DiMSOD: A Diffusion-Based Framework for Multi-Modal Salient Object Detection

Anonymous submission

Abstract

Multi-modal salient object detection (SOD) through the integration of additional data such as depth or thermal information has become a significant task in computer vision during recent years. Traditionally, the challenges of identifying salient objects in RGB, RGB-D (Depth), and RGB-T (Thermal) images are tackled separately, which often leads to issues like poorly defined object edges or overconfident inaccurate predictions. Recent studies have shown that designing a unified end-to-end framework to handle all these three types of SOD tasks simultaneously is both necessary and difficult. To address this need, we propose a novel approach that treats multi-modal SOD as a conditional mask generation task utilizing diffusion models. Specifically, we introduce DiMSOD, which enables the concurrent use of local (depth maps, thermal maps) and global controls (images) within a unified model for progressive denoising and refined prediction. DiMSOD is efficient, only requiring fine-tuning of local control adapter on the existing stable diffusion model, which not only reduces the fine-tuning cost and model size, making it more viable for real-world applications, but also enhances the integration of multi-modal conditional controls. Additionally, we have developed modules including SOD-ControlNet, Feature Adaptive Network (FAN), and Feature Injection Attention Network (FIAN) to further enhance the model's performance. Extensive experiments demonstrate that DiMSOD efficiently detects salient objects across RGB, RGB-D, and RGB-T datasets, achieving superior performance compared to previous methods. Our code and datasets are accessible at: https://anonymous.4open.science/r/DiMSOD-0B47/.

Introduction

Salient Object Detection (SOD) aims to accurately detect and locate the most salient objects in a given image, mimicking the human visual perception system (Gao et al. 2023b). It serves as an essential preliminary step for numerous other computer vision applications, including object detection (Cheng et al. 2023), visual tracking (Li et al. 2023), image segmentation (Chen et al. 2024), and quality assessment (Zhai and Min 2020). Despite the recent progress in SOD (Cai et al. 2024; Huo et al. 2024), most of these approaches primarily focus on processing individual RGB images. However, achieving accurate SOD results in challenging background and complex scenes remains difficult. In recent years, extensive use of depth cameras and infrared imaging devices have shown that the depth and thermal data gathered can significantly improve the performance of salient object detection. Nevertheless, the task of effectively combining multi-modal information without overestimating point estimates remains a considerable challenge, and it greatly influences the achievement of robust detection performance.

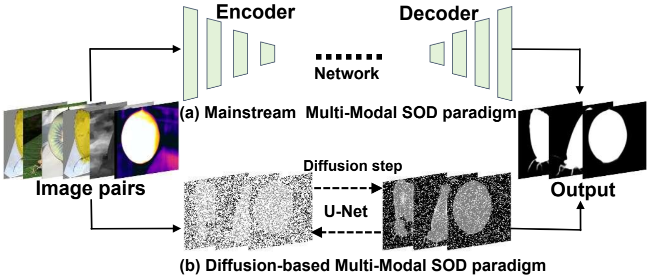
Encoder
Decoder
Network
(a) Mainstream Multi-Modal SOD paradigm
Image pairs
Output
Diffusion step
U-Net
(b) Diffusion-based Multi-Modal SOD paradigm

Figure 1: (a) Current SOD paradigm involves feeding images into the network for one-way prediction, resulting in a deterministic segmentation mask. (b) We propose a novel paradigm that decomposes SOD into a series of forward-and-reverse diffusion processes.

Currently, benefiting from the advancements of deep neural networks, SOD methods have evolved from designing ingenious low-level features (Jian and Yu 2023; Liu et al. 2017) to learning high-level models (Cai et al. 2024; Gu et al. 2023). By incorporating the advantages of dense feature interaction (Ma, Xia, and Li 2021; Cai et al. 2024), diverse attention module (Gu et al. 2023), and multi-task learning pipeline (Zong et al. 2023), deep learning-based methods have emerged as a promising technology for the SOD task. However, existing deep learning-based methods only focus on one specific type of input data. Considering the detection needs of RGB, RGB-D and RGB-T data, it is essential to develop a unified and comprehensive method to accommodate different data types. In response, a few recent studies (Gao et al. 2021; Jia et al. 2023; Luo et al. 2024) have focused on this direction. Although these methods have paved the way for the much-needed unified approach to multi-modal salient object detection, they still struggle to achieve precise localization and segmentation in complex scenarios. This limitation stems from their adherence to the paradigm shown in Fig. 1 (a), where a deterministic network solution generates a single output for a given input image. As a result, they fail to effectively integrate multi-modal information while avoiding overconfident incorrect predictions.

Given the unique challenges presented by multi-modal SOD, we propose using a diffusion model paradigm. As illustrated in Fig. 1 (b), we reformulate the multi-modal SOD task into a generative process, training the model to produce salient object masks by constructing a conditional noise-to-mask paradigm. Diffusion models (Ho, Jain, and Abbeel 2020a) have recently shown exceptional efficacy in generative modeling, particularly in conditional generation tasks (Dhariwal and Nichol 2021). Their inherent iterative denoising mechanism replaces the need for complex refinement modules in popular multi-modal SOD models, allowing for gradual distinction between salient object boundaries and background context. The random sampling process enables the generation and evaluation of multiple predictions, which further reduces the risk of the model making overconfident and erroneous estimations. The integration of ControlNet (Zhang, Rao, and Agrawala 2023) with diffusion models introduces cross-modal information, thereby providing better guidance for the denoising process. However, applying diffusion models and ControlNet to multi-modal SOD directly still faces several shortcomings, including limited discriminative ability, inadequate mask refinement, high fine-tuning costs, and relatively unstable controllability. To address these, we have tailored our method, DiMSOD, which leverages the denoising process of diffusion models to progressively correct the discrepancies between the initial noise and the ground truth. Depth maps and thermal maps are utilized as local auxiliary control conditions, while the image itself serves as a global auxiliary control condition. We have also improved upon ControlNet and proposed SOD-ControlNet, embedding our proposed Feature Adaptive Network (FAN) into the ControlNet and altering the method of conditional injection. By employing a multi-scale conditional injection strategy, we inject the introduced cross-modal information from depth maps and thermal maps into all resolutions. This significantly enhances the expressive power of DiMSOD, reduces model size and fine-tuning costs and aids in accurately identifying salient objects. Furthermore, to effectively bridge the gap between the diffusion noise embeddings and the conditional semantic features when integrating global image control into the model, we designed a Feature Injection Attention Network (FIAN). This network enhances the denoising process by aggregating the conditional semantic features of the image with the features from the diffusion model encoder through a cross-attention mechanism.

To summarize, our contributions are as follows:

- We are the first to formulate the multi-modal SOD as a generative denoising process and propose a stable diffusion-based model, DiMSOD. It can identify salient objects by denoising noisy masks to generate object masks based on the input images, depth maps, and thermal maps.
- We introduce SOD-ControlNet, specifically designed for multi-modal SOD. It integrates our proposed Feature Adaptive Network, effectively injecting depth and thermal condition information across all resolutions, thereby enabling efficient cross-modal information fusion.
- We have also designed a Feature Injection Attention Network to facilitate the interaction between noise embeddings and image features, thereby integrating global semantic information from the image to enhance the denoising process.

Related Work

SOD is a fundamental task in computer vision (Li et al. 2024; Xia et al. 2024). There are currently numerous methods focused on SOD for individual types of data, such as RGB, RGB-D, or RGB-T (Pang et al. 2023; Lee et al. 2023; Konwer et al. 2023). However, as solving multi-modal SOD using one single model is a relatively novel field of study, there are only a limited number of methods available currently, among which the most notable ones are MMNet (Gao et al. 2021), AiONet (Jia et al. 2023), and VSCode (Luo et al. 2024). In MMNet (Gao et al. 2021), a cross-modal multi-stage fusion module (CMFM) is proposed, which consists of two stages: feature response and adversarial combination. This module explores the complementarity of information from different modalities. In AiONet (Jia et al. 2023), a multi-modal feature extraction (MMFE) framework is proposed, which concurrently extracts features from RGB, depth, and thermal modalities. This framework aims to mitigate performance degradation caused by interference among multi-modal features. VSCode (Luo et al. 2024) utilizes visual saliency transformer as the foundational model and incorporates 2D prompts and discrimination loss within the encoder-decoder architecture. This approach facilitates the learning of both domain and task-specific knowledge, as well as shared knowledge.

Diffusion models (Cao et al. 2024; Graikos et al. 2022) sample noisy images using a forward Gaussian diffusion process and refine them iteratively through a backward denoised process to generate images. Diffusion models have demonstrated significant potential across various fields, such as image super-resolution (Gao et al. 2023a), image synthesis (Gu et al. 2022), image inpainting (Zhang et al. 2023), depth estimation (Ke et al. 2023), medical image segmentation (Zhang, Rao, and Agrawala 2023), and semantic segmentation (Ji et al. 2023). Different from these works, we propose the first diffusion-based model for multi-modal SOD. Moreover, our proposed SOD-ControlNet, Feature Adaptive Network, and Feature Injection Attention Network modules are perfectly aligned with the requirements of multi-modal SOD, effectively addressing challenges that other multi-modal SOD approaches have been unable to overcome.

Method

Network Architecture

We have developed DiMSOD, a model built upon a pretrained text-to-image LDM, Stable Diffusion v2 (Rombach et al. 2022), which leverages the excellent image priors obtained from LAION-5B (Schuhmann et al. 2022). With minimal modifications to the model components, we have con-

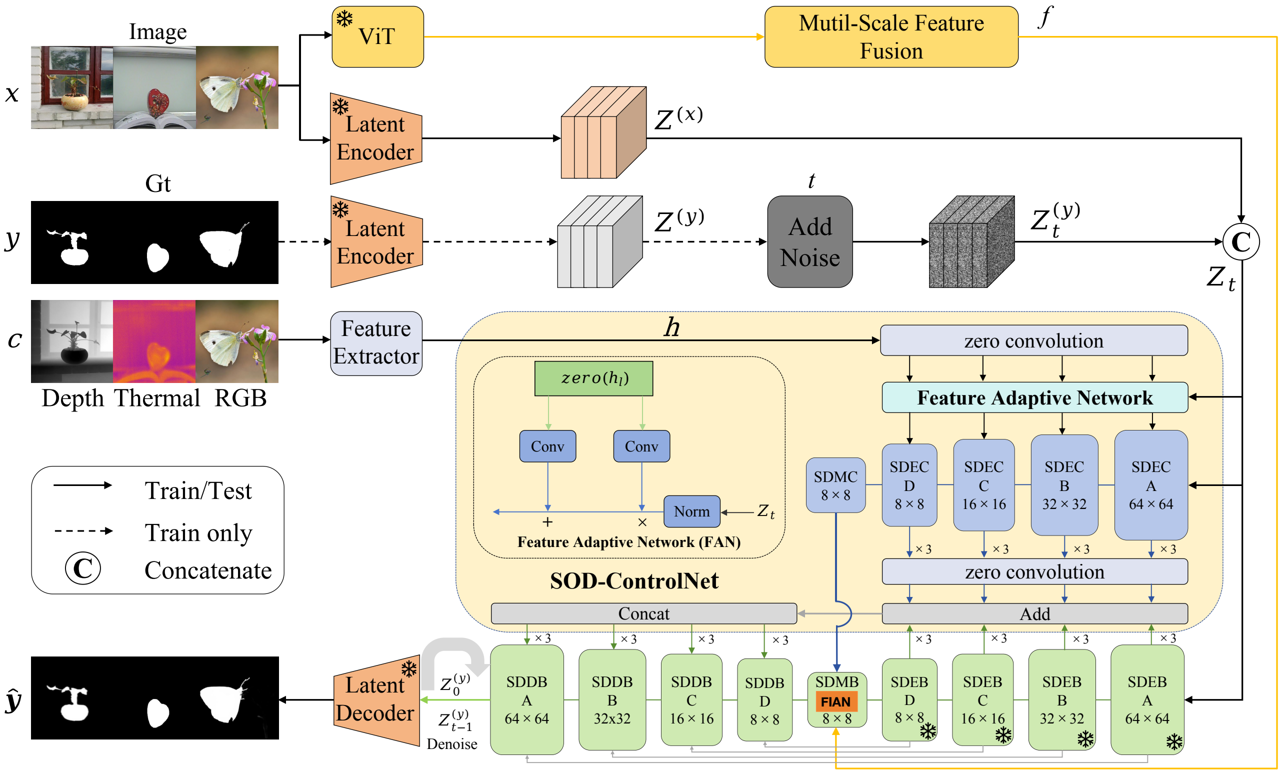

Figure 2: **Overview of the Architecture of DiMSOD**. We encode the image $\mathbf{x}$ and saliency map $\mathbf{y}$ into the latent space using the pre-trained Stable Diffusion VAE and fine-tune the U-Net by optimizing the noise loss relative to the saliency latent code. The SOD-ControlNet is proposed to leverage cross-modal information, such as depth maps and thermal maps, to control the generation of saliency masks. The Feature Injection Attention Network (FIAN) is introduced to implicitly guide the diffusion process with the corresponding conditional semantic features between diffusion information $\mathbf{z}_t$ and image features f that have been processed through the ViT backbone.

verted it into a multi-modal salient object detector. Fig. 2 provides an overview of the network architecture of DiMSOD. We introduced SOD-ControlNet and an Feature Adaptive Network to extract conditional semantic features and cross-modality interaction features from RGB images and depth map (thermal map), resulting in condition features rich in multi-scale details. Moreover, we developed a feature injection attention network based on cross-attention, enabling the salient object and localization information in the global conditional semantic features to guide the denoising process. These modules effectively integrate cross-modal information into the diffusion model, bridging the gap between diffusion features and image features, and thereby guiding the denoising network to generate more refined salient object boundaries and accurate predictions.

Saliency Encoder, Decoder and Local Feature Extractor. We start by encoding both the input image and its associated saliency map into a latent space using the frozen VAE. To accommodate the saliency map, we expand it to three channels to mimic an RGB image. This adjustment is crucial because the encoder was originally designed for 3-channel (RGB) inputs, whereas the saliency map has only a single channel. Through our experience and practice, we have found that saliency maps can be reconstructed from the encoded latent codes without any modifications to the latent space structure or the VAE. During inference, the predicted saliency map is obtained by averaging the three channels of the saliency latent code after it has been decoded at the end of the diffusion process. Our feature extractor is composed of a stack of convolutional layers and SiLU activations, enabling the extraction of conditional features at various resolutions. Additionally, the feature extractor projects the conditions into the corresponding latent spaces of different encoding layers, enhancing the alignment between the local conditional features and noise features.

Local Control Adapter. SD employs a U-Net-like (Ronneberger, Fischer, and Brox 2015) architecture as its denoising model, consisting primarily of an encoder, a middle block, and a decoder. Each encoder and decoder contains 12 corresponding blocks. Inspired by ControlNet (Zhang, Rao, and Agrawala 2023), we introduced SOD-ControlNet, as depicted in Fig. 2. In this illustration, each SDEB and SDDB represents an encoder block and a decoder block, respectively, while each SDEC and SDDC represents the copied versions of SDEB and SDDB. The diagram shows four blocks, each of which needs to be repeated three times. Additionally, SDMB denotes the middle block, and SDMC is the copied version of SDMB. For brevity, we denote the output of the i-th block in SDEB and SDDB as e_i and d_i . Similarly, $e^{'}$ and $g^{'}$ denote the output of the i-th block in SDEC and SDDC, while m and m' represent the output of SDMB and SDMC, respectively. Due to the skip connections in the U-Net and our intention to incorporate the local control information from the SOD-ControlNet during the decoding process, we modify the input for the i-th decoder block as:

$$\begin{cases} concat(m+m', e_j + Z(e_j^{'})) & i=1, i+j=13. \\ concat(d_{i-1}, d_j + Z(e_j^{'})) & 2 \leq i \leq 12, i+j=13. \end{cases}$$

where Z signifies a zero convolutional layer with weights that progressively increase from zero to gradually embed control information into SD. Our SOD-ControlNet differs from ControlNet in the way it handles conditions. While ControlNet directly adds conditions to the input noise and injects them into copied encoders, we employ a multi-scale condition injection strategy, adapting the conditions to all resolutions before injecting them into the copied encoders. Specifically, we begin by extracting multi-resolution conditional features (Depth, Thermal, RGB) using a feature extractor composed of stacked convolutional layers. We then select the first block from each resolution level (i.e., 64×64, 32×32, 16×16, and 8×8) within the copied encoder (i.e., the SDEC in Fig. 2) for condition injection. Inspired by the Feature Denormalization technique in SPADE (Park et al. 2019) , we develop the Feature Adaptive Network (FAN) for the injection process. FAN can modulate the normalization (i.e., $Norm \, \|\cdot\|$) of the input noise features using conditional features, as detailed below:

$$FAN_l(Z_t, h_l) = \|Z_t\| \cdot (1 + c_\gamma(Z(h_l))) + c_\beta(Z(h_l)),$$

where Z_t represents noise features, with a resolution of l. The h_l denotes the features obtained from local conditions c (Depth map, Thermal map, RGB image) after passing through the feature extractor, with a resolution of l. Learnable convolutional layers, c_γ and c_β, are used to transform condition features into spatially-sensitive scale and shift modulation coefficients.

Multi-scale Feature Fusion and Feature Injection Attention Network. For an RGB image $\mathbb{R}^{H \times W \times 3}$, we use the Swin Transformer (Gao et al. 2019) as our visual backbone to extract the top three high-level image features f_i, $i \in \{1, 2, 3\}$, with resolutions of $\frac{H}{s} \times \frac{W}{s}$, where $s \in \{8, 16, 32\}$. These features are then combined using multi-scale feature fusion. The feature fusion process consists of three branches for processing f_i each enhancing features via two 3×3 convolutional kernels. The features from these three branches are then aggregated through convolution and downsampling operations, resulting in image features f with a size of $\mathbb{R}^{\frac{H}{64} \times \frac{W}{64} \times C}$.

To incorporate salient information and semantic details from the original input features during the denoising process, we propose the Feature Injection Attention Network (FIAN), which is integrated into the the UNet-based denoising network. We use the multi-scale feature f and the deepest diffusion feature $e_{12} \in \mathbb{R}^{\frac{H}{64} \times \frac{W}{64} \times C}$ from SDEB as inputs to FIAN. In detail, we utilize e_{12} to generate the query $\mathbf{Q}$, key $\mathbf{K}$, and the value $\mathbf{V_1}$ by linear projection of a square matrix. Similarly, we use f to generate the $\mathbf{P}$ and $\mathbf{V_2}$. To reduce computational complexity and perform information weighting and fusion, we do not generate $\mathbf{Q}$ and $\mathbf{K}$ for f. Instead, we use $\mathbf{P}$ as an intermediary to connect with e_{12}. The details of FIAN are shown in Fig. 3, where $M_1 = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)$, $M_2 = \text{softmax}\left(\frac{KP^T}{\sqrt{d}}\right)$, $m = M_1 \times M_2 \times (V_1 + V_2)$. Here, the resolution of m is $\mathbb{R}^{\frac{H}{64} \times \frac{W}{64} \times C}$. However, it is important to note that m here is merely a placeholder and does not represent the complete output from SDMB. The final feature m in SDMB is obtained only after adding m to the features output by SDMC.

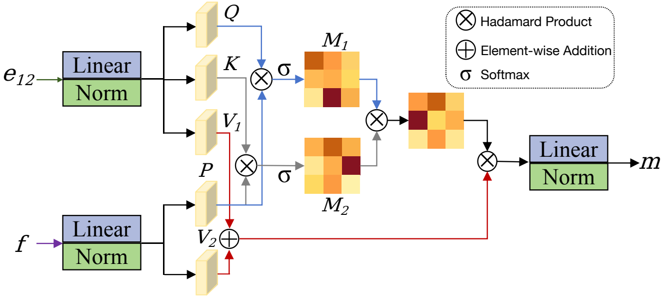

Figure 3: Details on how to conduct interactions between image features and the diffusion features using FIAN.

Training

We formulate multi-modal SOD as a conditional denoising diffusion generation task and train DiMSOD to fit the conditional distribution $D(\mathbf{y} \,|\mathbf{x}, c)$ over saliency $\mathbf{y} \in \mathbb{R}^{H \times W}$, where the global condition x is input image and the local condition c is the corresponding depth map or thermal map. In the *forward* process, which begins at $\mathbf{y}_0 := \mathbf{y}$ from the conditional distribution, Gaussian noise is gradually added to the ground-truth $\mathbf{y}_t$ at levels $t \in \{1, ..., T\}$ to obtain noisy mapping $\mathbf{y}_t$ as

$$\mathbf{y}_t = \sqrt{\bar{\alpha}_t}\mathbf{y}_0 + \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon}, \tag{1}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$, $\bar{\alpha}_t := \prod_{s=1}^{t} 1 - \beta_s$, and $\{\beta_1, \ldots, \beta_T\}$ denotes the variance schedule of a process with T steps. In the *reverse* process, the noise present in $\mathbf{y}_t$ is progressively eliminated to produce $\mathbf{y}_{t-1}$ using the conditional denoising model $\boldsymbol{\epsilon}_\theta(\cdot)$, which is parameterized by learned parameters.

To enable the input image $\mathbf{x}$ to conditionally guide the latent denoiser $\boldsymbol{\epsilon}_\theta(\mathbf{z}_t^{(\mathbf{y})}, \mathbf{z}^{(\mathbf{x})}, c, t)$, we concatenate the image latent code $\mathbf{z}^{(\mathbf{x})}$ and the saliency latent code $\mathbf{z}_t^{(\mathbf{y})}$ into a unified input $\mathbf{z}_t = \text{cat}(\mathbf{z}_t^{(\mathbf{y})}, \mathbf{z}^{(\mathbf{x})})$. Additionally, to enhance the use of saliency features, we overlay the saliency latent code onto the input image latent code, resulting in the transformed input represented as $\mathbf{z}_t = \text{cat}(\mathbf{z}_t^{(\mathbf{y})}, \mathbf{z}^{(\mathbf{x})} + \mathbf{z}_t^{(\mathbf{y})})$. This adjustment is intended to improve the convergence efficiency of the model without introducing predictive errors, as observed through experience. Following this, the input channels are doubled to accommodate the expanded input $\mathbf{z}_t$. To prevent the activation levels from inflating and to preserve the pre-trained structure as much as possible, we duplicate the weight tensor of the input layer and halve its values.

During training, the parameters θ are updated by first taking the input $(\mathbf{x} + \mathbf{y}, \mathbf{y})$ from the training data. The mask $\mathbf{y}$ is then noised with sampled multi-resolution noise $\boldsymbol{\epsilon}$ at a randomly selected timestep t. The noise estimate $\hat{\boldsymbol{\epsilon}}$ is computed using $\hat{\boldsymbol{\epsilon}} = \boldsymbol{\epsilon}_\theta(\mathbf{y}_t, \mathbf{x}, c, t)$. Finally, the denoising diffusion objective function $\mathbb{E}_{\mathbf{y}_0, c, \boldsymbol{\epsilon} \sim \mathcal{N}(0,I), t \sim \mathcal{U}(T)} \left| \boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}} \right|_2^2$ is minimized.

Latent diffusion models improve computational efficiency and image generation by performing diffusion in a low-dimensional latent space, which is created within the VAE's bottleneck and is trained separately from the denoiser (Rombach et al. 2022). To translate our formulation into the latent space, a latent code is defined as $\mathcal{E}$: $\mathbf{z}^{(\mathbf{y})} = \mathcal{E}(\mathbf{y})$, which is generated by the encoder. Given a saliency latent code, the saliency mask can be reconstructed using the decoder $\mathcal{D}$ as follows: $\hat{\mathbf{y}} = \mathcal{D}(\mathbf{z}^{(\mathbf{y})})$. The conditioning image $\mathbf{x}$ is also mapped into the latent space, resulting in $\mathbf{z}^{(\mathbf{x})} = \mathcal{E}(\mathbf{x})$. Subsequently, the denoiser is trained in the latent space as $\boldsymbol{\epsilon}_\theta(\mathbf{z}_t^{(\mathbf{y})}, \mathbf{z}^{(\mathbf{x})}, c, t)$. The adapted inference procedure introduces an additional step in which the decoder $\mathcal{D}$ reconstructs the data $\hat{\mathbf{y}}$ from the estimated clean saliency latent code $\mathbf{z}_0^{(\mathbf{y})}$: $\hat{\mathbf{y}} = \mathcal{D}(\mathbf{z}_0^{(\mathbf{y})})$.

Inference

During inference, $\mathbf{y} := \mathbf{y}_0$ is reconstructed by iteratively applying the denoiser $\boldsymbol{\epsilon}_\theta(\mathbf{y}_t, \mathbf{x}, c, t)$ to a normally distributed variable $\mathbf{y}_T$. We begin by initializing the saliency latent code with standard Gaussian noise and encoding the input image into the latent space. We then progressively denoise this latent code following the same schedule used during training. From our experience, we have observed that initializing with standard Gaussian noise yields better results compared to using multi-resolution noise, despite the model being trained with the latter. To expedite the inference process, we adopt the non-Markovian sampling with recalibrated steps as described in DDIM (Song, Meng, and Ermon 2020). Finally, using the VAE decoder, we generate the ultimate saliency map from the latent code and apply channel-wise averaging for post-processing.

Experiments

Experimental Setup

Our proposed DiMSOD is trained jointly using three different types of SOD datasets, following the recent work (Jia et al. 2023), our training dataset consists of the following subsets and resize it to 512×512 : the RGB dataset DUTS-TR (Wang et al. 2017) with 10,553 images, the RGB-T dataset VT5000 (Tu et al. 2022b) with 2,500 images, the RGB-D dataset NJUD (Ju et al. 2014) with 1,485 image, NLPR (Peng et al. 2014) with 700 images, and DUTLF-Depth (Piao et al. 2019) with 800 images. Stable Diffusion v2 (Rombach et al. 2022) is used as our backbone when implementing DiMSOD in PyTorch (Paszke et al. 2019). The initial pre-training configurations with a v-objective (Salimans and Ho 2022) are adhered to our experiments. In training, we implement the DDPM noise scheduler (Ho, Jain, and Abbeel 2020b) with 1,000 diffusion steps. For inference, we employ DDIM scheduler (Song, Meng, and Ermon 2020) and sample 50 steps. For the final prediction, we combine outcomes from 10 inference iterations initiated with diverse initial noise. Training our method takes 50 epoches with a batch size of 32. We adopt the Adam optimizer with a learning rate of 3×10^{-5}. We also implement training data augmentation through random horizontal and vertical flips. Training our DiMSOD until convergence spans around 1.5 days with a single Nvidia RTX 4090.

Evaluation Datasets and Metrics

For RGB datasets, we evaluate DiMSOD on 5 widely used benchmark datasets that are not seen during training, including DUT-OMRON (Yang et al. 2013) (5,168 images), ECSSD (Yan et al. 2013) (1,000 images), PASCAL-S (Li et al. 2014b) (850 images), HKU-IS (Li and Yu 2015) (4,447 images), and DUTS-TE (Wang et al. 2017) (5,019 Images). For RGB-D datasets , we use the test sets of DUTLF-Depth (Piao et al. 2019) (400 images), NJUD (Ju et al. 2014) (500 images), NLPR (Peng et al. 2014) (300 images), SIP (Fan et al. 2020a) (929 images), LFSD(Li et al. 2014a)(100 images). For RGB-T datasets ,we use the testset of VT5000(Tu et al. 2022b) (2,500 images) ,VT821(Wang et al. 2018)(821 images) , VT1000(Tu et al. 2019) (1,000 images).

Four metrics are evaluated on each dataset: **F$_\beta$-measure** (Achanta et al. 2009), **MAE**, **E-measure** (Fan et al. 2018), **S-measure** (Fan et al. 2017).

Comparisons with state-of-the-art

For all the RGB, RGB-D, and RGB-T experiments, we conducted comprehensive comparisons with state-of-the-art multi-modal SOD methods, including MMNet, AiONet, and VSCode. Additionally, for RGB SOD, we compared our approach against other specialized methods , namely, F^{3}Net, MINet(Pang et al. 2020),U^{2}Net (Qin et al. 2020), PFSNet (Ma, Xia, and Li 2021), VST(Liu et al. 2021), EDN (Wu et al. 2022), SHNet (Zhang et al. 2022), SRfor (Yun and Lin 2022), USOD (Zhou et al. 2023a), M^{3}Net (Yuan, Gao, and Tan 2023), and UTD (Huo et al. 2024). For RGB-D SOD, the compared methods are HDF-Net, CMWNet (Li et al. 2020), DANet, PGA-Net, BBS-Net (Fan et al. 2020b) , DD-CNN (Wang et al. 2022) and PICR.

For RGB-T SOD, the compared methods are R3Net, SGDL, M3S-NIR, ADF (Tu et al. 2022b), DCNet (Tu et al. 2022a), and LSNet (Zhou et al. 2023b). For fair comparisons, all results either come directly from the authors or are reproduced using the model retrained on the identical training dataset with the suggested settings. The code for evaluating the model is derived from F^{3}Net.

Quantitative Evaluation. For RGB SOD, the results are given in Table 1. We can find that DiMSOD outperformed in all metrics across the three benchmark datasets. Due to space limitations, detailed results for all datasets are provided in the supplementary materials.

Table 1: Quantitative comparisons between DiMSOD and other methods on three RGB SOD benchmark datasets.

Methods	DUTS-TE				HKU-IS				PASCAL-S			
	$F_\beta\uparrow$	$M\downarrow$	$E_\xi\uparrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$M\downarrow$	$E_\xi\uparrow$	$S_\alpha\uparrow$	$F_\beta\uparrow$	$M\downarrow$	$E_\xi\uparrow$	$S_\alpha\uparrow$
$\mathrm{F^3Net_{20}}$	.891	.035	.901	.888	.936	.028	.952	.917	.871	.061	.858	.854
$\mathrm{MINet_{20}}$	.883	.037	.897	.884	.934	.028	.953	.918	.866	.063	.850	.849
$\mathrm{U2Net_{20}}$	.872	.044	.886	.873	.935	.031	.948	.915	.859	.073	.842	.838
$\mathrm{MMNet^*{}_{21}}$	.877	.034	.911	.875	.927	.026	.953	.907	.862	.060	.858	.843
PFSNet_{21}	.896	.036	.902	.892	.943	.026	.956	.924	.875	.063	.856	.854
$\mathrm{VST_{21}}$	.890	.037	.891	.896	.942	.029	.952	.928	.875	.060	.837	.865
EDN_{22}	.893	.035	.904	.891	.939	.027	.948	.922	.879	.062	.857	.855
SHNet_{22}	.883	.030	.938	.908	.926	.025	.959	.926	.855	.056	.910	.884
$\mathrm{SRfor_{22}}$	.905	.029	.919	.904	.943	.025	.956	.928	.890	.052	.870	.873
$\mathrm{USOD_{23}}$	.810	.047	.901	.884	.902	.037	.947	.908	.831	.073	.890	.857
$\mathrm{AiONet^*{}_{23}}$	.856	.040	.927	.882	.933	.028	.921	.911	.873	.049	.873	.868
$\mathrm{M^3Net_{23}}$	.909	.026	.921	.908	.947	.024	.958	.931	.897	.050	.878	.879
$\mathrm{VSCode^*{}_{24}}$	.906	.032	.918	.902	.943	.027	.956	.924	.886	.059	.865	.862
$\mathrm{UTD_{24}}$	.904	.043	.926	.900	.933	.028	.926	.921	.854	.063	.845	.851
Ours	**.918**	**.024**	**.927**	**.919**	**.948**	**.023**	**.961**	**.937**	**.915**	**.041**	**.887**	**.881**

Table 2: Quantitative comparisons between DIMSOD and other methods on three RGB-D SOD benchmark datasets.

Methods	DUTD				LFSD				NJUD			
	$M\downarrow$	$F_\beta\uparrow$	$S_\alpha\uparrow$	$E_\xi\uparrow$	$M\downarrow$	$F_\beta\uparrow$	$S_\alpha\uparrow$	$E_\xi\uparrow$	$M\downarrow$	$F_\beta\uparrow$	$S_\alpha\uparrow$	$E_\xi\uparrow$
$\mathrm{PGANet_{20}}$	.048	.889	.894	.898	.071	.868	.865	.874	.035	.927	.925	.903
$\mathrm{DANet_{20}}$	.029	.936	.931	.933	.074	.854	.858	.829	.037	.914	.915	.917
$\mathrm{HDF_{20}}$	.025	.944	.933	.893	.066	.870	.866	.872	.030	.926	.919	.897
$\mathrm{MMNet^*{}_{21}}$	.039	.916	.913	.891	.071	.862	.863	.898	.040	.913	.911	.917
$\mathrm{BBSNet_{21}}$	.029	.934	.930	.916	.061	.879	.877	.857	.036	.918	.918	.912
$\mathrm{CMW_{21}}$	.036	.915	.905	.903	.087	.832	.827	.831	.037	.911	.910	.915
$\mathrm{AiONet\ ^*{}_{23}}$	.029	.933	.928	.933	.072	.859	.853	.864	.039	.907	.904	.908
$\mathrm{DDCNN_{23}}$	.024	.947	.941	.923	.054	.892	**.890**	.898	.035	.919	.922	.915
$\mathrm{PICR_{23}}$	.029	.938	.932	.924	.068	.883	.875	.848	.035	.928	.925	.881
$\mathrm{VSCode^*{}_{24}}$	.034	.927	.919	.925	.072	.862	.859	.869	.038	.910	.910	.908
OURS	**.020**	**.951**	**.943**	**.938**	**.053**	**.894**	.888	**.899**	**.029**	**.931**	**.927**	**.921**

Table 3: Quantitative comparisons between DiMSOD and other methods on three RGB-T SOD benchmark datasets.

Method	VT821				VT1000				VT5000			
	$S_\alpha\uparrow$	$E_\xi\uparrow$	$F_\beta\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\xi\uparrow$	$F_\beta\uparrow$	$M\downarrow$	$S_\alpha\uparrow$	$E_\xi\uparrow$	$F_\beta\uparrow$	$M\downarrow$
$\mathrm{R3Net_{18}}$	.786	.809	.660	.073	.842	.859	.761	.055	.757	.790	.615	.083
$\mathrm{M3S_{19}}$	.723	.859	.734	.140	.726	.827	.717	.145	.652	.780	.575	.168
$\mathrm{SGDL_{20}}$	.765	.847	.731	.085	.787	.856	.764	.090	.750	.824	.672	.089
$\mathrm{MMNet^*{}_{21}}$	.871	.895	.803	.045	.915	.933	.880	.027	.868	.896	.799	.043
$\mathrm{ADF_{22}}$	.810	.842	.717	.077	.910	.921	.847	.034	.864	.891	.778	.048
$\mathrm{DCNet_{22}}$	.877	.913	.822	.033	.923	.949	.902	.021	.872	.921	.819	.035
$\mathrm{LSNet_{23}}$	.877	.911	.827	.033	.924	.936	.887	.022	.876	.916	.827	.036
$\mathrm{AiONet\ ^*{}_{23}}$	.904	.937	.882	.028	.841	**.974**	**.941**	**.020**	.896	.937	.875	.035
$\mathrm{VSCode^*{}_{24}}$	.892	.923	.830	.029	.929	.941	.893	.024	.886	.926	.823	.033
Ours	**.912**	**.939**	**.897**	**.025**	**.933**	.955	.893	**.020**	**.906**	**.939**	**.881**	**.029**

It demonstrates the excellent performances of the proposed DiMSOD. On the three larger datasets, DUTS-TE, HKUIS, and PASCAL-S, we achieved the average improvements in F_β, MAE, E_ξ, and S_α are 1.2%, 4.6%, 3%, and 4%, respectively. For RGB-D SOD, as shown in Table 2, our method also achieves the best performance. It outperforms the second-best model by 3% and 1% in MAE and F_β, respectively. For RGB-T SOD, the overall results of our DiMSOD is almost the best, although there are some victories and defeats compared to AiONet on VT100. Our approach performs better on RGB and RGB-D datasets compared to RGB-T, indicating a high level of consistency between salient objects and depth information in many cases. Hence, we can conclude that by leveraging the proposed model framework, DiMSOD demonstrates a competitive advantage across RGB, RGB-D, and RGB-T datasets.

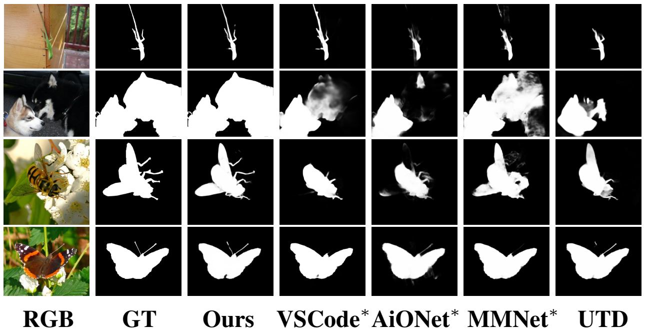

Figure 4: Visual comparison of saliency map results generated by various methods for RGB SOD.

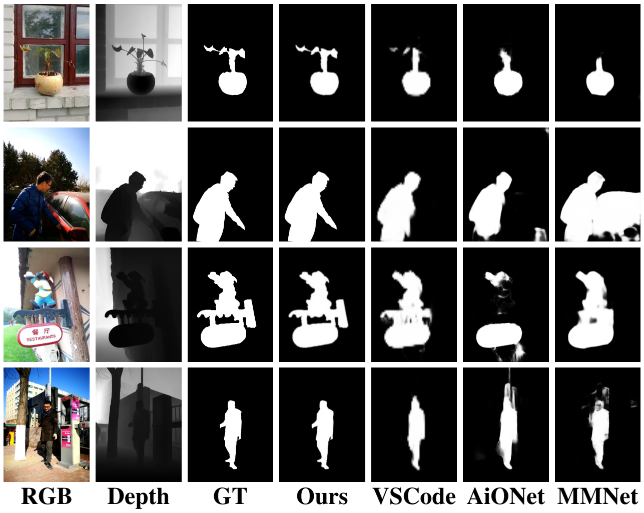

Figure 5: Visual comparison of saliency map results generated by various methods for RGB-D SOD.

Qualitative Evaluation. Fig. 4, Fig. 5, and Fig. 6 show the comprehensive visual comparison of many challenging samples, including complex backgrounds, rich edge details, small objects, and multiple salient objects. From the results, we can find that compared with other methods, our method exhibits good structural completeness and has more intricate details. For more experimental results, please refer to our supplementary materials. Besides, previous models have muddled the identification of edge components, even when accurately pinpointing the object's location. Nevertheless, DiMSOD captures intricate object textures effectively in an incredibly detailed way, addressing the segmentation mask blurring issue presented in other methods. More detailed output results can be seen in the last column of Fig. 7, where our model demonstrates excellent handling of the texture and edge details of the targets.

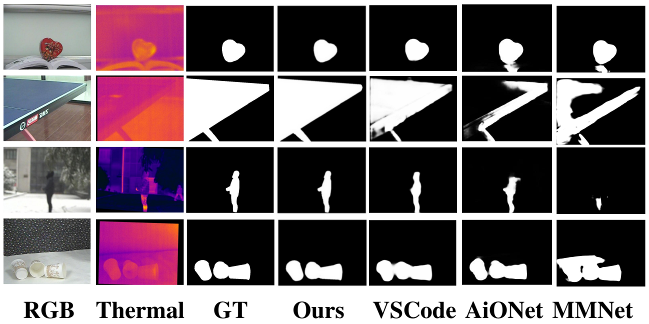

Figure 6: Visual comparison of saliency map results generated by various methods for RGB-T SOD.

Ablation Studies

As depicted in Table 4, experimental results demonstrate that ViT, SOD-ControlNet (SOD-CNet), Feature Injection Attention Network (FIAN) can all improve multi-model SOD performance very well. Combining them with Stable Diffusion (SD) led to significant improvements across all evaluation metrics.

Effectiveness of ViT backbone. From Table 4, we can see that No.2 has an average improvement of 3% and 7.8% over No.1 for F_β and MAE on the there types datasets, respectively. Fig. 7 illustrates how the critical clues identified by ViT are seamlessly incorporated into the diffusion process through the assistance of FF and FIAN.

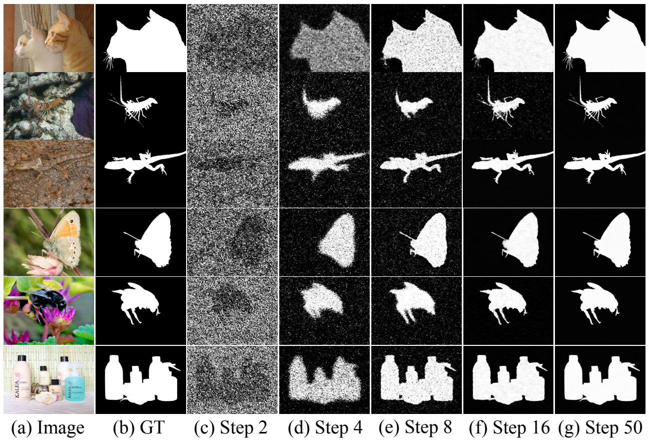

Figure 7: Visual results of the DiMSOD sampling process.

Effectiveness of SOD-ControlNet. The purpose of SOD-ControlNet is to address the issue of cross-modal information fusion in multi-modal SOD. As shown in Table 4, compared to No.2, No.3 has an average improvement of 9.7% and 17.5% for F_β and MAE on the there types datasets, respectively. Meanwhile, a comparison between No.3 and No.2 reveals that while directly using ControlNet does provide some performance improvement, it is evidently less effective than SOD-ControlNet.

Effectiveness of Feature Injection Attention Network. From Table 4, wo can find that FIAN plays a key role in improving the performance of the model. The average improvement of DiMSOD with FIAN over No.4 without FIAN for F_β and MAE on the three types of datasets is 2.3% and 13%, respectively. It indicates that FIAN effectively integrates both diffusion and salient features from the backbone in a cohesive manner.

Table 4: Ablation studies of DiMSOD. The best results are highlighted in bold.

No.	Settings	DUTS-TE		NJUD		VT1000	
		F_β↑	M↓	F_β↑	M↓	F_β↑	M↓
1	SD	.881	.039	.884	.043	.851	.041
2	SD+ViT	.890	.037	.924	.038	.873	.039
3	SD+ViT+CNet	.897	.029	.927	.033	.884	.031
4	SD+ViT+SOD-CNet	.905	.026	.930	.030	.891	.024
Ours	SD+ViT+SOD-CNet+FIAN	**.918**	**.024**	**.931**	**.029**	**.893**	**.020**

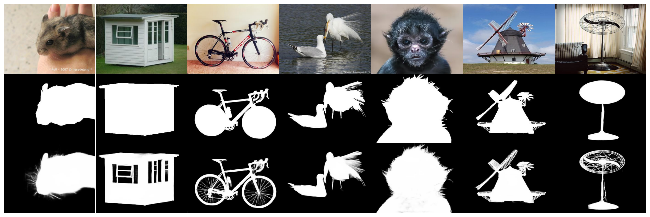

Figure 8: Remarkable results generated by DiMSOD.

Additionally, the outcomes from the sampling process at denoising steps 2, 4, 8, 16, and 50 were visualized in Fig. 7 . The core of DiMSOD lies in leveraging SOD-ControlNet to effectively integrate the rich visual priors stored in Stable Diffusion and cross-modal auxiliary information . This enhanced integration facilitates more accurate guidance in the generation of saliency masks. Despite being trained on relatively coarse SOD benchmark datasets, our model effectively segments the edges of salient objects, thanks to its robust visual priors. As illustrated in Fig. 8, the final results of salient object detection exhibit even greater precision and refinement compared to the ground-truth mask. Additional experimental results are available in the supplementary materials. We also offer the trained weights and inference code, enabling you to apply DiMSOD to your own images for direct experience.

Conclusion

In this paper, we presented DiMSOD, a diffusion-based framework for in RGB, RGB-D and RGB-T images. To our knowledge, this is the first framework to apply a denoising diffusion model to multi-modal SOD. DiMSOD decomposes multi-modal SOD into a series of forward and backward diffusion processes, leveraging key details from the semantic features under both global (image) and local conditions (depth map, thermal map) to guide the processes. Extensive quantitative and qualitative experiments demonstrate that DiMSOD outperforms other state-of-the-art methods across various benchmark datasets. Additionally, ablation studies confirm the effectiveness of the SOD-ControlNet and FIAN we introduced for multi-modal SOD. While our current model offers a strategy to balance accuracy and inference time, this is not a long-term solution. In the future, we will conduct further research and optimization to improve the model's inference efficiency.

References

Achanta, R.; Hemami, S.; Estrada, F.; and Susstrunk, S. 2009. Frequency-tuned salient region detection. In *2009 IEEE conference on computer vision and pattern recognition*, 1597–1604. IEEE.

Cai, X.; Wang, G.; Lou, J.; Jian, M.; Dong, J.; Chen, R.-C.; Stevens, B.; and Yu, H. 2024. Perceptual loss guided Generative adversarial network for saliency detection. *Information Sciences*, 654: 119625.

Cao, H.; Tan, C.; Gao, Z.; Xu, Y.; Chen, G.; Heng, P.-A.; and Li, S. Z. 2024. A survey on generative diffusion models. *IEEE Transactions on Knowledge and Data Engineering*.

Chen, K.; Liu, C.; Chen, H.; Zhang, H.; Li, W.; Zou, Z.; and Shi, Z. 2024. RSPrompter: Learning to prompt for remote sensing instance segmentation based on visual foundation model. *IEEE Transactions on Geoscience and Remote Sensing*.

Cheng, G.; Yuan, X.; Yao, X.; Yan, K.; Zeng, Q.; Xie, X.; and Han, J. 2023. Towards large-scale small object detection: Survey and benchmarks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34: 8780–8794.

Fan, D.-P.; Cheng, M.-M.; Liu, Y.; Li, T.; and Borji, A. 2017. Structure-measure: A new way to evaluate foreground maps. In *Proceedings of the IEEE international conference on computer vision*, 4548–4557.

Fan, D.-P.; Gong, C.; Cao, Y.; Ren, B.; Cheng, M.-M.; and Borji, A. 2018. Enhanced-alignment measure for binary foreground map evaluation.

Fan, D.-P.; Lin, Z.; Zhang, Z.; Zhu, M.; and Cheng, M.-M. 2020a. Rethinking RGB-D salient object detection: Models, data sets, and large-scale benchmarks. *IEEE Transactions on neural networks and learning systems*, 32(5): 2075–2089.

Fan, D.-P.; Zhai, Y.; Borji, A.; Yang, J.; and Shao, L. 2020b. BBS-Net: RGB-D salient object detection with a bifurcated backbone strategy network. In *ECCV*, 275–292. Springer.

Gao, S.; Liu, X.; Zeng, B.; Xu, S.; Li, Y.; Luo, X.; Liu, J.; Zhen, X.; and Zhang, B. 2023a. Implicit diffusion models for continuous super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10021–10030.

Gao, S.-H.; Cheng, M.-M.; Zhao, K.; Zhang, X.-Y.; Yang, M.-H.; and Torr, P. 2019. Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*, 43(2): 652–662.

Gao, W.; Fan, S.; Li, G.; and Lin, W. 2023b. A Thorough Benchmark and a New Model for Light Field Saliency Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Gao, W.; Liao, G.; Ma, S.; Li, G.; Liang, Y.; and Lin, W. 2021. Unified information fusion network for multi-modal RGB-D and RGB-T salient object detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(4): 2091–2106.

Graikos, A.; Malkin, N.; Jojic, N.; and Samaras, D. 2022. Diffusion models as plug-and-play priors. *Advances in Neural Information Processing Systems*, 35: 14715–14728.

Gu, S.; Chen, D.; Bao, J.; Wen, F.; Zhang, B.; Chen, D.; Yuan, L.; and Guo, B. 2022. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10696–10706.

Gu, Y.; Xu, H.; Quan, Y.; Chen, W.; and Zheng, J. 2023. Orsi salient object detection via bidimensional attention and full-stage semantic guidance. *IEEE Transactions on Geoscience and Remote Sensing*, 61: 1–13.

Ho, J.; Jain, A.; and Abbeel, P. 2020a. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Ho, J.; Jain, A.; and Abbeel, P. 2020b. Denoising Diffusion Probabilistic Models. arXiv:2006.11239.

Huo, F.; Liu, Z.; Guo, J.; Xu, W.; and Guo, S. 2024. UTDNet: A unified triplet decoder network for multimodal salient object detection. *Neural Networks*, 170: 521–534.

Ji, Y.; Chen, Z.; Xie, E.; Hong, L.; Liu, X.; Liu, Z.; Lu, T.; Li, Z.; and Luo, P. 2023. Ddp: Diffusion model for dense visual prediction. *arXiv preprint arXiv:2303.17559*.

Jia, X.; Zhao, Z.; Dongye, C.; and Zhang, Z. 2023. All in One: RGB, RGB-D, and RGB-T Salient Object Detection. *arXiv preprint arXiv:2311.14746*.

Jian, M.; and Yu, H. 2023. Towards reliable object representation via sparse directional patches and spatial center cues. *Fundamental Research*.

Ju, R.; Ge, L.; Geng, W.; Ren, T.; and Wu, G. 2014. Depth saliency based on anisotropic center-surround difference. In *2014 IEEE international conference on image processing (ICIP)*, 1115–1119. IEEE.

Ke, B.; Obukhov, A.; Huang, S.; Metzger, N.; Daudt, R. C.; and Schindler, K. 2023. Repurposing diffusion-based image generators for monocular depth estimation. *arXiv preprint arXiv:2312.02145*.

Konwer, A.; Hu, X.; Bae, J.; Xu, X.; Chen, C.; and Prasanna, P. 2023. Enhancing modality-agnostic representations via meta-learning for brain tumor segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 21415–21425.

Lee, Y.-L.; Tsai, Y.-H.; Chiu, W.-C.; and Lee, C.-Y. 2023. Multimodal Prompting with Missing Modalities for Visual Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14943–14952.

Li, G.; Liu, Z.; Ye, L.; Wang, Y.; and Ling, H. 2020. Cross-modal weighting network for RGB-D salient object detection. *ECCV*.

Li, G.; and Yu, Y. 2015. Visual saliency based on multiscale deep features. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5455–5463.

Li, J.; Ji, W.; Wang, S.; Li, W.; et al. 2024. DVSOD: RGB-D Video Salient Object Detection. *Advances in Neural Information Processing Systems*, 36.

Li, J.; Qiao, S.; Zhao, Z.; Xie, C.; Chen, X.; and Xia, C. 2023. Rethinking lightweight salient object detection via network depth-width tradeoff. *IEEE Transactions on Image Processing*.

Li, N.; Ye, J.; Ji, Y.; Ling, H.; and Yu, J. 2014a. Saliency detection on light field. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2806–2813.

Li, Y.; Hou, X.; Koch, C.; Rehg, J. M.; and Yuille, A. L. 2014b. The secrets of salient object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 280–287.

Liu, N.; Zhang, N.; Wan, K.; Shao, L.; and Han, J. 2021. Visual saliency transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, 4722–4732.

Liu, Q.; Hong, X.; Zou, B.; Chen, J.; Chen, Z.; and Zhao, G. 2017. Hierarchical contour closure-based holistic salient object detection. *IEEE Transactions on Image Processing*, 26(9): 4537–4552.

Luo, Z.; Liu, N.; Zhao, W.; Yang, X.; Zhang, D.; Fan, D.-P.; Khan, F.; and Han, J. 2024. VSCode: General Visual Salient and Camouflaged Object Detection with 2D Prompt Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17169–17180.

Ma, M.; Xia, C.; and Li, J. 2021. Pyramidal feature shrinking for salient object detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 2311–2318.

Pang, Y.; Zhao, X.; Zhang, L.; and Lu, H. 2020. Multi-scale interactive network for salient object detection. In *CVPR*, 9413–9422.

Pang, Y.; Zhao, X.; Zhang, L.; and Lu, H. 2023. CAVER: Cross-modal view-mixed transformer for bi-modal salient object detection. *IEEE Transactions on Image Processing*, 32: 892–904.

Park, T.; Liu, M.-Y.; Wang, T.-C.; and Zhu, J.-Y. 2019. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2337–2346.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Peng, H.; Li, B.; Xiong, W.; Hu, W.; and Ji, R. 2014. RGBD salient object detection: A benchmark and algorithms. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part III 13*, 92–109. Springer.

Piao, Y.; Ji, W.; Li, J.; Zhang, M.; and Lu, H. 2019. Depth-induced multi-scale recurrent attention network for saliency detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, 7254–7263.

Qin, X.; Zhang, Z.; Huang, C.; Dehghan, M.; Zaiane, O. R.; and Jagersand, M. 2020. U2-Net: Going deeper with nested U-structure for salient object detection. *Pattern recognition*, 106: 107404.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 234–241. Springer.

Salimans, T.; and Ho, J. 2022. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*.

Schuhmann, C.; Beaumont, R.; Vencu, R.; Gordon, C.; Wightman, R.; Cherti, M.; Coombes, T.; Katta, A.; Mullis, C.; Wortsman, M.; et al. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35: 25278–25294.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Tu, Z.; Li, Z.; Li, C.; and Tang, J. 2022a. Weakly alignment-free RGBT salient object detection with deep correlation network. *IEEE Transactions on Image Processing*, 31: 3752–3764.

Tu, Z.; Ma, Y.; Li, Z.; Li, C.; Xu, J.; and Liu, Y. 2022b. RGBT salient object detection: A large-scale dataset and benchmark. *IEEE Transactions on Multimedia*.

Tu, Z.; Xia, T.; Li, C.; Wang, X.; Ma, Y.; and Tang, J. 2019. RGB-T image saliency detection via collaborative graph learning. *IEEE Transactions on Multimedia*, 22(1): 160–173.

Wang, G.; Li, C.; Ma, Y.; Zheng, A.; Tang, J.; and Luo, B. 2018. RGB-T saliency detection benchmark: Dataset, baselines, analysis and a novel approach. In *Image and Graphics Technologies and Applications: 13th Conference on Image and Graphics Technologies and Applications, IGTA 2018, Beijing, China, April 8–10, 2018, Revised Selected Papers 13*, 359–369. Springer.

Wang, L.; Lu, H.; Wang, Y.; Feng, M.; Wang, D.; Yin, B.; and Ruan, X. 2017. Learning to detect salient objects with image-level supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 136–145.

Wang, X.; Zhu, L.; Tang, S.; Fu, H.; Li, P.; Wu, F.; Yang, Y.; and Zhuang, Y. 2022. Boosting RGB-D saliency detection by leveraging unlabeled RGB images. *IEEE Transactions on Image Processing*, 31: 1107–1119.

Wu, Y.-H.; Liu, Y.; Zhang, L.; Cheng, M.-M.; and Ren, B. 2022. EDN: Salient object detection via extremely-downsampled network. *IEEE Transactions on Image Processing*, 31: 3125–3136.

Xia, C.; Sun, Y.; Li, K.-C.; Ge, B.; Zhang, H.; Jiang, B.; and Zhang, J. 2024. RCNet: Related Context-Driven Network with Hierarchical Attention for Salient Object Detection. *Expert Systems with Applications*, 237: 121441.

Yan, Q.; Xu, L.; Shi, J.; and Jia, J. 2013. Hierarchical saliency detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1155–1162.

Yang, C.; Zhang, L.; Lu, H.; Ruan, X.; and Yang, M.-H. 2013. Saliency detection via graph-based manifold ranking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3166–3173.

Yuan, Y.; Gao, P.; and Tan, X. 2023. M^{3}Net: Multilevel, Mixed and Multistage Attention Network for Salient Object Detection. *arXiv preprint arXiv:2309.08365*.

Yun, Y. K.; and Lin, W. 2022. Selfreformer: Self-refined network with transformer for salient object detection. *arXiv preprint arXiv:2205.11283*.

Zhai, G.; and Min, X. 2020. Perceptual image quality assessment: a survey. *Science China Information Sciences*, 63: 1–52.

Zhang, G.; Ji, J.; Zhang, Y.; Yu, M.; Jaakkola, T. S.; and Chang, S. 2023. Towards coherent image inpainting using denoising diffusion implicit models.

Zhang, L.; Rao, A.; and Agrawala, M. 2023. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3836–3847.

Zhang, W.; Zheng, L.; Wang, H.; Wu, X.; and Li, X. 2022. Saliency hierarchy modeling via generative kernels for salient object detection. In *European Conference on Computer Vision*, 570–587. Springer.

Zhou, H.; Qiao, B.; Yang, L.; Lai, J.; and Xie, X. 2023a. Texture-Guided Saliency Distilling for Unsupervised Salient Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7257–7267.

Zhou, W.; Zhu, Y.; Lei, J.; Yang, R.; and Yu, L. 2023b. LSNet: Lightweight spatial boosting network for detecting salient objects in RGB-thermal images. *IEEE Transactions on Image Processing*, 32: 1329–1340.

Zong, M.; Wang, R.; Ma, Y.; and Ji, W. 2023. Spatial and temporal saliency based four-stream network with multitask learning for action recognition. *Applied Soft Computing*, 132: 109884.

Reproducibility Checklist

This paper:

- Includes a conceptual outline and/or pseudocode description of AI methods introduced (yes/partial/no/NA) (yes)
- Clearly delineates statements that are opinions, hypothesis, and speculation from objective facts and results (yes/no) (yes)
- Provides well marked pedagogical references for less-familiar readers to gain background necessary to replicate the paper (yes/no) (yes)

Does this paper make theoretical contributions? (yes/no) (yes)
If yes, please complete the list below.

- All assumptions and restrictions are stated clearly and formally. (yes/partial/no) (yes)
- All novel claims are stated formally (e.g., in theorem statements). (yes/partial/no) (yes)
- Proofs of all novel claims are included. (yes/partial/no) (yes)
- Proof sketches or intuitions are given for complex and/or novel results. (yes/partial/no) (yes)
- Appropriate citations to theoretical tools used are given. (yes/partial/no) (yes)
- All theoretical claims are demonstrated empirically to hold. (yes/partial/no/NA) (yes)
- All experimental code used to eliminate or disprove claims is included. (yes/no/NA) (yes)

Does this paper rely on one or more datasets? (yes/no) (yes)
If yes, please complete the list below.

- A motivation is given for why the experiments are conducted on the selected datasets. (yes/partial/no/NA) (yes)
- All novel datasets introduced in this paper are included in a data appendix. (yes/partial/no/NA) (yes)
- All novel datasets introduced in this paper will be made publicly available upon publication of the paper with a license that allows free usage for research purposes. (yes/-partial/no/NA) (yes)
- All datasets drawn from the existing literature (potentially including authors' own previously published work) are accompanied by appropriate citations. (yes/no/NA) (yes)
- All datasets drawn from the existing literature (potentially including authors' own previously published work) are publicly available. (yes/partial/no/NA) (yes)
- All datasets that are not publicly available are described in detail, with an explanation of why publicly available alternatives are not scientifically satisfying. (yes/-partial/no/NA) (yes)

Does this paper include computational experiments? (yes/no) (yes)
If yes, please complete the list below.

- Any code required for pre-processing data is included in the appendix. (yes/partial/no) (yes)
- All source code required for conducting and analyzing the experiments is included in a code appendix. (yes/partial/no) (yes)
- All source code required for conducting and analyzing the experiments will be made publicly available upon publication of the paper with a license that allows free usage for research purposes. (yes/partial/no) (yes)
- All source code implementing new methods has comments detailing the implementation, with references to the paper where each step comes from. (yes/partial/no) (yes)
- If an algorithm depends on randomness, then the method used for setting seeds is described in a way sufficient to allow replication of results. (yes/partial/no/NA) (yes)
- This paper specifies the computing infrastructure used for running experiments (hardware and software), including GPU/CPU models; amount of memory; operating system; names and versions of relevant software libraries and frameworks. (yes/partial/no) (yes)
- This paper formally describes evaluation metrics used and explains the motivation for choosing these metrics. (yes/partial/no) (yes)
- This paper states the number of algorithm runs used to compute each reported result. (yes/no) (yes)
- Analysis of experiments goes beyond single-dimensional summaries of performance (e.g., average; median) to include measures of variation, confidence, or other distributional information. (yes/no) (yes)
- The significance of any improvement or decrease in performance is judged using appropriate statistical tests (e.g., Wilcoxon signed-rank). (yes/partial/no) (yes)
- This paper lists all final (hyper-)parameters used for each model/algorithm in the paper's experiments. (yes/partial/no/NA) (yes)
- This paper states the number and range of values tried per (hyper-)parameter during development of the paper, along with the criterion used for selecting the final parameter setting. (yes/partial/no/NA) (yes)