Diffusion-driven RGB-D Salient Object Detection with Temporal Modulation

Shixiang Shi, Gongyang Li, *Member, IEEE*, Runmin Cong, *Senior Member, IEEE*, Shunxin Xiao, and Weisi Lin, *Fellow, IEEE*

***Abstract*—Existing RGB-D Salient Object Detection (SOD) methods are primarily built on the end-to-end prediction paradigm. Although these methods have achieved remarkable progress, they still struggle to generate accurate predictions in some complex scenes due to their lack of error correction capability. In this paper, we explore the use of conditional diffusion architectures for RGB-D SOD, producing saliency maps in a step-by-step generation paradigm. Accordingly, we propose *DiffRGBD*, a novel diffusion-driven framework with temporal modulation. The core of DiffRGBD is using time steps to control the conditional information injected into the denoising network in a two-stage temporal modulation manner. Specifically, our DiffRGBD comprises a feature extractor, a conditional generator, two temporal modulators, and a denoising network. First, the SAM2 encoder with adapters is adopted to extract hierarchical cross-modal features. Then, the Mutual-Differential Attention Module is responsible for generating the conditional information via effective cross-modal fusion. Notably, the conditional information continuously achieves channel modulation and spatial modulation in the Temporal Channel Enhancement Module and the Temporal Spatial Refinement Module (*i.e.*, two temporal modulators), resulting in comprehensive conditional information. Finally, conditional information is injected into the denoising network to guide the production of saliency maps. As the time step increases, our DiffRGBD can gradually correct errors and generate accurate saliency maps. Extensive experiments on seven public RGB-D SOD benchmarks demonstrate that our proposed DiffRGBD achieves superior performance over state-of-the-art methods. The code and results of our method are available at https://github.com/Shixiang02/DiffRGBD.**

***Index Terms*—RGB-D salient object detection, conditional diffusion model, temporal modulation, cross-modal fusion.**

I. Introduction

SALIENT Object Detection (SOD) aims to identify and segment the most visually distinctive objects in a scene [1]–[5], serving as a fundamental component in numerous vision applications such as object segmentation [6], image editing [7], and video analysis [8], [9]. Although RGB-based SOD methods have made remarkable progress, they often perform poorly in complex scenes where relying solely on RGB appearance information is insufficient. To alleviate these issues, RGB-D SOD has attracted increasing attention by introducing depth maps as an auxiliary modality to enhance the geometric structure and improve the separability of the foreground from background [10]–[14].

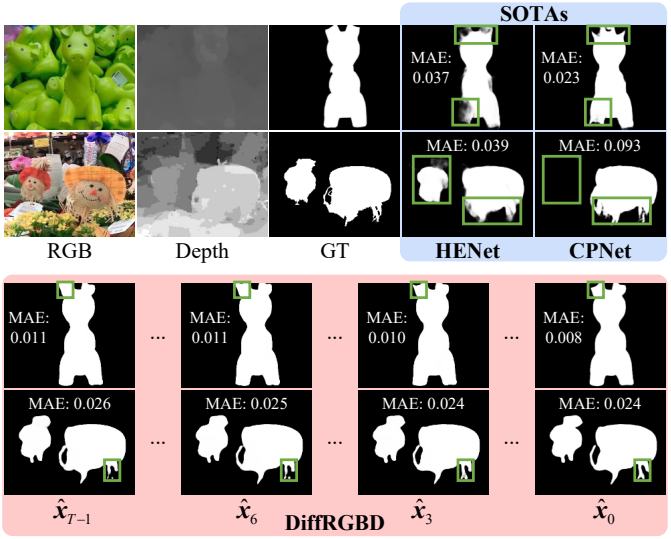

Fig. 1. Visual comparison of two paradigms. The first one is the existing end-to-end prediction paradigm (blue color), such as SOTA HENet [15] and CPNet [12]. This paradigm generates saliency maps in a single forward pass, which easily leads to uncorrected prediction errors (green boxes), resulting in high MAE. The other is our step-by-step generation paradigm (pink color). As the time step changes, our DiffRGBD can gradually correct prediction errors (green boxes), significantly reducing MAE.

In recent years, substantial progress has been made in RGB-D SOD [16], [17], driven by the development of Convolutional Neural Networks (CNNs) [18], [19] and Transformers [20]–[22]. Existing RGB-D SOD methods have demonstrated strong capabilities in learning powerful multi-modal representations, achieving remarkable performance. They typically follow the end-to-end prediction paradigm, which means that after model training, they directly infer the saliency values of inputs, *i.e.*, generating saliency maps in a single forward pass. This paradigm results in existing methods still struggling to handle some complex scenes, such as low contrast, cluttered backgrounds, and low-quality depth maps. As shown in Fig. 1, the state-of-the-art (SOTA) HENet [15] and CPNet [12] are two representatives of this paradigm. In the first scene with

Shixiang Shi and Gongyang Li are with the School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China, and the Fujian Key Laboratory of Pattern Recognition and Image Understanding, Xiamen University of Technology, Xiamen 361024, China (e-mail: shishixiang@shu.edu.cn; ligongyang@shu.edu.cn).

Runmin Cong is with the School of Control Science and Engineering, Shandong University, Jinan 250061, China (e-mail: rmcong@sdu.edu.cn).

Shunxin Xiao is with the Fujian Key Laboratory of Pattern Recognition and Image Understanding, School of Computer and Information Engineering, Xiamen University of Technology, Xiamen 361024, China (e-mail: xiaoshunxin.tj@gmail.com).

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.

low contrast and low-quality depth maps, HENet and CPNet cannot finely highlight the head and feet of the green toy. In the second one with cluttered backgrounds, HENet misses detecting details of the dolls, while CPNet fails to detect the doll on the left. This is primarily because the end-to-end prediction paradigm generates saliency maps in a single forward pass, without endowing existing methods with the ability to correct erroneous predictions in challenging scenes.

Recently, the field of image generation has undergone a complete transformation. The diffusion models demonstrate excellent generation quality [23], [24]. They can generate high-quality images from noise through the denoising process. Inspired by the diffusion model, we attempt to introduce it into RGB-D SOD, and propose the step-by-step generation paradigm, endowing the RGB-D SOD model with the ability to gradually correct errors and refine saliency maps. We formulate RGB-D SOD as a conditional diffusion process, in which the RGB image and depth map are encoded as conditional information to guide the denoising network to refine saliency maps step by step. In the conditional diffusion process, we believe that the denoising network has different requirements for conditional information at different time steps. Therefore, we propose DiffRGBD, which adopts the two-stage temporal modulation strategy to adjust the conditional information injected into the denoising network over time steps.

In particular, we employ the SAM2 encoder [25] as the feature extractor to extract hierarchical RGB and depth features. Then, we propose a Mutual-Differential Attention Module (MDAM) (*i.e.*, the conditional generator) to jointly model common semantics and modality-specific discrepancies from cross-modal features, generating conditional information. To enable temporally adaptive guidance, we subsequently arrange two temporal modulators, *i.e.*, the Temporal Channel Enhancement Module (TCEM) and the Temporal Spatial Refinement Module (TSRM). Both modulators implement information modulation based on time steps. TCEM performs channel modulation and provides initial structural guidance for the encoder in the denoising network. TSRM performs spatial modulation and provides rich semantic guidance for the decoder in the denoising network. Finally, the modulated conditional information is injected into the denoising network. In this way, our DiffRGBD is guided by the effective conditional information to progressively correct prediction errors across time steps, generating high-quality saliency maps. As shown in Fig. 1, our initial saliency map (*i.e.*, $\hat{\boldsymbol{x}}_{T-1}$) demonstrates high detection accuracy but has a slight flaw. As the time step increases, this flaw is gradually corrected, ultimately resulting in an accurate saliency map (*i.e.*, $\hat{\boldsymbol{x}}_0$).

The main contributions are summarized as follows:

- We propose *DiffRGBD*, a novel diffusion-driven RGB-D SOD framework, which follows the unique step-by-step generation paradigm to overcome the error correction limitations of the conventional end-to-end prediction paradigm. The core of DiffRGBD is the two-stage temporal modulation strategy, which leverages time steps to control the injection of conditional information into the denoising network.
- We propose a conditional generator, *i.e.*, MDAM, to achieve effective and complementary fusion of cross-modal features and capture both common semantics and modality-specific discrepancies among them, generating conditional information for subsequent two-stage temporal modulation.
- We propose two temporal modulators, *i.e.*, TCEM and TSRM, to modulate conditional information from the channel and spatial dimensions based on time steps, achieving effective guidance for the denoising network. TCEM provides structural guidance for the encoder in the denoising network, while TSRM provides rich semantic guidance for the decoder in the denoising network.

The remainder of this paper is organized as follows. In Sec. II, we review the related work. In Sec. III, we present the details of the proposed DiffRGBD. In Sec. IV, we report experiments and ablations on seven benchmarks. In Sec. V, we provide the conclusion.

II. Related Work

A. RGB-D Salient Object Detection

RGB-D SOD has garnered significant attention in recent years due to the complementary nature of RGB and depth information. Traditional RGB-based methods face limitations in complex scenes where appearance information alone is insufficient to discern salient objects [26], [27]. To mitigate these challenges, RGB-D methods leverage depth information to enhance foreground-background separation and capture spatial structures, enabling more robust detection in cluttered and low-contrast environments.

In the past few years, the end-to-end prediction paradigm has become a prominent solution for RGB-D SOD. This paradigm typically aims to fuse RGB and depth features in a single unified network, enabling joint learning of both modalities for more robust SOD. In this paradigm, CNNs have been extensively adopted due to their strong capability in capturing spatial features and their adaptability to multi-stream architectures. For example, Fan *et al.* [28] introduced a bifurcated backbone strategy to decouple multi-level features into teacher and student groups for effective RGB-D fusion. Li *et al.* [29] utilized depth-to-RGB modulation and RGB self-modulation to enhance adjacent scale features and achieve cross-modal modulation. Fu *et al.* [30] introduced a Siamese architecture with joint learning for shared feature extraction, achieving effective cross-modal feature interaction. Zhang *et al.* [31] explored complementary foreground and background information through dual attention mechanisms, achieving superior performance and real-time speed. He *et al.* [32] proposed a unified Mamba-based framework for SOD, *i.e.*, Samba, which redesigns the scanning strategy to preserve the spatial continuity of salient regions and achieves strong performance across multiple SOD tasks, including RGB-D SOD.

More recently, transformers [20], [21] have been introduced into the RGB-D SOD, offering a new perspective for multi-modal modeling in the end-to-end prediction paradigm. By leveraging the self-attention mechanism [20], these models

are capable of capturing long-range dependencies and contextual relationships within and across modalities, which are often difficult to model effectively for CNNs. For example, Liu *et al*. [16] used the Swin Transformer [21] to extract hierarchical features, along with attention-based cross-modal optimization to locate salient objects better. Cong *et al*. [10] proposed a CNN-assisted transformer method, which explores cross-modal interactions with positional constraints and global guidance via a point-aware mechanism and uses a CNN-induced refinement unit to enhance details. Chen *et al*. [33] modeled edge information through dual-band decomposition and explored multi-modal complementarity via cross-attention, achieving more accurate boundary prediction.

Despite the remarkable progress achieved by existing CNN and Transformer-based RGB-D SOD methods, they still suffer from the inherent limitations of the end-to-end prediction paradigm. These methods generate saliency maps in a single forward pass, lacking the ability to correct predictions once they are made. To overcome this limitation, we propose the step-by-step generation paradigm using diffusion models, enabling progressive refinement of saliency maps and error correction with increasing time steps.

B. Diffusion Models and Applications

Diffusion models [23], [24] have drawn increasing attention due to their powerful generative capabilities, particularly their ability to model complex data distributions. Unlike conventional end-to-end models that perform inference in a single forward pass, diffusion-based models for vision tasks generate predictions by gradually denoising random noise over a sequence of time steps. This process allows models to gradually improve predictions, making it particularly suitable for tasks that require step-wise correction and fine-grained generation.

Due to their outstanding performance in generating high-quality outputs, there has been a growing interest in incorporating conditional information into the diffusion process. This adaptive injection of conditions enables the model to generate specific outputs, whether for image enhancement [34], semantic segmentation [35], or other tasks [36]–[38]. For example, Zhang *et al*. [39] proposed a unified generative approach that reformulates multi-modal SOD as a conditional denoising process, leveraging diffusion models to effectively integrate RGB, depth, and thermal inputs. Song *et al*. [40] proposed DiffSOD, a diffusion-based model that formulates SOD as a noise-to-image denoising process guided by appearance and structure conditions via specialized control adapters. Sun *et al*. [41] employed a conditional diffusion model to tackle the camouflaged object detection task, achieving precise predictions of objects that are highly similar to their background. Han *et al*. [42] proposed a diffusion-based method for SOD in optical remote sensing images, leveraging a global-local denoising network and a consistency assessment strategy to improve refinement and reduce overconfident predictions.

Due to the characteristic of the diffusion model that gradually refines the prediction over time steps, we believe that the demand for conditional information varies at different stages of the denoising process. However, existing diffusion models for visual tasks ignore this phenomenon, and do not adjust the condition information injected into the denoising network. Differently, we propose the two-stage temporal modulation strategy to control the injection of conditional information into the denoising network across time steps, enabling adaptive and effective use of conditional cues along the diffusion process.

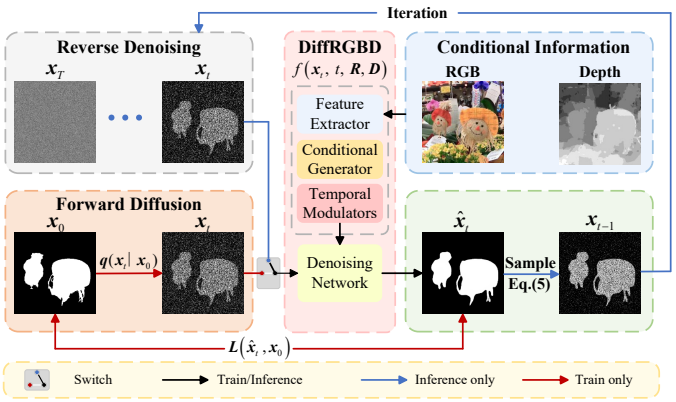

Fig. 2. Diffusion principle of our DiffRGBD, which consists of the diffusion and denoising processes. In the diffusion process, Gaussian noise is progressively added to the ground truth mask $\boldsymbol{x}_0$ to form a noisy mask $\boldsymbol{x}_t$. Then our DiffRGBD, *i.e.*, $f(\boldsymbol{x}_t, t, \boldsymbol{R}, \boldsymbol{D})$, is trained to predict the denoised mask $\hat{\boldsymbol{x}}_t$. In the denoising process, our DiffRGBD iteratively denoises the random noise $\boldsymbol{x}_T$ to generate the final saliency map.

III. Proposed Method

A. Diffusion Principle

Diffusion models are generative frameworks that model data generation as a Markov transformation from noise to structure, involving a forward diffusion and a reverse denoising process.

As illustrated in Fig. 2, in the forward process, the ground truth mask $\boldsymbol{x}_0$ is gradually degraded into a noisy mask $\boldsymbol{x}_t$ by adding Gaussian noise over T time steps, simulating the progressive corruption of the ground truth mask. This forward process can be described as:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}\left(\boldsymbol{x}_t; \sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\boldsymbol{I}\right), \tag{1}$$

where $\boldsymbol{\alpha}_t \in (0,1)$ represents the noise schedule at each time step, and $\boldsymbol{I}$ represents the identity matrix. Owing to the Markov property of the forward process, the marginal distribution of $\boldsymbol{x}_t$ given $\boldsymbol{x}_0$ can be derived as:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_t; \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I}\right), \tag{2}$$

where $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$.

In the reverse denoising process, a neural network f is employed to iteratively refine the random Gaussian noise $\boldsymbol{x}_T$ over time steps to restore the ground truth mask $\boldsymbol{x}_0$, where at each time step $\hat{\boldsymbol{x}}_t$ is first predicted and then used to sample $\boldsymbol{x}_{t-1}$. Each step is described as:

$$p(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) := \mathcal{N}(\boldsymbol{x}_{t-1}; \mu(\boldsymbol{x}_t, \boldsymbol{t}), \sigma_t^2 \boldsymbol{I}), \tag{3}$$

$$\sigma_t^2 = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\left(1-\alpha_t\right), \tag{4}$$

$$\mu(\boldsymbol{x}_t, t) = \frac{\sqrt{\alpha_t}\left(1-\bar{\alpha}_{t-1}\right)}{1-\bar{\alpha}_t}\boldsymbol{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\left(1-\alpha_t\right)}{1-\bar{\alpha}_t}\hat{\boldsymbol{x}}_t, \tag{5}$$

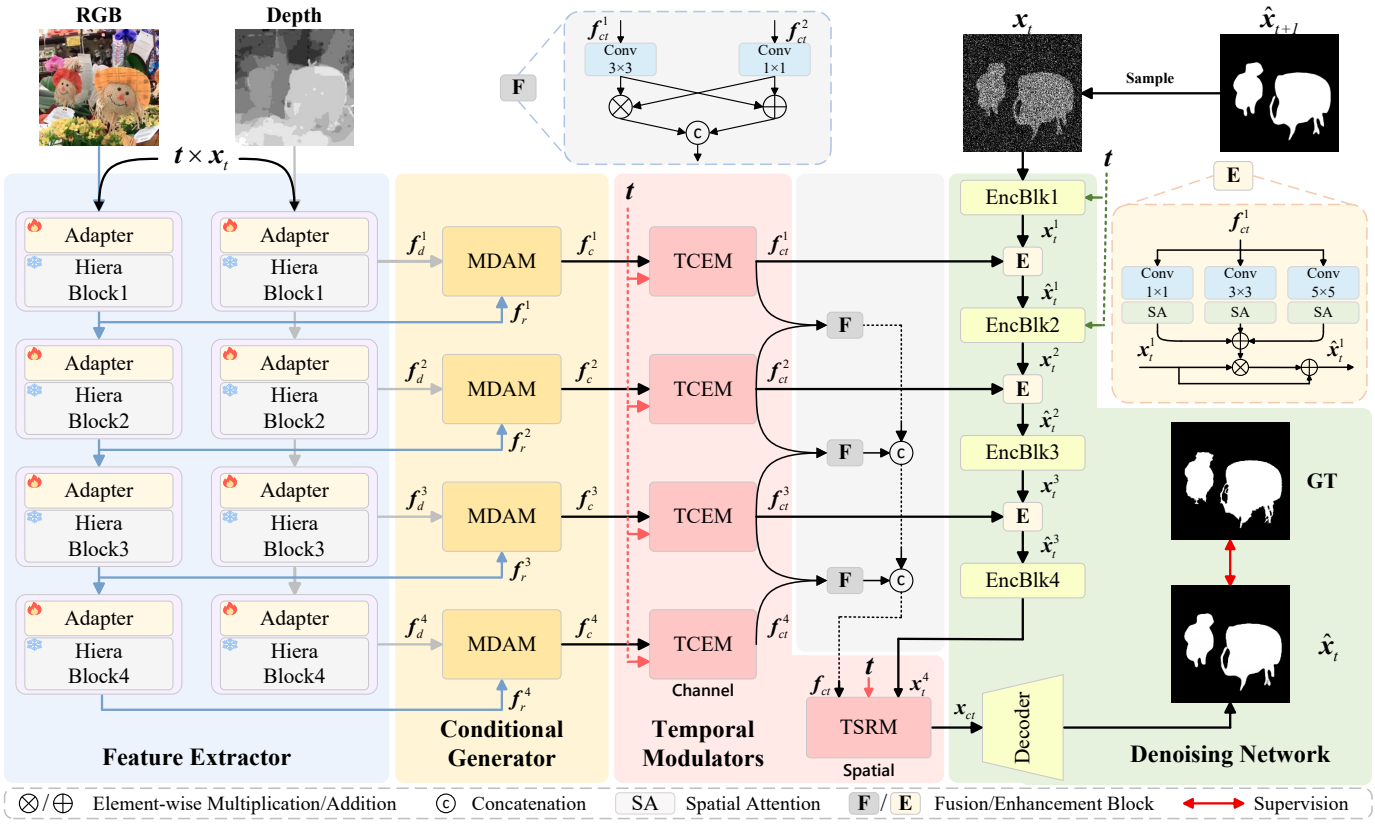

Fig. 3. The architecture of our DiffRGBD, which integrates RGB images and depth maps as conditional inputs for SOD. The SAM2 encoder with adapters processes these inputs to extract multi-level features, which are then fused to generate conditional information using the MDAM. Then, two temporal modulators adaptively adjust the features at each time step, ensuring the generation the more effective conditional information. Finally, the denoising network iteratively refines the noisy mask across time steps, improving the quality of the saliency map.

where σ_t^2 and $\mu(\boldsymbol{x}_t, t)$ represents the variance and mean of the normal distribution $\mathcal{N}$, respectively, and $\hat{\boldsymbol{x}}_t$ represents the output of the neural network f.

Specifically, for our DiffRGBD, the diffusion model is conditioned on the input RGB image and depth map, and the denoising process can be described as:

$$p(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{R}, \boldsymbol{D}) := \mathcal{N}(\boldsymbol{x}_{t-1}; \mu(\boldsymbol{x}_t, t, \boldsymbol{R}, \boldsymbol{D}), \sigma_t^2 t), \quad (6)$$

where $\boldsymbol{R}$ and $\boldsymbol{D}$ represent the input RGB image and depth map, respectively.

B. Network Overview

As illustrated in Fig. 3, our DiffRGBD consists of four key components of a feature extractor, a conditional generator (*i.e.*, MDAM), two temporal modulators (*i.e.*, TCEM and TSRM), and a denoising network. Given an RGB image and the corresponding depth map (both resolutions are 352×352), we adopt the SAM2 encoder (*i.e.*, Hiera) [25], [43] as the feature extractor to get hierarchical features. Specifically, along the diffusion process, the noise level in the denoising input $\boldsymbol{x}_t$ is correlated with the diffusion time step, resulting in strong time-dependent characteristics. To utilize the salient information from $\boldsymbol{x}_t$ while mitigating the influence of noise, we encode the time step into the time embedding $\boldsymbol{t} \in \mathbb{R}^{1\times352\times352}$ with the same dimension as $\boldsymbol{x}_t$, and then modulate $\boldsymbol{x}_t$ through multiplying. This temporal modulation adaptively regulates the contribution of $\boldsymbol{x}_t$ across diffusion steps, enabling effective exploitation of saliency cues at low-noise stages while suppressing noisy interference at high-noise stages. The time-modulated $\boldsymbol{x}_t$ is then injected into the dual-branch encoder by the element-wise addition with the RGB image and the depth map, respectively.

In the feature extractor, the parameters of the Hiera block are frozen, and lightweight adapters [44] are inserted before each Hiera block for fine-tuning efficiently. Each adapter is implemented as a two-layer feed-forward network with GELU activation, where the feature channels are first projected to an intermediate dimension of 32 and then projected back to the original channel dimension, without altering the backbone feature size. The extracted four-level features are denoted as $\boldsymbol{f}_r^i$ and $\boldsymbol{f}_d^i$ ($i \in \{1, 2, 3, 4\}$) for RGB features and depth features, respectively. We then employ the MDAM to fuse RGB and depth features at each level, resulting in conditional features $\boldsymbol{f}_c^i$ that capture both common semantics and modality-specific cues of $\boldsymbol{f}_r^i$ and $\boldsymbol{f}_d^i$. These features undergo time-adaptive channel modulation in the TCEM, generating $\boldsymbol{f}_{ct}^i$, which are injected into the encoder of the denoising network for initial structural guidance through enhancement blocks. Fusion blocks then aggregate all modulated features into a unified representation $\boldsymbol{f}_{ct}$, which undergoes time-adaptive spatial modulation in the TSRM and refines the spatial structure of $\boldsymbol{x}_t^4$, generating $\boldsymbol{x}_{ct}$. $\boldsymbol{x}_{ct}$ is injected into the decoder of

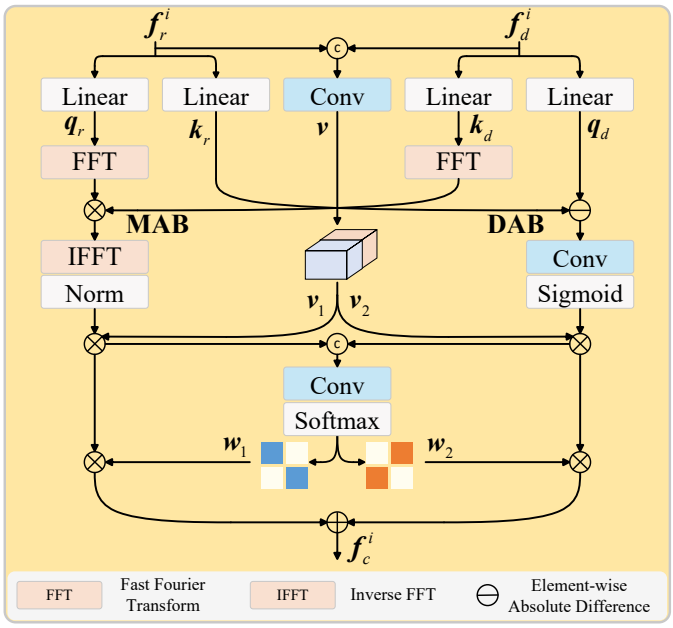

Fig. 4. Illustration of the Mutual-Differential Attention Module. The left part is the Mutual Attention Branch (MAB), while the right part is the Differential Attention Branch (DAB).

the denoising network to provide rich semantic guidance for generating $\hat{\boldsymbol{x}}_t$. In this way, the denoising network iteratively refines $\boldsymbol{x}_t$ across time steps, and generates the final saliency map.

C. Conditional Generator

For RGB-D SOD, the complementary nature of RGB images and depth maps naturally suggests utilizing the combined RGB-D features as conditional information. However, existing RGB-D SOD methods [10], [12], [33] often focus on capturing the common semantics between RGB images and depth maps to achieve cross-modal fusion, neglecting the fact that salient objects can appear differently across these modalities in complex scenes. This discrepancy makes it difficult for RGB-D SOD methods to reliably detect salient objects based solely on common features, leading to suboptimal performance.

To address this limitation, we propose the Mutual-Differential Attention Module (MDAM) as a conditional generator, which effectively fuses both common and modality-specific information between RGB images and depth maps. As illustrated in Fig. 4, our MDAM adopts a dual-branch architecture, and its inputs are $\boldsymbol{f}_r^i$ and $\boldsymbol{f}_d^i$. The mutual attention branch captures the common semantic content of RGB and depth features through frequency-domain point products, while the differential attention branch detects localized inter-modality discrepancies by computing absolute differences between them. Specifically, for RGB features $\boldsymbol{f}_r^i \in \mathbb{R}^{c_i \times h_i \times w_i}$ and depth features $\boldsymbol{f}_d^i \in \mathbb{R}^{c_i \times h_i \times w_i}$, which are first projected into query and key using linear layers, *i.e.*, $\boldsymbol{q}_r \in \mathbb{R}^{c_i \times h_i \times w_i}$ and $\boldsymbol{k}_r \in \mathbb{R}^{c_i \times h_i \times w_i}$ for $\boldsymbol{f}_r^i$ and $\boldsymbol{q}_d \in \mathbb{R}^{c_i \times h_i \times w_i}$ and $\boldsymbol{k}_d \in \mathbb{R}^{c_i \times h_i \times w_i}$ for $\boldsymbol{f}_d^i$. $\boldsymbol{f}_r^i$ and $\boldsymbol{f}_d^i$ are meanwhile concatenated and passed through a convolutional layer to generate the initial RGB-D fusion features $\boldsymbol{v} \in \mathbb{R}^{2c_i \times h_i \times w_i}$. $\boldsymbol{v}$ is split along the channel dimension into two parts, *i.e.*, $\boldsymbol{v}_1 \in \mathbb{R}^{c_i \times h_i \times w_i}$ and $\boldsymbol{v}_2 \in \mathbb{R}^{c_i \times h_i \times w_i}$, for processing in the mutual attention branch and the differential attention branch.

In the mutual attention branch (the left part of MDAM in Fig. 4), we compute the cross-modal attention by performing a Fast Fourier Transform (FFT) on $\boldsymbol{q}_r$ and $\boldsymbol{k}_d$ to transfer the attention calculation into the frequency domain. In this way, the computational complexity is reduced compared to the standard attention mechanism. The element-wise multiplication in the frequency domain helps capture the common semantics between the RGB and depth modalities. After performing an inverse FFT (IFFT) and normalization, the attention output is multiplied by $\boldsymbol{v}_1$, yielding a fusion feature that emphasizes the common semantic information shared by both modalities. This process is described as follows:

$$\boldsymbol{f}_{mutu} = Norm(F^{-1}(F(\boldsymbol{q}_r) \otimes F(\boldsymbol{k}_d))) \otimes \boldsymbol{v}_1, \tag{7}$$

where F represents FFT, F^{-1} represents IFFT, $Norm$ represents the normalization operation, and $\otimes$ is the element-wise multiplication.

In the differential attention branch (the right part of MDAM in Fig. 4), we compute the absolute difference between $\boldsymbol{k}_r$ and $\boldsymbol{q}_d$ to capture the modality-specific discrepancies between the RGB and depth features. This difference is passed through a convolutional layer and a sigmoid activation function to generate the differential attention weight, which is then multiplied by $\boldsymbol{v}_2$ to emphasize the unique characteristics of each modality. This process is described as follows:

$$\boldsymbol{f}_{diff} = \mathcal{S}(Conv(\boldsymbol{q}_r \ominus \boldsymbol{k}_d)) \otimes \boldsymbol{v}_2, \tag{8}$$

where $\ominus$ is the element-wise absolute difference, $Conv$ is a convolutional layer, and $\mathcal{S}$ is the sigmoid activation function.

To combine the outputs of the two branches, we concatenate the two outputs and apply a fusion convolutional layer, followed by a softmax normalization. The output is then split along the channel dimension to obtain two sets of attention weights, *i.e.*, $\boldsymbol{w}_1 \in \mathbb{R}^{c_i \times h_i \times w_i}$ and $\boldsymbol{w}_2 \in \mathbb{R}^{c_i \times h_i \times w_i}$. These weights are used to adaptively combine the outputs from both branches, producing the conditional information $\boldsymbol{f}_c^i \in \mathbb{R}^{c_i \times h_i \times w_i}$, which effectively captures both common semantics and modality-specific differences. This process is described as follows:

$$\boldsymbol{f}_c^i = (\boldsymbol{f}_{mutu} \otimes \boldsymbol{w}_1) \oplus (\boldsymbol{f}_{diff} \otimes \boldsymbol{w}_2), \tag{9}$$

where $\oplus$ is the element-wise summation.

D. Temporal Modulators

Existing diffusion models typically use condition information indiscriminately at different time steps. However, for the denoising process of diffusion models in SOD, the early denoising process requires global contextual information to mitigate the impact of heavy noise, while the late denoising process relies more on local fine-grained details to achieve object reconstruction. In other words, the requirements for conditional information vary greatly in different denoising time steps. Thus, we propose two temporal modulators, *i.e.*, the Temporal Channel Enhancement Module (TCEM) and the

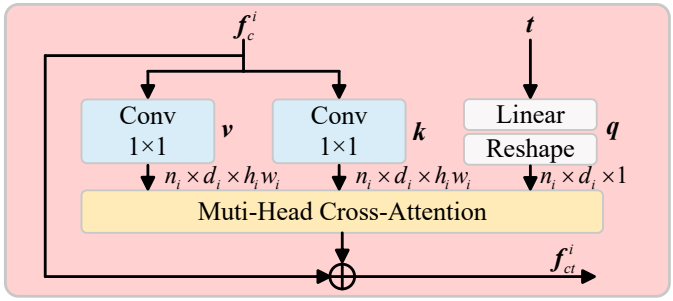

Fig. 5. Illustration of the Temporal Channel Enhancement Module.

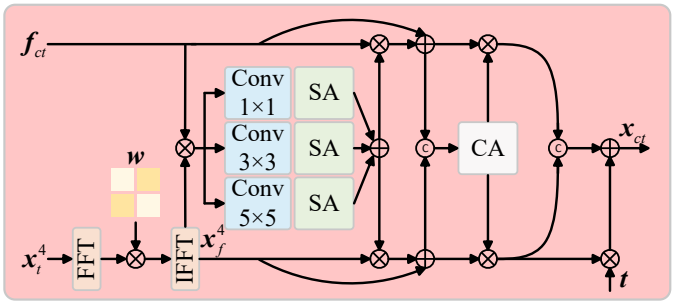

Fig. 6. Illustration of the Temporal Spatial Refinement Module.

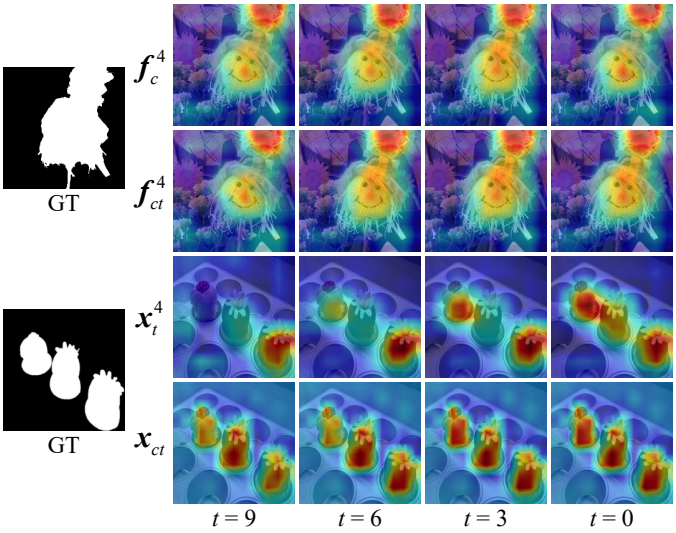

Fig. 7. Feature visualization of before and after temporal modulation at different time steps.

Temporal Spatial Refinement Module (TSRM), to modulate the conditional information at each time step, significantly enhancing the effectiveness of the conditional information.

1) Temporal Channel Enhancement Module. TCEM performs an effective time-conditioned channel modulation through multi-head cross-attention. As illustrated in Fig. 5, given the conditional feature $\boldsymbol{f}_c^i \in \mathbb{R}^{c_i \times h_i \times w_i}$ and the time embedding $\boldsymbol{t} \in \mathbb{R}^c$, we first project $\boldsymbol{f}_c^i$ into $\boldsymbol{k} \in \mathbb{R}^{n_i \times d_i \times h_i w_i}$ and $\boldsymbol{v} \in \mathbb{R}^{n_i \times d_i \times h_i w_i}$ using the convolutional layer. The time embedding $\boldsymbol{t}$ is then transformed via a linear layer to match the channel dimension, resulting in $\boldsymbol{q} \in \mathbb{R}^{n_i \times d_i \times 1}$. Then we modulate $\boldsymbol{f}_c^i$ through multi-head cross-attention [20]. In the attention computation, $\boldsymbol{q}$ is broadcast across the spatial dimensions, which reduces the computational cost and achieves efficient channel-wise modulation. This design is effective because the time step does not carry spatial or semantic information. Using channel-wise modulation allows the network to avoid unnecessary fine-grained spatial attention, while still adapting features across different time steps. Finally, a residual connection is applied, generating the time-modulated conditional feature $\boldsymbol{f}_{ct}^i \in \mathbb{R}^{c_i \times h_i \times w_i}$. This process is described as follows:

$$\boldsymbol{f}_{ct}^i = \boldsymbol{f}_c^i \oplus MHCA(\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{v}), \tag{10}$$

where $MHCA$ is the multi-head cross-attention. As shown in Fig. 3, $\boldsymbol{f}_{ct}^i$ provides initial structural guidance for the denoising process through enhancement blocks. Meanwhile, we integrate four-level conditional features $\boldsymbol{f}_{ct}^i$ $(i \in \{1, 2, 3, 4\})$ through fusion blocks and concatenation operations, getting $\boldsymbol{f}_{ct}$.

2) Temporal Spatial Refinement Module. TSRM performs time-conditioned spatial modulation on $\boldsymbol{f}_{ct}$. As illustrated in Fig. 6, we apply FFT to convert the deepest feature of the encoder in the denoising network $\boldsymbol{x}_t^4$ to the frequency domain, then multiply it by a learnable weight matrix $\boldsymbol{w}$ to effectively suppress high-frequency noise. The resulting feature is then transformed back to the spatial domain through IFFT, generating $\boldsymbol{x}_f^4$. Next, we multiply $\boldsymbol{x}_f^4$ with $\boldsymbol{f}_{ct}$ and apply three parallel convolutional layers with different kernel sizes for the subsequent application of spatial attentions [45]. The three resulting spatial attention maps are added to generate a collaborative attention map containing multi-scale spatial information, which is multiplied by $\boldsymbol{f}_{ct}$ and $\boldsymbol{x}_f^4$, generating $\boldsymbol{f}_s$ and $\boldsymbol{x}_s$ that enhance spatial semantic features through residual connections. This process is described as follows:

$$\boldsymbol{f}_s = \boldsymbol{f}_{ct} \oplus (\boldsymbol{f}_{ct} \otimes (\sum\nolimits_{k=1}^{3} SA(C_{2k-1}(\boldsymbol{x}_f^4 \otimes \boldsymbol{f}_{ct})))), \tag{11}$$

$$\boldsymbol{x}_s = \boldsymbol{x}_f^4 \oplus (\boldsymbol{x}_f^4 \otimes (\sum\nolimits_{k=1}^{3} SA(C_{2k-1}(\boldsymbol{x}_f^4 \otimes \boldsymbol{f}_{ct})))), \tag{12}$$

where C_{2k-1} is a convolutional layer with kernel size of $(2k-1) \times (2k-1)$ and SA is the spatial attention.

Then, $\boldsymbol{x}_s$ and $\boldsymbol{f}_s$ are concatenated and further modulated through channel attention [46] to generate a collaborative channel attention map for channel enhancement of $\boldsymbol{f}_s$ and $\boldsymbol{x}_s$, generating $\boldsymbol{f}_c$ and $\boldsymbol{x}_c$. This process is described as follows:

$$\boldsymbol{f}_c = \boldsymbol{f}_s \otimes CA(Cat(\boldsymbol{f}_s, \boldsymbol{x}_s)), \tag{13}$$

$$\boldsymbol{x}_c = \boldsymbol{x}_s \otimes CA(Cat(\boldsymbol{f}_s, \boldsymbol{x}_s)), \tag{14}$$

where Cat is the concatenation operation and CA is the channel attention. We fuse $\boldsymbol{f}_c$ and $\boldsymbol{x}_c$ through the concatenation, generating $\hat{\boldsymbol{x}}_{ct}$, which is enriched with rich semantic information from the conditional information. To further balance the dominance of $\boldsymbol{x}_c$ and $\boldsymbol{f}_c$ at the different time steps in the denoising process, we encode the time step into time embedding $\boldsymbol{t} \in \mathbb{R}^{1 \times 11 \times 11}$ with the same dimension as $\boldsymbol{x}_c$, and then multiply them. Finally, we add it with $\hat{\boldsymbol{x}}_{ct}$, generating the semantically refined $\boldsymbol{x}_{ct}$. This process is described as follows:

$$\boldsymbol{x}_{ct} = \hat{\boldsymbol{x}}_{ct} \oplus (\boldsymbol{x}_c \otimes \boldsymbol{t}). \tag{15}$$

As shown in Fig. 3, $\boldsymbol{x}_{ct}$ provides rich semantic guidance for the denoising process.

To further demonstrate the effects of the proposed temporal modulators, we visualize feature maps before and after TCEM (*i.e.*, $\boldsymbol{f}_c$ and $\boldsymbol{f}_{ct}$), as well as before and after TSRM (*i.e.*, $\boldsymbol{x}_t^4$ and $\boldsymbol{x}_{ct}$), under different time steps in Fig. 7. We observe that the features exhibit more concentrated responses on salient regions while suppressing background interference after temporal modulation. Moreover, as the diffusion process progresses, the feature responses become progressively more refined and better aligned with salient objects.

E. Denoising Network and Loss Function

1) Denoising Network. As shown in Fig. 3, the denoising network employs a four-level encoder-decoder architecture. Its first two encoder blocks (*i.e.*, EncBlk1 and EncBlk2) consist of two convolutional blocks and a residual block for temporal information injection, while the remaining encoder blocks (*i.e.*, EncBlk3 and EncBlk4) consist of two convolutional blocks. The convolutional blocks are responsible for feature transformation and spatial resolution adjustment, whereas the residual blocks are designed to inject time embeddings into the feature representations. Enhancement block is inserted after the first three encoder blocks to provide initial structural guidance for the denoising process using the conditional information $\boldsymbol{f}_{ct}^i$. This process of enhancement block can be described as follows:

$$\hat{\boldsymbol{x}}_t^i = \boldsymbol{x}_t^i \oplus (\boldsymbol{x}_t^i \otimes (\sum\nolimits_{k=1}^{3} SA(C_{2k-1}(\boldsymbol{f}_{ct}^i)))). \tag{16}$$

Corresponding to the encoder, the decoder consists of four convolutional blocks, which receive $\boldsymbol{x}_{ct}$ and output $\hat{\boldsymbol{x}}_t$.

2) Loss Function. We use a hybrid loss function composed of weighted Binary Cross-Entropy (BCE) loss and weighted Intersection-over-Union (IoU) loss to train our DiffRGBD. The total loss function L is formulated as follows:

$$L = \ell_{bce}^w(\hat{\boldsymbol{x}}_t, \mathbf{G}) + \ell_{iou}^w(\hat{\boldsymbol{x}}_t, \mathbf{G}), \tag{17}$$

where $\hat{\boldsymbol{x}}_t$ is the output of our denoising network, $\mathbf{G}$ is the ground truth, and ℓ_{bce}^w and ℓ_{iou}^w are the weighted BCE loss and the weighted IOU loss, respectively.

IV. Experiments

A. Experimental Setup

1) Datasets. To fully evaluate the performance of our DiffRGBD, we conduct extensive experiments on seven RGB-D SOD datasets. DUT [49] contains 1200 images captured in real life by the Lytro camera. LFSD [50] contains 100 images collected by the Lytro camera, which are labeled by multiple annotators. NJU2K [65] contains 1985 images collected from 3-D movies and daily life. NLPR [48] contains 1000 images captured by Microsoft Kinect cameras, and the images often contain multiple salient objects. SIP [63] contains 929 images captured by Huawei mobile phones, with the salient objects being primarily human figures. SSD [64] contains 80 images captured from indoor and outdoor scenes of 3-D movies. STERE [47] contains 1000 images collected from public websites. Following the previous methods [10], [12], we use 700 images from NLPR, 1485 images from NJU2K, and 800 images from DUT as the training set, with the remaining images used for testing.

2) Evaluation Metrics. We employ five widely used metrics for evaluation, including mean absolute error (M) [66], average F-measure (F_m) [67], average E-measure (E_m) [68], S-measure (S_m) [69] and weighted F-measure (F_m^ω) [70]. Specifically, M is used to evaluate the pixel-level error between the saliency map and the ground truth. F_m computes the weighted mean of precision and recall. E_m is used to evaluate the statistical properties at the image level and the pixel matching degree in local regions between the ground truth and the saliency map. S_m is used to evaluate the structural similarity between the ground truth and the saliency map. F_m^ω enhances F_m by incorporating weighted precision and recall, enabling it to more effectively address imbalanced saliency distributions.

3) Implementation Details. We implement our DiffRGBD using PyTorch [71] with an NVIDIA GTX 4090 GPU. We adopt the SAM2 encoder with adapters as the backbone network. In the training and testing phases, we resize the RGB images and depth maps to 352×352. We use the AdamW optimizer for parameter optimization with an initial learning rate of $1e^{-4}$, a batch size of 6, and a total of 150 training epochs. We apply several data augmentation strategies, such as random flipping, color jittering, and rotation, to mitigate the over-fitting problem. During sampling, we set the total number of time steps, *i.e.*, T, to 10. We adopt a signal-to-noise ratio-based variance schedule [41] to regulate the noise level across time steps, which facilitates stable training and effective progressive refinement.

B. Comparison with State-of-the-arts

We compare our DiffRGBD with eighteen SOTA RGB-D SOD methods on seven public RGB-D benchmarks, including DSNet [51], DCFNet [52], CIRNet [53], CFIDNet [54], SwinNet [16], DIGRNet [55], C2DFNet [56], HINet [57], CAVER [58], PICRNet [10], CATNet [17], RD3D+ [59], LAFB [60], MAGNet [61], CPNet [12], FasterSal [62], EM-Trans [33], and HENet [15]. The saliency maps of the methods mentioned above are sourced from the authors or generated by running public source codes.

1) Quantitative Evaluation. We report the quantitative results of our method and other methods on seven datasets in Tab. I and Tab. II. To compare the overall performance across seven datasets, we also report the average metrics of all methods on these datasets in Tab. II. It can be observed that our DiffRGBD outperforms all methods on seven datasets, except for E_m and S_m on the NJU2K dataset. On the STERE, DUT, LFSD, SIP, and SSD datasets, our DiffRGBD significantly outperforms state-of-the-art methods, while also achieving comparable performance to these methods on the NJU2K and NLPR datasets. Besides, on the average metrics across the seven datasets, our method outperforms the suboptimal CPNet by 0.9%, 0.8%, and 1.1% on F_m, S_m, and F_m^ω, respectively. In terms of F_m^ω, our methods outperform the representative HENet by 1.0% and 1.5% on the LFSD and SIP datasets.

TABLE I
QUANTITATIVE COMPARISONS ON STERE [47], NLPR [48], DUT [49], AND LFSD [50] DATASETS. THE TOP TWO RESULTS IN EACH COLUMN ARE HIGHLIGHTED IN **RED** AND **BLUE**.

Model	Pub.	Backbone	Params (M)↓	STERE					NLPR					DUT					LFSD	
				$M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	$M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	$M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	$M\downarrow$	$F_m\uparrow$
DSNet [51]	TIP'21	ResNet-50	-	.036	.894	.939	.915	.882	.024	.907	.943	.926	.886	.079	.807	.857	.841	.774	.069	.848
DCFNet [52]	CVPR'21	VGG-16	111.5	.039	.886	.939	.901	.875	.021	.898	.957	.923	.892	.071	.812	.888	.836	.766	.075	.835
CIRNet [53]	TIP'22	ResNet-50	103.2	.039	.890	.932	.915	.872	.023	.900	.952	.933	.889	.031	.923	.949	.932	.904	.068	.867
CFIDNet [54]	NCA'22	ResNet-50	393.5	.043	.881	.933	.901	.867	.026	.891	.947	.922	.881	.039	.903	.940	.916	.887	.071	.849
SwinNet [16]	TCSVT'22	Swin T	198.7	.033	.895	.947	.919	.889	.018	.919	.966	**.941**	.913	.021	.942	.969	.948	.933	.059	.874
DIGRNet [55]	TMM'23	ResNet-50	635.8	.038	.891	.940	.916	.877	.023	.904	.954	.935	.895	.033	.920	.946	.926	.898	.067	.851
C2DFNet [56]	TMM'23	ResNet-50	47.5	.038	.881	.937	.902	.871	.021	.905	.955	.927	.897	.026	.932	.958	.933	.918	.065	.859
HINet [57]	PR'23	ResNet-50	98.9	.049	.859	.918	.892	.839	.026	.887	.945	.922	.876	.054	.854	.903	.884	.826	.076	.829
CAVER [58]	TIP'23	ResNet-50	93.7	.033	.896	.947	.913	.889	.020	.906	.961	.928	.901	.042	.892	.932	.903	.874	.063	.864
PICRNet [10]	MM'23	Swin T	111.9	.031	.905	.951	.920	.898	.019	.916	.965	.935	.911	.021	.943	.967	.943	.933	.053	**.884**
CATNet [17]	TMM'24	Swin T	262.6	.030	.904	.952	.921	.900	.018	.922	.966	.940	.916	.020	.950	.971	**.953**	.942	.051	.878
RD3D+ [59]	TNNLS'24	ResNet-50	28.9	.039	.880	.932	.914	.867	.022	.898	.953	.933	.889	.031	.923	.952	.936	.908	.076	.831
LAFB [60]	TCSVT'24	ResNet-50	453.0	.040	.886	.936	.899	.870	.024	.901	.955	.924	.894	.032	.920	.953	.926	.906	.065	.857
MAGNet [61]	KBS'24	SMT	16.1	.030	.904	.951	.922	.892	.018	.918	.965	.939	.908	.021	.944	.967	.943	.935	.054	.878
CPNet [12]	IJCV'24	Swin T	216.5	**.029**	.903	**.954**	.920	.901	**.016**	**.925**	**.969**	.940	**.922**	**.019**	**.953**	**.972**	.951	**.948**	**.050**	**.884**
FasterSal [62]	TMM'25	MobileNet	3.4	.040	.873	.937	.888	.866	.022	.900	.957	.920	.898	.031	.921	.954	.920	.909	.063	.850
EM-Trans [33]	TNNLS'25	PVTv2	-	**.029**	**.913**	**.953**	.925	**.905**	**.017**	.920	.965	.940	.917	-	-	-	-	-	-	-
HENet [15]	TCSVT'25	MobileViT	11.9	**.029**	**.913**	.952	**.926**	.903	**.016**	.922	**.970**	**.942**	.918	.020	.945	.971	.949	.938	**.050**	.883
Ours	TCSVT'26	Hiera	337.4	**.027**	**.918**	**.954**	**.927**	**.913**	**.016**	**.927**	**.970**	**.942**	**.924**	**.016**	**.962**	**.975**	**.954**	**.955**	**.045**	**.891**

TABLE II
QUANTITATIVE COMPARISONS ON LFSD [50], SIP [63], SSD [64], AND NJU2K [65] DATASETS. ADDITIONALLY, WE REPORT THE AVERAGE SCORE OF EACH EVALUATION METRIC. THE TOP TWO RESULTS IN EACH COLUMN ARE HIGHLIGHTED IN **RED** AND **BLUE**.

Model	LFSD			SIP					SSD					NJU2K					**Average**				
	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	$M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	$M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	$M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$	$M\downarrow$	$F_m\uparrow$	$E_m\uparrow$	$S_m\uparrow$	$F_m^\omega\uparrow$
DSNet [51]	.889	.868	.826	.052	.863	.910	.876	.840	.045	.859	.906	.885	.838	.034	.907	.943	.921	.898	.048	.868	.913	.890	.849
DCFNet [52]	.878	.841	.805	.051	.875	.916	.875	.848	.049	.836	.903	.864	.814	.035	.903	.944	.911	.893	.049	.864	.918	.879	.842
CIRNet [53]	.890	.875	.838	.052	.875	.911	.888	.848	.049	.840	.897	.878	.816	.035	.908	.940	.925	.895	.042	.886	.924	.907	.866
CFIDNet [54]	.894	.870	.828	.060	.856	.899	.864	.825	.050	.850	.914	.879	.829	.038	.898	.937	.914	.886	.047	.875	.923	.895	.858
SwinNet [16]	.912	.886	.854	.035	.912	.942	.911	.896	.040	.865	.917	.892	.851	.027	.923	.955	.934	.917	.033	.904	.944	.919	.893
DIGRNet [55]	.892	.873	.828	.053	.879	.913	.885	.849	.053	.830	.889	.866	.804	.028	.918	.952	.933	.909	.042	.885	.927	.905	.866
C2DFNet [56]	.897	.863	.835	.052	.865	.912	.871	.841	.047	.847	.911	.872	.827	.038	.898	.936	.907	.885	.041	.884	.929	.896	.868
HINet [57]	.877	.852	.802	.066	.839	.886	.856	.805	.049	.836	.899	.865	.808	.039	.895	.933	.915	.881	.051	.857	.909	.884	.834
CAVER [58]	.907	.873	.844	.042	.889	.931	.892	.872	.041	.850	.919	.878	.834	.031	.914	.950	.920	.906	.039	.887	.935	.901	.874
PICRNet [10]	.917	.888	.864	.040	.901	.934	.898	.883	.047	.847	.917	.874	.832	.029	.919	.952	.927	.912	.034	.902	.943	.912	.890
CATNet [17]	.921	.894	.863	.035	.914	.944	.911	.897	-	-	-	-	-	**.026**	.927	.956	.932	.922	-	-	-	-	-
RD3D+ [59]	.876	.861	.807	.047	.881	.917	.891	.857	.044	.841	.900	.882	.820	.033	.909	.943	.927	.899	.042	.880	.925	.906	.864
LAFB [60]	.899	.864	.867	.051	.877	.918	.876	.850	.041	.864	.916	.882	.842	.033	.906	.945	.916	.897	.041	.887	.932	.898	.870
MAGNet [61]	.918	.889	.859	.037	.912	.943	.908	.888	.043	.858	.924	.885	.836	.028	.923	.956	.929	.911	.032	.905	.946	.916	.896
CPNet [12]	.921	.893	.869	.035	**.916**	.941	.907	**.900**	**.035**	**.876**	.930	**.894**	**.864**	**.025**	**.931**	**.959**	**.935**	**.926**	**.030**	**.913**	**.949**	**.920**	**.904**
FasterSal [62]	.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 [33]	-	-	-	.039	.910	.937	.903	.892	.039	.862	.931	.886	.847	.027	.925	.955	.930	.918	-	-	-	-	-
HENet [15]	**.924**	**.900**	**.870**	**.032**	**.916**	**.946**	**.914**	**.900**	.036	.863	**.932**	.893	.851	.027	.922	.955	**.934**	.916	.031	.902	.947	.918	.892
Ours	**.928**	**.904**	**.880**	**.031**	**.926**	**.949**	**.915**	**.915**	**.028**	**.895**	**.949**	**.906**	**.888**	**.025**	**.932**	**.957**	.931	**.927**	**.027**	**.922**	**.955**	**.928**	**.915**

In addition, to ensure fair comparison and provide clear insight into model complexity, we list the backbones and the number of learnable parameters of all methods in Tab. I. Our DiffRGBD has 337.4M parameters, which is comparable to recent high-performance models such as CPNet (216.5M) and LAFB (453.0M).

2) Qualitative Evaluation. We show the visual comparison of our DiffRGBD with nine SOTA RGB-D SOD methods in Fig. 8, including some challenging scenes, such as low contrast (*i.e.*, 1st and 2nd rows), multiple objects (*i.e.*, 3rd and 4th rows), complex scenes (*i.e.*, 5th and 6th rows) and images with similar foreground and background (*i.e.*, 7th and 8th rows). It can be seen that our DiffRGBD outperforms other methods in these complex scenes, demonstrating better detection results and stronger generalization ability.

C. Ablation Studies

To thoroughly analyze the effectiveness of our DiffRGBD, we conduct a series of ablation experiments on the DUT and NLPR datasets. Specifically, we assess the effectiveness of three modules (*i.e.*, MDAM, TCEM, and TSRM), the effectiveness of each component in MDAM and TSRM, the

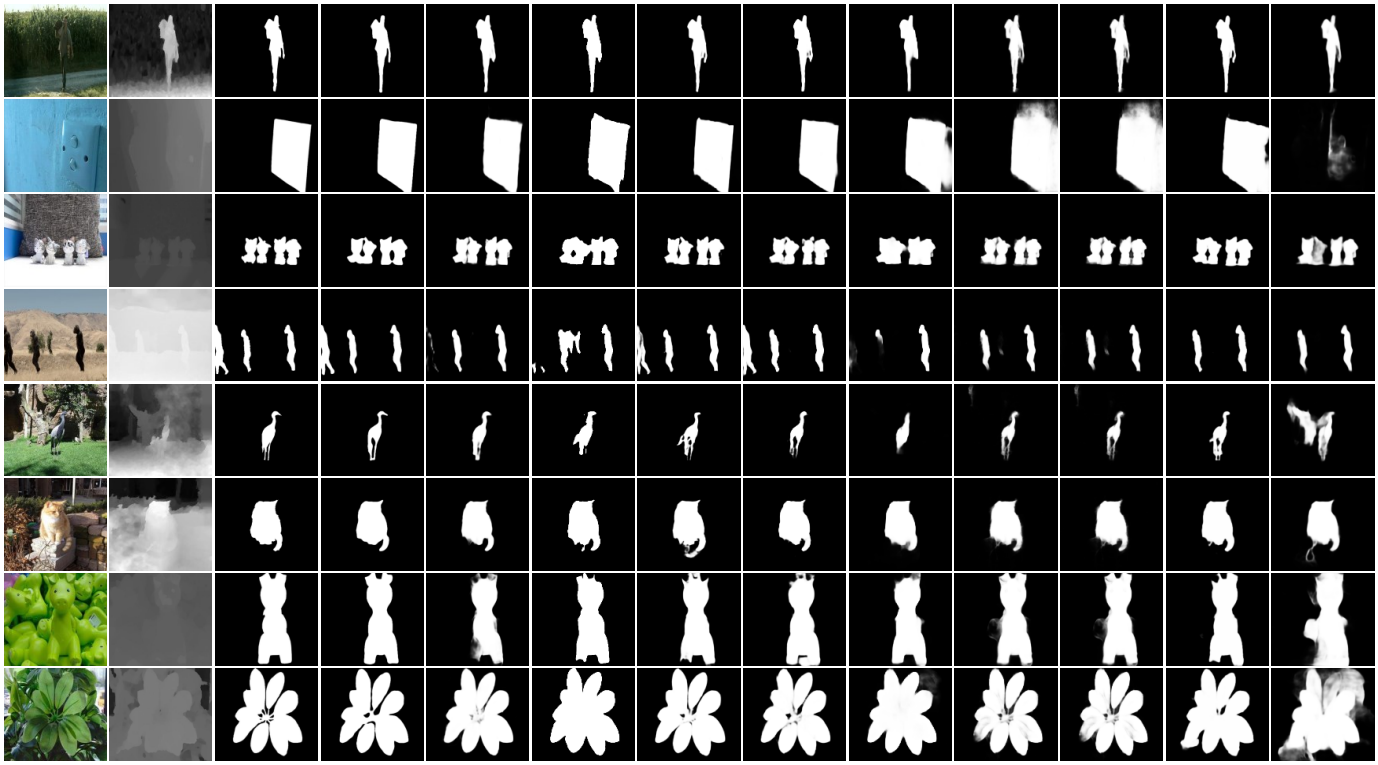

Fig. 8. Visual comparison with nine SOTA RGB-D SOD methods.

TABLE III
ABLATION STUDY ON THREE MODULES OF OUR DIFFRGBD. THE BEST RESULT OF EACH METRIC IS SHOWN IN **BOLD**.

No.	Baseline	MDAM	TCEM	TSRM	DUT			NLPR		
					$S_m\uparrow$	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$M\downarrow$	$E_m\uparrow$
1	✓				.940	.023	.964	.922	.022	.959
2	✓	✓			.949	.020	.970	.935	.020	.963
3	✓	✓	✓		.948	.019	.972	.935	.019	.964
4	✓	✓		✓	.949	.019	.971	.938	.017	.965
5	✓	✓	✓	✓	**.954**	**.016**	**.975**	**.942**	**.016**	**.970**

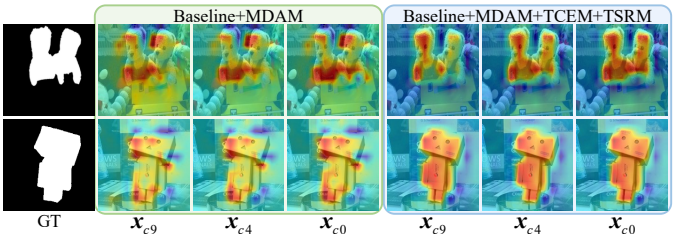

Fig. 9. Feature visualization of No. 2 and No. 5 in Tab. III.

effectiveness of different time steps, and the effectiveness of specific designs in feature extractor.

1) Effectiveness of Three Modules. We evaluate the effectiveness of the three modules via four variants, including Baseline, Baseline+MDAM, Baseline+MDAM+TCEM, and Baseline+MDAM+TSRM. Baseline is created by removing TCEM and replacing MDAM and TSRM with the element-wise summation operation. The experimental results are reported in Tab. III.

We can observe that Baseline+MDAM (No.2) achieves a significant improvement compared to Baseline (No.1) on the DUT and NLPR datasets, demonstrating the effectiveness of MDAM. From No.2 to No.4 of Tab. III, we can conclude that incorporating TCEM or TSRM individually into Baseline+MDAM leads to performance improvement, demonstrating the effectiveness of these two modules. By comparing the complete DiffRGBD with No.3 and No.4, it can be seen that the best performance is achieved when TCEM and TSRM are jointly utilized, demonstrating the necessity of the proposed two-stage temporal modulation strategy.

To provide an intuitive understanding of temporal modulation, we visualize feature maps before and after applying the two-stage temporal modulation strategy. Specifically, we visualize $\boldsymbol{x}_{ct}$ of No.2 and the complete DiffRGBD (No.5) in Fig. 9. $\boldsymbol{x}_{ct}$ is the feature generated by the proposed temporal modulators, as shown in Fig. 3. Compared with No.2, which does not incorporate temporal modulation, No.5 produces features with more concentrated responses on salient regions and suppressed background interference, and these responses become progressively more refined over time steps. This observation demonstrates that our two-stage temporal modulation strategy effectively enhances salient feature representations, thereby facilitating more accurate saliency prediction.

2) Effectiveness of Two Branches in MDAM. MDAM consists of a mutual attention branch and a differential attention branch. To verify the effectiveness of these two branches, we provide three variants: 1) removing the mutual attention

TABLE IV
ABLATION STUDY ON EACH COMPONENT OF MDAM AND TSRM. THE BEST RESULT OF EACH METRIC IS SHOWN IN **BOLD**.

	Variants	DUT			NLPR		
		$S_m\uparrow$	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$M\downarrow$	$E_m\uparrow$
MDAM	*w/o MAB*	.947	.021	.968	.935	.018	.963
	w/o DAB	.946	.019	.970	.934	.018	.965
	w/o weight	.949	.018	.971	.939	**.016**	.969
TSRM	*w/o FFT*	.948	.019	.970	.936	.018	.965
	w/o coll	.950	.017	.973	.933	.019	.966
	w/o t	.950	.018	.973	.939	.017	.968
	w/o CA	.952	.018	.973	.939	.017	.966
	w/o SA	.950	.019	.973	.938	.017	.967
	Ours	**.954**	**.016**	**.975**	**.942**	**.016**	**.970**

TABLE V
ABLATION STUDY ON DIFFERENT TIME STEPS. THE BEST RESULT OF EACH METRIC IS SHOWN IN **BOLD**.

Time steps	DUT			NLPR		
	$S_m\uparrow$	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$M\downarrow$	$E_m\uparrow$
T=5	.953	**.016**	.972	.940	.016	.968
T=15	.953	.017	.974	**.942**	**.015**	.969
T=10 **(Ours)**	**.954**	**.016**	**.975**	**.942**	.016	**.970**

TABLE VI
ABLATION STUDY ON SPECIFIC DESIGNS IN FEATURE EXTRACTOR. THE BEST RESULT OF EACH METRIC IS SHOWN IN **BOLD**.

Variants	DUT			NLPR		
	$S_m\uparrow$	$M\downarrow$	$E_m\uparrow$	$S_m\uparrow$	$M\downarrow$	$E_m\uparrow$
w/o adapters	.939	.024	.963	.933	.019	.965
w/o r_adapter	.949	.019	.970	.936	.018	.967
w/o d_adapter	.941	.022	.964	.935	.017	.967
w/o $\boldsymbol{t}$	.934	.024	.965	.929	.020	.963
w/o $\boldsymbol{t}\times\boldsymbol{x}_t$	.950	.018	.972	.934	.019	.965
Ours	**.954**	**.016**	**.975**	**.942**	**.016**	**.970**

branch (*i.e.*, *w/o MAB*), 2) removing the differential attention branch (*i.e.*, *w/o DAB*) and 3) removing the weighted fusion mechanism (*i.e.*, *w/o weight*). The experimental results are reported in the upper part of Tab. IV. We can observe that removing either branch of MDAM leads to a drop in detection performance. *w/o DAB* performs better than *w/o MAB* in terms of M (0.019 *v.s.* 0.021) and E_m (0.970 *v.s.* 0.968), indicating that the common semantics captured by the mutual attention branch play a more crucial role. Meanwhile, the differential attention branch further enhances detection performance by capturing modality-specific discrepancies. Furthermore, removing the weighted fusion strategy (*w/o weight*) results in decreased performance compared with the complete MDAM, demonstrating that the adaptive fusion of branch features is effective and necessary for balancing the contributions of MAB and DAB.

3) Effectiveness of Key Components in TSRM. TSRM provides rich semantic guidance for the denoising process through several key components, including the FFT operation, the collaborative modulation, and the temporal refinement. To evaluate the effectiveness of each component, we provide five variants: 1) removing the FFT operation and the learnable weight matrix (*i.e.*, *w/o FFT*), 2) removing the collaborative modulation by applying spatial and channel attentions to $\boldsymbol{f}_{ct}$ and $\boldsymbol{x}_t^4$ independently (*i.e.*, *w/o coll*), 3) removing the temporal refinement (*i.e.*, *w/o t*), 4) removing the channel attention (*i.e.*, *w/o CA*) and 5) removing the spatial attention (*i.e.*, *w/o SA*). The experimental results are reported in the bottom part of Tab. IV. We can observe that removing either component leads to a drop in detection performance, demonstrating the effectiveness of these three components. Compared with the complete DiffRGBD, *w/o FFT* prevents the high-frequency noise in $\boldsymbol{x}_t^4$ from being suppressed, limiting the effectiveness of subsequent modulation. *w/o coll* does not consider the adaptive alignment between $\boldsymbol{x}_t^4$ and $\boldsymbol{f}_{ct}$, resulting in misalignment in the regions of focus during independent attention modulation. *w/o t* does not consider that at different time steps, $\boldsymbol{f}_c$ and $\boldsymbol{x}_c$ play different roles in generating accurate saliency maps, leading to a performance drop. Moreover, *w/o CA* and *w/o SA* reveal that both channel and spatial attentions contribute positively to saliency prediction. Removing either attention reduces the ability of TSRM to selectively enhance informative features and suppress irrelevant responses.

4) Effectiveness of Different Time Steps. To evaluate the effectiveness of different time steps T, we conduct variants with T=5, 10, and 15, while keeping all other settings unchanged. As reported in Tab. V, the model with T=10 achieves the best overall performance on both datasets. When a smaller number of time steps (T=5) is used, the diffusion process is insufficient to fully correct prediction errors, leading to inferior results. In contrast, increasing the number of steps to T=15 does not bring further performance gains and even causes slight degradation in some metrics, indicating that excessive diffusion steps may introduce redundant refinements. These results demonstrate that a moderate diffusion step provides a favorable balance between effective saliency refinement and model stability. Therefore, we adopt T=10 in our DiffRGBD.

5) Effectiveness of Specific Designs in Feature Extractor. Our feature extractor includes two specific designs: modality-specific adapters and time-modulated noisy mask injection (*i.e.*, $\boldsymbol{t}\times\boldsymbol{x}_t$). Here, we first verify the effectiveness of modality-specific adapters with three variants: 1) removing adapters in both branches (*i.e.*, *w/o adapters*), 2) removing adapters in the RGB branch (*i.e.*, *w/o r_adapter*), and 3) removing adapters in the depth branch (*i.e.*, *w/o d_adapter*). Quantitative results in Tab. VI demonstrate that removing adapters leads to a consistent drop in detection performance on both datasets. *w/o adapters* produces the largest performance degradation, indicating that adapters are essential for effective feature encoding. Removing adapters in either branch also results in performance degradation. Specifically, *w/o r_adapter* performs worse than the complete model in terms of S_m (0.949 *v.s.* 0.954 on DUT), demonstrating the importance of RGB feature adaptation. Similarly, *w/o d_adapter* performs worse than the complete model in terms of M (0.022 *v.s.* 0.016 on DUT), indicating that adapters effectively improve structural

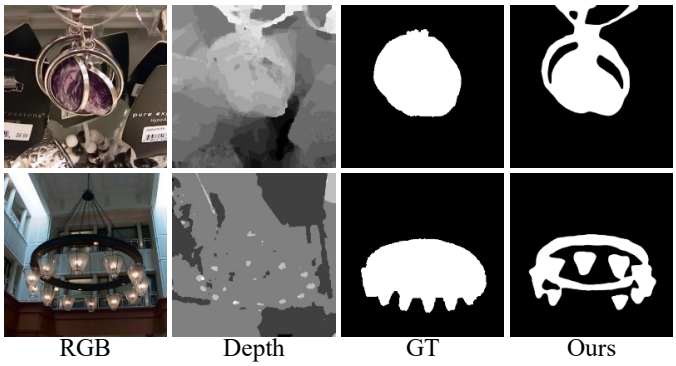

Fig. 10. Failure cases.

feature representation.

In addition, we verify the effectiveness of time-modulated noisy mask injection with two variants: directly injecting $\boldsymbol{x}_t$ without temporal modulation (*i.e. w/o* $\boldsymbol{t}$), and removing the injection of time-modulated noisy mask (*i.e. w/o* $\boldsymbol{t} \times \boldsymbol{x}_t$). As observed, both *w/o* $\boldsymbol{t}$ and *w/o* $\boldsymbol{t} \times \boldsymbol{x}_t$ perform worse than the complete model. Notably, *w/o* $\boldsymbol{t}$ performs worse than *w/o* $\boldsymbol{t} \times \boldsymbol{x}_t$, which can be attributed to the fact that directly injecting $\boldsymbol{x}_t$ may introduce modality conflicts and noise interference. Meanwhile, *w/o* $\boldsymbol{t} \times \boldsymbol{x}_t$ yields inferior performance, demonstrating that the saliency information in $\boldsymbol{x}_t$ contributes positively to performance improvement.

Overall, the best performance is achieved when both adapters and time-modulated feature injection are employed, demonstrating that the specific designs in our feature extractor effectively enhance feature representation.

D. Failure Cases and Limitations

Despite the strong performance of DiffRGBD, our method still encounters challenges in certain complex scenarios, as shown in Fig. 10. In the first case, inconsistent saliency cues between RGB and depth modalities lead to incomplete saliency prediction. In the second case, the depth map suffers from severe quality degradation, which introduces unreliable conditional guidance and affects the final prediction. These cases suggest that handling severe modality inconsistency remains a challenging problem and offers room for further improvement.

In addition, as a diffusion-based framework, our DiffRGBD requires iterative refinement over multiple time steps, resulting in a higher computational cost compared with end-to-end methods. As shown in Tab. V, the performance gap between T=5 and T=10 is relatively small. Here, we report the inference speed of T=5 and T=10, that is, 5.4 frames per second (FPS) of T=5 and 3.0 FPS of T=10. This indicates that diffusion models involve a trade-off between performance and inference efficiency. Therefore, in future work, we will explore more efficient diffusion strategies, such as adaptive time step scheduling and accelerated sampling, to dynamically balance refinement effectiveness and computational cost.

V. Conclusion

In this paper, we propose DiffRGBD, a novel diffusion-driven framework for RGB-D SOD. It follows a step-by-step generation paradigm to generate accurate saliency maps. Considering the varying conditional information requirements at different denoising stages, we adopt the two-stage temporal modulation strategy to modulate the conditional information at different time steps, achieving adaptive injection into the denoising network. Specifically, we first employ MDAM as a conditional generator to generate complementary condition information from the basic cross-modal features. Followed by two temporal modulators (*i.e.*, TCEM and TSRM), the conditional information is endowed with the ability to dynamically modulate over time steps and effectively injected into the denoising network. Through the step-by-step generation of predictions during the denoising process, our DiffRGBD generates accurate saliency maps. Comprehensive experiments on seven public datasets demonstrate the superiority of our DiffRGBD and the effectiveness of our two-stage temporal modulation strategy.

References

[1] X. Hu, C.-W. Fu, L. Zhu, T. Wang, and P.-A. Heng, "SAC-Net: Spatial attenuation context for salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 3, pp. 1079–1090, 2021.

[2] Y. Chen, G. Li, P. An, Z. Liu, X. Huang, and Q. Wu, "Light field salient object detection with sparse views via complementary and discriminative interaction network," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 2, pp. 1070–1085, 2024.

[3] M. Huang, G. Li, Z. Liu, and L. Zhu, "Lightweight distortion-aware network for salient object detection in omnidirectional images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 10, pp. 6191–6197, 2023.

[4] G. Li, Z. Liu, D. Zeng, W. Lin, and H. Ling, "Adjacent context coordination network for salient object detection in optical remote sensing images," *IEEE Trans. Cybern.*, vol. 53, no. 1, pp. 526–538, 2023.

[5] J. Jin, Q. Jiang, Q. Wu, B. Xu, and R. Cong, "Underwater salient object detection via dual-stage self-paced learning and depth emphasis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 35, no. 3, pp. 2147–2160, 2025.

[6] G. Li, Z. Liu, R. Shi, Z. Hu, W. Wei, Y. Wu, M. Huang, and H. Ling, "Personal fixations-based object segmentation with object localization and boundary preservation," *IEEE Trans. Image Process.*, vol. 30, pp. 1461–1475, 2021.

[7] W. Wang and J. Shen, "Deep cropping via attention box prediction and aesthetics assessment," in *Proc. IEEE ICCV*, Oct. 2017, pp. 2205–2213.

[8] X. Wang, J. Li, L. Zhu, Z. Zhang, Z. Chen, X. Li, Y. Wang, Y. Tian, and F. Wu, "VisEvent: Reliable object tracking via collaboration of frame and event flows," *IEEE Trans. Cybern.*, vol. 54, no. 3, pp. 1997–2010, 2024.

[9] B. Xu, Q. Jiang, X. Zhao, C. Lu, H. Liang, and R. Liang, "Multidimensional exploration of segment anything model for weakly supervised video salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 35, no. 4, pp. 2987–2998, 2025.

[10] 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.

[11] 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.

[12] 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, no. 8, pp. 3067–3085, Aug. 2024.

[13] 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.

[14] 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, 2021.

[15] 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.

[16] 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.

[17] 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.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015, pp. 1–14.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE CVPR*, 2016, pp. 770–778.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NeurIPS*, Dec. 2017, pp. 6000–6010.

[21] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE ICCV*, Oct. 2021, pp. 9992–10 002.

[22] 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, Sept. 2022.

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

[24] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. IEEE CVPR*, Jun. 2022, pp. 10 674–10 685.

[25] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollar, and C. Feichtenhofer, "SAM 2: Segment anything in images and videos," in *Proc. ICLR*, vol. 2025, 2025, pp. 28 085–28 128.

[26] 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, 2024.

[27] G. Li, Z. Bai, Z. Liu, X. Zhang, and H. Ling, "Salient object detection in optical remote sensing images driven by transformer," *IEEE Trans. Image Process.*, vol. 32, pp. 5257–5269, 2023.

[28] 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.

[29] 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, 2020.

[30] K. Fu, D.-P. Fan, G.-P. Ji, Q. Zhao, J. Shen, and C. Zhu, "Siamese network for RGB-D salient object detection and beyond," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5541–5559, 2022.

[31] Z. Zhang, Z. Lin, J. Xu, W.-D. Jin, S.-P. Lu, and D.-P. Fan, "Bilateral attention network for RGB-D salient object detection," *IEEE Trans. Image Process.*, vol. 30, pp. 1949–1961, 2021.

[32] J. He, K. Fu, X. Liu, and Q. Zhao, "Samba: A unified mamba-based framework for general salient object detection," in *CVPR*, Jun. 2025, pp. 25 314–25 324.

[33] 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.

[34] H. Xia, B. Bao, F. Liao, J. Chen, B. Wang, and Z. Li, "A patch-based method for underwater image enhancement with denoising diffusion models," *IEEE Trans. Cybern.*, vol. 55, no. 1, pp. 269–281, 2025.

[35] 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.

[36] S. Zhang, J. Huang, W. Tang, L. Tian, Y. Wei, and J. Liu, "Multi-modal salient object detection via a unified diffusion model," in *Proc. IEEE ICASSP*, 2025, pp. 1–5.

[37] Y. Ji, Z. Chen, E. Xie, L. Hong, X. Liu, Z. Liu, T. Lu, Z. Li, and P. Luo, "DDP: Diffusion model for dense visual prediction," in *Proc. IEEE ICCV*, 2023, pp. 21 684–21 695.

[38] Z. Chen, R. Gao, T.-Z. Xiang, and F. Lin, "Diffusion model for camouflaged object detection," in *Proc. ECAI*, vol. 372, 2023, pp. 445–452.

[39] S. Zhang, J. Huang, W. Tang, Y. Wu, T. Hu, X. Xu, and J. Liu, "DiMSOD: A diffusion-based framework for multi-modal salient object detection," in *Proc. AAAI*, vol. 39, no. 10, 2025, pp. 10 103–10 111.

[40] M. Song, L. Li, X. Yu, and C. Chen, "Pushing the boundaries of salient object detection: A denoising-driven approach," *IEEE Trans. Image Process.*, vol. 34, pp. 3903–3917, 2025.

[41] 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, Apr. 2025.

[42] 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, 2025.

[43] C. Ryali, Y.-T. Hu, D. Bolya, C. Wei, H. Fan, P.-Y. Huang, V. Aggarwal, A. Chowdhury, O. Poursaeed, J. Hoffman *et al.*, "Hiera: A hierarchical vision transformer without the bells-and-whistles," in *Proc. ICML*, Jul. 2023, pp. 29 441–29 454.

[44] X. Xiong, Z. Wu, S. Tan, W. Li, F. Tang, Y. Chen, S. Li, J. Ma, and G. Li, "SAM2-UNet: Segment anything 2 makes strong encoder for natural and medical image segmentation," *arXiv preprint arXiv:2408.08870*, 2024.

[45] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proc. IEEE CVPR*, Jun. 2019, pp. 3141–3149.

[46] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE CVPR*, Jun. 2018, pp. 7132–7141.

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

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

[49] 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*, Nov. 2019, pp. 7253–7263.

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

[51] 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.

[52] W. Ji, J. Li, S. Yu, M. Zhang, Y. Piao, S. Yao, Q. Bi, K. Ma, Y. Zheng, H. Lu *et al.*, "Calibrated RGB-D salient object detection," in *Proc. IEEE CVPR*, Jun. 2021, pp. 9466–9476.

[53] 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.

[54] 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.

[55] 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.

[56] 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.

[57] 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.

[58] 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.

[59] 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.

[60] 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.

[61] 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.*, vol. 299, p. 112126, 2024.

[62] 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.

[63] 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, May. 2020.

[64] C. Zhu and G. Li, "A three-pathway psychobiological framework of salient object detection using stereoscopic technology," in *Proc. IEEE ICCVW*, Oct. 2017, pp. 3008–3014.
[65] R. Ju, L. Ge, W. Geng, T. Ren, and G. Wu, "Depth saliency based on anisotropic center-surround difference," in *Proc. IEEE ICIP*, Oct. 2014, pp. 1115–1119.
[66] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE CVPR*, Jun. 2012, pp. 733–740.
[67] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE CVPR*, Jun. 2009, pp. 1597–1604.
[68] 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.
[69] 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*, Oct. 2017, pp. 4558–4567.
[70] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps," in *Proc. IEEE CVPR*, Jun. 2014, pp. 248–255.
[71] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. NeurIPS*, Dec. 2019, pp. 8024–8035.