0.869**	**0.849**
75	0.054	0.921	0.861	0.828	0.069	0.895	0.856	0.831
80	0.064	0.897	0.832	0.791	0.077	0.884	0.850	0.826
90	0.062	0.901	0.834	0.794	0.079	0.882	0.844	0.819

TABLE VII
ABLATION STUDIES ON THE SAMPLED POINT NUMBER ON $\boldsymbol{m}_E$.

Number	SSD				LFSD			
	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$
5	0.056	0.907	0.847	0.802	0.075	0.891	0.855	0.832
10 (Ours)	**0.047**	**0.917**	**0.856**	**0.829**	**0.064**	**0.902**	**0.869**	**0.849**
15	0.049	0.911	**0.856**	0.824	0.069	0.890	0.858	0.837

and report the mean and standard deviation. For SAM-PAG, the reported pseudo annotation generation time covers the complete offline pipeline, including the extended annotation generation (EAG), image transformation branch (ITB), the prompt extension branch (PEB), consistency-based pseudo mask generation (CPMG), and DenseCRF post-processing. Note that the reported "Total" excludes extended annotation generation and DenseCRF post-processing. Since SAM-PAG is only used to generate pseudo annotations before training, it does not introduce additional computational cost during the training and inference stage of *S^2Diff*. For *S^2Diff*, the training time is measured over the 150 epochs, excluding the validation time. The inference latency is measured over the complete 10 step denoising process. GPU memory usage is reported as the peak allocated memory during the corresponding procedure.

C. Ablation Studies

Here, we design some ablation studies to thoroughly demonstrate the effectiveness of each component of our method. We conduct these experiments on the SSD and LFSD datasets.

1) Contribution of SAM-PAG and S^2Diff. To evaluate the contribution of SAM-PAG and S^2Diff, we provide three combinations of pseudo labels and saliency detectors, including training a non-diffusion saliency detector with basic SAM-generated pseudo labels (*i.e.*, *Base*), training the proposed S^2Diff detector with basic SAM-generated pseudo labels (*i.e.*, *Basic SAM*), and training a non-diffusion saliency detector with the proposed SAM-PAG pseudo labels (*i.e.*, *Non-Diff*). Specifically, for *Basic SAM*, we directly sample point prompts from the scribble annotations and feed them into SAM to generate pseudo labels. For the non-diffusion detector, we keep the feature extractor from the conditional feature generation network of S^2Diff to extract RGB-D features, and adopt the decoder of the denoising network for saliency prediction.

As shown in Tab. IV, using SAM-PAG pseudo labels improves the performance under the same detector, indicating that the proposed SAM-PAG improves supervision quality. In addition, replacing the non-diffusion detector with S^2Diff under the same supervision further promotes the performance, showing the effectiveness of the proposed S^2Diff. The full setting (*i.e.*, Ours) achieves the best performance, demonstrating that SAM-PAG provides reliable supervision, S^2Diff achieves robust detection, and they provide complementary gains.

2) Effectiveness of Each Component of SAM-PAG. We evaluate the effectiveness of the image transformation branch (ITB), the prompt extension branch (PEB), and the consistency-based pseudo mask generation (CPMG) method in SAM-PAG. As illustrated in Tab. V, removing ITB or PEB (*i.e.*, *w/o ITB* and *w/o PEB*) leads to a drop in final performance, proving the role of two branches in improving the quality of pseudo labels. As for the CPMG method, we remove it (*i.e.*, *w/o CPMG*) and make the average fusion of all masks generated by two branches. It is clear that selecting and weighting masks based on consistency contribute to reliable pseudo labels.

3) Superpixel Number in the Prompt Extension Branch. The superpixel number denotes our annotation extension granularity. An excessive number leads to insufficient expansion, while a small number leads to blurred object and background boundaries, and then affects the labeling of sampled points. Therefore, we demonstrate the impact of superpixel number on performance in Tab. VI. The results show that the best performance is around 70, while that of other superpixel number settings is relatively suboptimal. The performance varies smoothly across neighboring settings (*e.g.*, 65 and 75), indicating that the optimal superpixel number lies within a reasonable range around 70. So we set the superpixel number to 70 in our SAM-PAG.

4) Sampled Point Number on $\boldsymbol{m}_E$. In SAM-PAG, we sample points from scribble and extended annotation $\boldsymbol{m}_E$ as prompts of SAM. Generally, more prompts favor segmentation accuracy and ambiguity reduction. We show the experimental results on the sampled point number on $\boldsymbol{m}_E$ in Tab. VII. Contrary to the intuitive understanding, the more points sampled, the better. The best performance occurs when the sampled point number is 10. We believe this is because $\boldsymbol{m}_E$ is derived from superpixel propagation and is inevitably subject to errors. Therefore, if an excessive number of points are sampled on

TABLE VIII
ABLATION STUDIES ON EFFECTIVENESS OF EACH MODULE IN S^2DIFF.

Base	CCGM	CIM	SSD $M\downarrow$	SSD $E_m\uparrow$	SSD $S_m\uparrow$	SSD $F_m^\omega\uparrow$	LFSD $M\downarrow$	LFSD $E_m\uparrow$	LFSD $S_m\uparrow$	LFSD $F_m^\omega\uparrow$
✓			0.067	0.894	0.835	0.795	0.085	0.867	0.837	0.810
✓		✓	0.061	0.896	0.840	0.792	0.079	0.883	0.842	0.815
✓	✓		0.056	0.912	0.842	0.805	0.083	0.871	0.840	0.812
✓	✓	✓	**0.047**	**0.917**	**0.856**	**0.829**	**0.064**	**0.902**	**0.869**	**0.849**

TABLE IX
ABLATION STUDIES ON EFFECTIVENESS OF EACH COMPONENT IN CCGM.

FFT	Intra-SSM	Inter-SSM	SSD $M\downarrow$	SSD $E_m\uparrow$	SSD $S_m\uparrow$	SSD $F_m^\omega\uparrow$	LFSD $M\downarrow$	LFSD $E_m\uparrow$	LFSD $S_m\uparrow$	LFSD $F_m^\omega\uparrow$
			0.061	0.896	0.840	0.792	0.079	0.883	0.842	0.815
✓			0.052	0.899	0.844	0.800	0.078	0.877	0.843	0.821
✓	✓		0.051	0.910	0.851	0.809	0.075	0.884	0.848	0.821
✓		✓	0.053	**0.917**	**0.856**	0.825	0.074	0.885	0.852	0.830
✓	✓	✓	**0.047**	**0.917**	**0.856**	**0.829**	**0.064**	**0.902**	**0.869**	**0.849**

it (*e.g.*, 15 points), this will introduce erroneous information that is detrimental to SAM segmentation. Conversely, an insufficient number of sampled points (*e.g.*, 5 points) results in an inadequacy of information provided by prompts.

5) Effectiveness of Each Module in S^2Diff. To verify the effectiveness of each module in our S^2Diff, we provide three variants, including Base, Base+CCGM, Base+CIM. We replace CCGM and CIM with an element-wise summation operation simultaneously to build "Base". As demonstrated in Tab. VIII, the removal of CCGM and CIM leads to a significant drop in the performance of Base. Base+CCGM and Base+CIM are better than baseline, and the joint introduction of CCGM and CIM (*i.e.*, the complete S^2Diff) develops the performance most, illustrating the necessity of two modules.

6) Effectiveness of Each Component in CCGM. CCGM aims to fuse cross-modal features and provide conditional features to the denoising network, which is composed of frequency integration, intra-SSM, and inter-SSM. To assess the impact of each component in CCGM, we provide four variants in Tab. IX. We can observe that FFT is better than the first row (removing the entire CCGM), demonstrating the role of frequency exchange in cross-modal fusion. With the addition of Intra-SSM and Inter-SSM, the overall performance develops, proving that each component avails the global improvement of the conditional feature. And when those components are used jointly, the performance achieves the best.

7) Effectiveness of Each Component in CIM. CIM injects conditional context into noise features, which consists of a channel-wise attention (CA) and an SSM. We research the components of CIM with three variants, including removing CA (*i.e.*, *w/o CA*), removing SSM (*i.e.*, *w/o SSM*), and performing CA on the conditional feature (*i.e.*, *CondFea w/ CA*), and we present the results in Tab. X. Comparing *w/o CA* and *w/o* SSM with *w/o CIM*, we can find that the performance increases due to the effectiveness of CA and SSM. CA modulates the features with the channel attention map of noise features. It can be observed that *CondFea w/ CA* is worse than our modulation on noise features. We analyze that the noise feature carries more object information than the conditional feature, and the information becomes increasingly clear with the raise in time steps. Therefore, our modulation of noise features is more conducive to denoising.

TABLE X
ABLATION STUDIES ON EFFECTIVENESS OF EACH COMPONENT IN CIM.

Variant	SSD $M\downarrow$	SSD $E_m\uparrow$	SSD $S_m\uparrow$	SSD $F_m^\omega\uparrow$	LFSD $M\downarrow$	LFSD $E_m\uparrow$	LFSD $S_m\uparrow$	LFSD $F_m^\omega\uparrow$
w/o CIM	0.056	0.912	0.842	0.805	0.083	0.871	0.840	0.812
w/o CA	0.052	0.908	0.853	0.809	0.072	0.886	0.849	0.824
w/o SSM	0.050	0.913	0.852	0.807	0.078	0.878	0.841	0.816
CondFea w/ CA	0.053	0.911	0.843	0.806	0.072	0.887	0.854	0.835
Ours	**0.047**	**0.917**	**0.856**	**0.829**	**0.064**	**0.902**	**0.869**	**0.849**

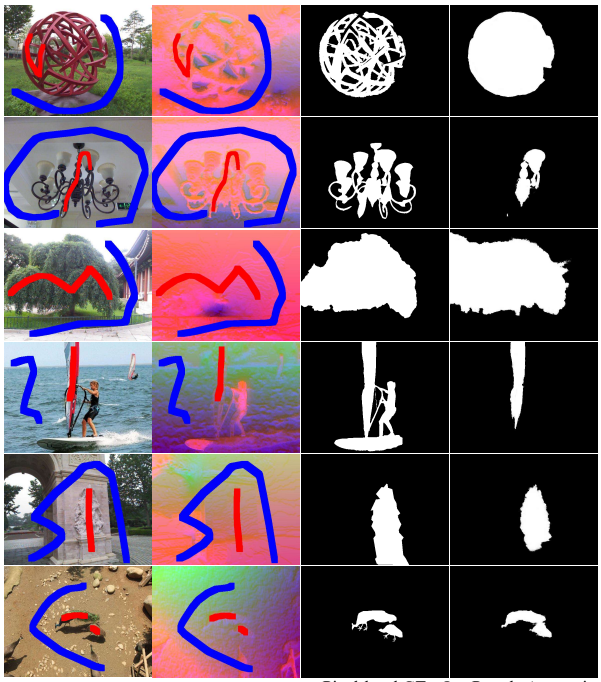

Fig. 9. Failure cases of our SAM-PAG pseudo annotations.

D. Failure Cases and Limitation of SAM-PAG

In SAM-PAG, we explicitly adopt SAM as a frozen external foundation model to introduce generic segmentation priors into weakly-supervised RGB-D SOD. This setting is consistent with the goal of weak supervision, which aims to reduce the cost of task-specific manual annotations, especially dense pixel-level annotations. Specifically, we do not use pixel-level ground truths from RGB-D SOD datasets, nor do we adapt or fine-tune SAM on the downstream RGB-D SOD task. Instead, SAM serves as a frozen general segmentation prior to facilitating the conversion from sparse scribble annotations to dense pseudo annotations. Therefore, our method remains weakly-supervised with respect to the target RGB-D SOD task, while generating high-quality pseudo labels by leveraging the general priors offered by SAM.

However, despite the overall effectiveness of our SAM-PAG, failures still occur in several challenging scenarios, as illustrated in Fig. 9. The 1st and 2nd rows show cases with complex structures, including intricate internal regions and slender branches. In these cases, SAM-PAG may lose

interleaved details and fine-scale branches, since SAM is less sensitive to complex topological structures and sparse scribbles can not provide sufficiently detailed prompts. The 3rd row shows that low-quality HHA map may introduce inaccurate geometric information, leading to unreliable pseudo annotations. The 4th row presents a saliency-incomplete case caused by insufficient scribble annotations, where SAM only segments the annotated sail region, but fails to include the surfer and the windsurfing board. The 5th row shows a low contrast case, where ambiguous object boundaries make it difficult to accurately separate the sculpture from the wall. The last row illustrates a shadow-induced failure case, where shadows weaken the contrast between foreground and background, and disturb the appearance consistency of the object.

These cases suggest that the segmentation priors of SAM are not always aligned with the objective of SOD. SAM-PAG may still be affected by complex structures, imperfect HHA maps, insufficient scribble guidance, low foreground-background contrast, and shadows. Therefore, robust pseudo annotation generation under complex scenes remains an important direction for future research.

VI. Conclusion

In this paper, we propose a novel scribble-supervised RGB-D SOD method, composed of SAM-PAG and S^2Diff. SAM-PAG aims to generate pixel-level pseudo annotations with the strong segmentation ability of the visual foundation model SAM through the dual-branch structure and consistency-based pseudo mask generation method. S^2Diff is a state space interaction-based conditional diffusion model, generating accurate saliency maps by iterative refinement of the noisy mask with the supervision of scribble and dense pseudo annotations. CCGM and CIM are the cores of S^2Diff. CCGM interweaves cross-modal features through frequency exchange and implicit-explicit interaction to generate global conditional information. CIM models the noise features with the conditional context from CCGM to guide the denoising process. Experiments on seven datasets illustrate the superiority of our method.

References

[1] G. Li, Z. Liu, and H. Ling, "ICNet: Information conversion network for RGB-D based salient object detection," *IEEE Trans. Image Process.*, vol. 29, pp. 4873–4884, Mar. 2020.

[2] D.-P. Fan, Y. Zhai, A. Borji, J. Yang, and L. Shao, "BBS-Net: RGB-D salient object detection with a bifurcated backbone strategy network," in *Proc. ECCV*, Aug. 2020, pp. 275–292.

[3] G. Li, Z. Liu, L. Ye, Y. Wang, and H. Ling, "Cross-modal weighting network for RGB-D salient object detection," in *Proc. ECCV*, Aug. 2020, pp. 665–681.

[4] G. Li, Z. Liu, M. Chen, Z. Bai, W. Lin, and H. Ling, "Hierarchical alternate interaction network for RGB-D salient object detection," *IEEE Trans. Image Process.*, vol. 30, pp. 3528–3542, Mar. 2021.

[5] Z. Zeng, H. Liu, F. Chen, and X. Tan, "AirSOD: A lightweight network for RGB-D salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 3, pp. 1656–1669, 2024.

[6] S. Shi, G. Li, R. Cong, S. Xiao, and W. Lin, "Diffusion-driven RGB-D salient object detection with temporal modulation," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1–12, 2026.

[7] J. Chen, G. Li, Z. Zhang, L. Chang, and D. Zeng, "STENet: Superpixel token enhancing network for RGB-D salient object detection," *IEEE Trans. Multimedia*, pp. 1–12, 2026.

[8] Y. Ding, W. Chen, G. Zhang, Z. Feng, and X. Li, "Cross-modal weakly supervised RGB-D salient object detection with a focus on filamentary structures," *Sensors*, vol. 25, no. 10, May 2025.

[9] Y. Zhang, Z. Zhang, T. Liu, and J. Kong, "Category-aware saliency enhance learning based on CLIP for weakly supervised salient object detection," *Neural Process. Lett.*, vol. 56, no. 2, p. 49, Feb. 2024.

[10] Y. Xu, X. Yu, J. Zhang, L. Zhu, and D. Wang, "Weakly supervised RGB-D salient object detection with prediction consistency training and active scribble boosting," *IEEE Trans. Image Process.*, vol. 31, pp. 2148–2161, 2022.

[11] Z. Liu, M. Hayat, H. Yang, D. Peng, and Y. Lei, "Deep hypersphere feature regularization for weakly supervised RGB-D salient object detection," *IEEE Trans. Image Process.*, vol. 32, pp. 5423–5437, 2023.

[12] A. Li, Y. Mao, J. Zhang, and Y. Dai, "Mutual information regularization for weakly-supervised RGB-D salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 1, pp. 397–410, 2024.

[13] L. Li, J. Han, N. Liu, S. Khan, H. Cholakkal, R. M. Anwer, and F. S. Khan, "Robust perception and precise segmentation for scribble-supervised RGB-D saliency detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 1, pp. 479–496, Jan. 2024.

[14] Y. Wang, L. Zhang, P. Zhang, Y. Zhuge, J. Wu, H. Yu, and H. Lu, "Learning local-global representation for scribble-based RGB-D salient object detection via transformer," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 11, pp. 11 592–11 604, Jul. 2024.

[15] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," in *Proc. IEEE ICCV*, Oct. 2023, pp. 3992–4003.

[16] G. Li, Z. Bai, and Z. Liu, "Texture-semantic collaboration network for ORSI salient object detection," *IEEE Trans. Circuits Syst. II-Express Briefs*, vol. 71, no. 4, pp. 2464–2468, Apr. 2024.

[17] G. Li, S. Shi, Y. Wu, W. Lin, and Z. Bai, "Lightweight ORSI salient object detection via frequency and mutual assistance attention," *IEEE Trans. Geosci. Remote Sens.*, vol. 64, pp. 1–12, Apr. 2026.

[18] Z. Liu, X. Huang, G. Zhang, X. Fang, L. Wang, and B. Tang, "Scribble-supervised RGB-T salient object detection," in *Proc. IEEE ICME*, Mar. 2023, pp. 2369–2374.

[19] J. Ma, Y. He, F. Li, L. Han, C. You, and B. Wang, "Segment anything in medical images," *Nat. Commun.*, vol. 15, no. 1, p. 654, Jan. 2024.

[20] J. Wu, Z. Wang, M. Hong, W. Ji, H. Fu, Y. Xu, M. Xu, and Y. Jin, "Medical SAM adapter: Adapting segment anything model for medical image segmentation," *Med. Image Anal.*, vol. 102, p. 103547, 2025.

[21] Z. Liu, S. Deng, X. Wang, L. Wang, X. Fang, and B. Tang, "SSFam: Scribble supervised salient object detection family," *IEEE Trans. Multimedia*, pp. 1988–2000, Mar. 2025.

[22] L. Tang, P.-T. Jiang, H. Xiao, and B. Li, "Towards training-free open-world segmentation via image prompt foundation models," *Int. J. Comput. Vis.*, vol. 133, pp. 1–15, Jul. 2024.

[23] J. Hu, J. Lin, S. Gong, and W. Cai, "Relax image-specific prompt requirement in SAM: a single generic prompt for segmenting camouflaged objects," in *Proc. AAAI*, 2024, p. 12511–12518.

[24] Y.-J. Yuan, Y. Zhang, S. Zhang, and H. Wang, "SaliencyCLIP-SAM: Bridging text and image towards text-driven salient object detection," in *Proc. ICIG*, 2025, pp. 29–41.

[25] X. Hu, F. Sun, J. Liu, F. Xu, and X. Zhang, "ST-SAM: SAM-driven self-training framework for semi-supervised camouflaged object detection," in *Proc. ACM MM*, Oct. 2025, p. 8194–8203.

[26] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proc. ICML*, vol. 139, Jul. 2021, pp. 8748–8763.

[27] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. NeurIPS*, vol. 33, Dec. 2020, pp. 6840–6851.

[28] Y. Qing, S. Liu, H. Wang, and Y. Wang, "DiffUIE: Learning latent global priors in diffusion models for underwater image enhancement," *IEEE Trans. Multimedia*, vol. 27, pp. 2516–2529, Dec. 2025.

[29] J. Yang, B. Zhong, Q. Liang, Z. Mo, S. Zhang, and S. Song, "Uncertainty-guided diffusion model for camouflaged object detection," *IEEE Trans. Multimedia*, vol. 27, pp. 4656–4669, Jan. 2025.

[30] K. Sun, Z. Chen, X. Lin, X. Sun, H. Liu, and R. Ji, "Conditional diffusion models for camouflaged and salient object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 4, pp. 2833–2848, 2025.

[31] W. Wu, Y. Zhao, M. Z. Shou, H. Zhou, and C. Shen, "DiffuMask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models," in *Proc. IEEE ICCV*, Oct. 2023, pp. 1206–1217.

[32] J. Han, J. Sun, F. Wang, F. Sun, and H. Li, "ORSIDiff: Diffusion model for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–15, Jun. 2025.

[33] Y. Hou and T. Li, "DiffORSINet: Salient object detection in optical remote sensing images via conditional diffusion model," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, pp. 1–13, Dec. 2025.

[34] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning rich features from RGB-D images for object detection and segmentation," in *Proc. ECCV*, Sep. 2014, pp. 345–360.

[35] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and Sabine.Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–82, Nov. 2012.

[36] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Proc. NeurIPS*, 2011, pp. 109–117.

[37] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "PVT v2: Improved baselines with pyramid vision transformer," *Comput. Vis. Media*, vol. 8, pp. 415–424, 2022.

[38] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," *arXiv preprint arXiv:2312.00752*, 2023.

[39] Y. Liu, Y. Tian, Y. Zhao, H. Yu, L. Xie, Y. Wang, Q. Ye, J. Jiao, and Y. Liu, "VMamba: Visual state space model," in *Proc. NeurIPS*, vol. 37, 2024, pp. 103 031–103 063.

[40] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. ECCV*, Sept. 2018, pp. 3–19.

[41] Y. Niu, Y. Geng, X. Li, and F. Liu, "Leveraging stereopsis for saliency analysis," in *Proc. IEEE CVPR*, 2012, pp. 454–461.

[42] H. Peng, B. Li, W. Xiong, W. Hu, and R. Ji, "RGBD salient object detection: A benchmark and algorithms," in *Proc. ECCV*, 2014, pp. 92–109.

[43] Y. Piao, W. Ji, J. Li, M. Zhang, and H. Lu, "Depth-induced multi-scale recurrent attention network for saliency detection," in *Proc. IEEE ICCV*, 2019, pp. 7254–7263.

[44] N. Li, J. Ye, Y. Ji, H. Ling, and J. Yu, "Saliency detection on light field," in *Proc. IEEE CVPR*, 2014, pp. 2806–2813.

[45] H. Wen, C. Yan, X. Zhou, R. Cong, Y. Sun, B. Zheng, J. Zhang, Y. Bao, and G. Ding, "Dynamic selective network for RGB-D salient object detection," *IEEE Trans. Image Process.*, vol. 30, pp. 9179–9192, 2021.

[46] W. Ji, J. Li, S. Yu, M. Zhang, Y. Piao, S. Yao, Q. Bi, K. Ma, Y. Zheng, H. Lu, and L. Cheng, "Calibrated RGB-D salient object detection," in *Proc. IEEE CVPR*, 2021, pp. 9471–9481.

[47] R. Cong, Q. Lin, C. Zhang, C. Li, X. Cao, Q. Huang, and Y. Zhao, "CIR-Net: Cross-modality interaction and refinement for RGB-D salient object detection," *IEEE Trans. Image Process.*, vol. 31, pp. 6800–6815, 2022.

[48] T. Chen, X. Hu, J. Xiao, G. Zhang, and S. Wang, "CFIDNet: Cascaded feature interaction decoder for RGB-D salient object detection," *Neural. Comput. Appl.*, vol. 34, no. 10, pp. 7547–7563, 2022.

[49] Z. Liu, Y. Tan, Q. He, and Y. Xiao, "SwinNet: Swin transformer drives edge-aware RGB-D and RGB-T salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 7, pp. 4486–4497, 2022.

[50] X. Cheng, X. Zheng, J. Pei, H. Tang, Z. Lyu, and C. Chen, "Depth-induced gap-reducing network for RGB-D salient object detection: An interaction, guidance and refinement approach," *IEEE Trans. Multimedia*, vol. 25, pp. 4253–4266, 2023.

[51] M. Zhang, S. Yao, B. Hu, Y. Piao, and W. Ji, "C^2DFNet: Criss-cross dynamic filter network for RGB-D salient object detection," *IEEE Trans. Multimedia*, vol. 25, pp. 5142–5154, 2023.

[52] H. Bi, R. Wu, Z. Liu, H. Zhu, C. Zhang, and T.-Z. Xiang, "Cross-modal hierarchical interaction network for RGB-D salient object detection," *Pattern Recognit.*, vol. 136, p. 109194, 2023.

[53] Y. Pang, X. Zhao, L. Zhang, and H. Lu, "CAVER: Cross-modal view-mixed transformer for bi-modal salient object detection," *IEEE Trans. Image Process.*, vol. 32, pp. 892–904, 2023.

[54] R. Cong, H. Liu, C. Zhang, W. Zhang, F. Zheng, R. Song, and S. Kwong, "Point-aware interaction and cnn-induced refinement network for RGB-D salient object detection," in *Proc. ACM MM*, Oct. 2023, pp. 406–416.

[55] F. Sun, P. Ren, B. Yin, F. Wang, and H. Li, "CATNet: A cascaded and aggregated transformer network for RGB-D salient object detection," *IEEE Trans. Multimedia*, vol. 26, pp. 2249–2262, 2024.

[56] Q. Chen, Z. Zhang, Y. Lu, K. Fu, and Q. Zhao, "3-D convolutional neural networks for RGB-D salient object detection and beyond," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 3, pp. 4309–4323, 2024.

[57] K. Wang, Z. Tu, C. Li, C. Zhang, and B. Luo, "Learning adaptive fusion bank for multi-modal salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 8, pp. 7344–7358, 2024.

[58] X. Hu, F. Sun, J. Sun, F. Wang, and H. Li, "Cross-modal fusion and progressive decoding network for RGB-D salient object detection," *Int. J. Comput. Vis.*, vol. 132, pp. 3067–3085, 2024.

[59] J. Zhang, R. Zhang, L. Xu, X. Lu, Y. Yu, M. Xu, and H. Zhao, "FasterSal: Robust and real-time single-stream architecture for RGB-D salient object detection," *IEEE Trans. Multimedia*, vol. 27, pp. 2477–2488, 2025.

[60] G. Chen, Q. Wang, B. Dong, R. Ma, N. Liu, H. Fu, and Y. Xia, "EM-Trans: Edge-aware multimodal transformer for RGB-D salient object detection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 2, pp. 3175–3188, 2025.

[61] H. Gao, F. Wang, M. Wang, F. Sun, and H. Li, "Highly efficient RGB-D salient object detection with adaptive fusion and attention regulation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 35, no. 4, pp. 3104–3118, 2025.

[62] 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 Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 2075–2089, 2020.

[63] C. Zhu and G. Li, "A three-pathway psychobiological framework of salient object detection using stereoscopic technology," in *Proc. IEEE ICCVW*, 2017, pp. 3008–3014.

[64] R. Ju, L. Ge, W. Geng, T. Ren, and G. Wu, "Depth saliency based on anisotropic center-surround difference," in *Proc. IEEE ICIP*, 2014, pp. 1115–1119.

[65] M. Tang, A. Djelouah, F. Perazzi, Y. Boykov, and C. Schroers, "Normalized cut loss for weakly-supervised cnn segmentation," in *Proc. IEEE CVPR*, Jun. 2018, pp. 1818–1827.

[66] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE CVPR*, 2009, pp. 1597–1604.

[67] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, and A. Borji, "Enhanced-alignment measure for binary foreground map evaluation," in *Proc. IJCAI*, Jul. 2018, pp. 698–704.

[68] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *Proc. IEEE ICCV*, 2017, pp. 4548–4557.

[69] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps?" in *Proc. IEEE CVPR*, 2014, pp. 248–255.

[70] M. Zhong, J. Sun, P. Ren, F. Wang, and F. Sun, "MAGNet: Multi-scale awareness and global fusion network for RGB-D salient object detection," *Knowl. Based Syst.*, p. 112126, 2024.