Visual Saliency Transformer

Nian Liu[1*] Ni Zhang[2*] Kaiyuan Wan[2] Ling Shao[1] Junwei Han[2†]
[1]Inception Institute of Artificial Intelligence [2]Northwestern Polytechnical University
{liunian228, nnizhang.1995, kaiyuan.wan0106, junweihan2010}@gmail.com, ling.shao@ieee.org

Abstract

Existing state-of-the-art saliency detection methods heavily rely on CNN-based architectures. Alternatively, we rethink this task from a convolution-free sequence-to-sequence perspective and predict saliency by modeling long-range dependencies, which can not be achieved by convolution. Specifically, we develop a novel unified model based on a pure transformer, namely, Visual Saliency Transformer (VST), for both RGB and RGB-D salient object detection (SOD). It takes image patches as inputs and leverages the transformer to propagate global contexts among image patches. Unlike conventional architectures used in Vision Transformer (ViT), we leverage multi-level token fusion and propose a new token upsampling method under the transformer framework to get high-resolution detection results. We also develop a token-based multi-task decoder to simultaneously perform saliency and boundary detection by introducing task-related tokens and a novel patch-task-attention mechanism. Experimental results show that our model outperforms existing methods on both RGB and RGB-D SOD benchmark datasets. Most importantly, our whole framework not only provides a new perspective for the SOD field but also shows a new paradigm for transformer-based dense prediction models. Code is available at https://github.com/nnizhang/VST.

1. Introduction

SOD aims to detect objects that attract peoples' eyes and can help many vision tasks, *e.g.*, [58, 19]. Recently, RGB-D SOD has also gained growing interest with the extra spatial structure information from the depth data. Current state-of-the-art SOD methods are dominated by convolutional architectures [28], on both RGB and RGB-D data. They often adopt an encoder-decoder CNN architecture [47, 57], where the encoder encodes the input image to multi-level features and the decoder integrates the extracted features to predict the final saliency map. Based on this simple architecture, most efforts have been made to build a powerful decoder for predicting better saliency results. To this end, they introduced various attention models [37, 80, 7], multi-scale feature integration methods [24, 49, 16, 43], and multi-task learning frameworks [67, 77, 82, 69, 25]. An additional demand for RGB-D SOD is to effectively fuse cross-modal information, *i.e.*, the appearance information and the depth cues. Existing works propose various modality fusion methods, such as feature fusion [22, 4, 16, 18, 89], knowledge distillation [53], dynamic convolution [48], attention models [31, 78], and graph neural networks [43]. Hence, CNN-based methods have achieved impressive results [66, 88].

However, all previous methods are limited in learning global long-range dependencies. Global contexts [21, 83, 56, 44, 37] and global contrast [75, 2, 8] have been proved crucial for saliency detection for a long time. Nevertheless, due to the intrinsic limitation of CNNs that they extract features in local sliding windows, previous methods can hardly exploit the crucial global cues. Although some methods utilized fully connected layers [36, 22], global pooling layers [44, 37, 65], and non-local modules [38, 7] to incorporate the global context, they only did such in certain layers and the standard CNN-based architecture remains unchanged.

Recently, Transformer [61] was proposed to model global long-range dependencies among word sequences for machine translation. The core idea is the self-attention mechanism, which leverages the query-key correlation to relate different positions in a sequence. Transformer stacks the self-attention layers multiple times in both encoder and decoder, thus can model long-range dependencies in every layer. Hence, it is natural to introduce the Transformer to SOD, leveraging the global cues in the model all the way.

In this paper, for the first time, we rethink SOD from a new sequence-to-sequence perspective and develop a novel unified model for both RGB and RGB-D SOD based on a pure transformer, which is named Visual Saliency Transformer. We follow the recently proposed ViT models [12, 74] to divide each image into patches and adopt the Transformer model on the patch sequence. Then, the Transformer propagates long-range dependencies between image patches, without any need of using convolution. However,

*Equal contribution.
†Corresponding author.

it is not straightforward to apply ViT for SOD. On the one hand, how to perform dense prediction tasks based on pure transformer still remains an open question. On the other hand, ViT usually tokenizes the image to a very coarse scale. How to adapt ViT to the high-resolution prediction demand of SOD is also unclear.

To solve the first problem, we design a token-based transformer decoder by introducing task-related tokens to learn decision embeddings. Then, we propose a novel patch-task-attention mechanism to generate dense-prediction results, which provides a new paradigm for using transformer in dense prediction tasks. Motivated by previous SOD models [82, 87, 79, 25] that leveraged boundary detection to boost the SOD performance, we build a multi-task decoder to simultaneously conduct saliency and boundary detection by introducing a saliency token and a boundary token. This strategy simplifies the multitask prediction workflow by simply learning task-related tokens, thus largely reduces the computational costs while obtaining better results. To solve the second problem, inspired by the Tokens-to-Token (T2T) transformation [74], which reduces the length of tokens, we propose a new reverse T2T transformation to upsample tokens by expanding each token into multiple sub-tokens. Then, we upsample patch tokens progressively and fuse them with low-level tokens to obtain the final full-resolution saliency map. In addition, we also use a cross modality transformer to deeply explore the interaction between multi-modal information for RGB-D SOD. Finally, our VST outperforms existing state-of-the-art SOD methods with a comparable number of parameters and computational costs, on both RGB and RGB-D data.

Our main contributions can be summarized as follows:

- For the first time, we design a novel unified model based on the pure transformer architecture for both RGB and RGB-D SOD, from a new perspective of sequence-to-sequence modeling.
- We design a multi-task transformer decoder to jointly conduct saliency and boundary detection by introducing task-related tokens and patch-task-attention.
- We propose a new token upsampling method for transformer-based framework.
- Our proposed VST model achieves state-of-the-art results on both RGB and RGB-D SOD benchmark datasets, which demonstrates its effectiveness and the potential of transformer-based models for SOD.

2. Related Work

2.1. Deep Learning Based SOD

CNN-based approaches have become a mainstream trend in both RGB and RGB-D SOD and achieved promising performance. Most methods [24, 65, 49, 84, 16] leveraged a multi-level feature fusion strategy by using UNet [57] or HED-style [71] network structures. Some works introduced the attention mechanism to learn more discriminative features, including spatial and channel attention [52, 80, 16, 7] or pixel-wise contextual attention [37]. Other works [36, 64, 11, 42, 6] tried to design recurrent networks to refine the saliency map step-by-step. In addition, some works introduced multi-task learning, *e.g.*, fixation prediction [67], image caption [77], and edge detection [54, 82, 69, 79, 25] to boost the SOD performance.

As for RGB-D SOD, many methods have designed various models to fuse RGB and depth features and obtained significant results. Some models [4, 5, 18] adopted simple feature fusion methods, *i.e.*, concatenation, summation, or multiplication. Some others [81, 30, 52, 31] leveraged the depth cues to generate spatial or channel attention to enhance the RGB features. Besides, dynamic convolution [48], graph neural networks [43], and knowledge distillation [53] were also adopted to implement multi-modal feature fusion. In addition, [38, 39, 7] adopted the cross-attention mechanism to propagate long-range cross-modal interactions between RGB and depth cues.

Different from previous CNN-based methods, we are the first to rethink SOD from a sequence-to-sequence perspective and propose a unified model based on pure transformer for both RGB and RGB-D SOD. In our model, we follow [54, 82, 69, 79, 25] to leverage boundary detection to boost the SOD performance. However, different from these CNN-based models, we design a novel token-based multitask decoder to achieve this goal under the transformer framework.

2.2. Transformers in Computer Vision

Vaswani *et al*. [61] first proposed a transformer encoder-decoder architecture for machine translation, where multi-head self-attention and point-wise feed-forward layers are stacked multiple times. Recently, more and more works have introduced the Transformer model to various computer vision tasks and achieved excellent results. Some works combined CNNs and transformers into hybrid architectures for object detection [3, 91], panoptic segmentation [62], lane shape prediction [40], and so on. Typically, they first use CNNs to extract image features and then leverage the Transformer to incorporate long-range dependencies.

Other works design pure transformer models to process images from the sequence-to-sequence perspective. ViT [12] divided each image into a sequence of flattened 2D patches and then adopted the Transformer for image classification. Touvron *et al*. [60] introduced a teacher-student strategy to improve the data-efficiency of ViT and Wang *et al*. [68] proposed a pyramid architecture to adapt ViT for dense prediction tasks. T2T-ViT [74] adopted the T2T module to model local structures, thus generating multiscale token features. In this work, we adopt T2T-ViT as the backbone and propose a novel multitask decoder and a reverse

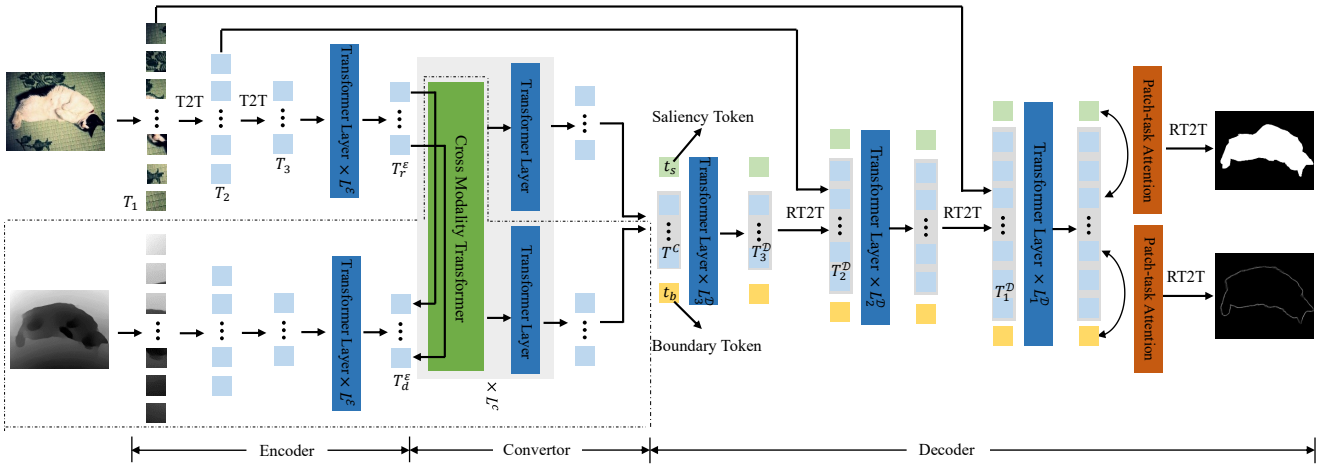

Figure 1. Overall architecture of our proposed VST model for both RGB and RGB-D SOD. It first uses an encoder to generate multi-level tokens from the input image patch sequence. Then, a convertor is adopted to convert the patch tokens to the decoder space, and also performs cross-modal information fusion for RGB-D data. Finally, a decoder simultaneously predicts the saliency map and the boundary map via the proposed task-related tokens and the patch-task-attention mechanism. An RT2T transformation is also proposed to progressively upsample patch tokens. The dotted line represents exclusive components for RGB-D SOD.

T2T token upsampling method. It is noteworthy that our usage of task-related tokens is different from previous models. In [12, 60], the class token is directly used for image classification via adopting a multilayer perceptron on the token embedding. However, we can not obtain dense prediction results directly from a single task token. Thus, we propose to perform patch-task-attention between patch tokens and the task tokens to predict saliency and boundary maps. We believe our strategy will also inspire future transformer models for other dense prediction tasks.

Another related work to ours is [86], which introduces transformer into the semantic segmentation task. The authors adopted a vision transformer as a backbone and then reshaped the token sequences to 2D image features. Then, they predicted full-resolution segmentation maps using convolution and bilinear upsampling. Their model still falls into the hybrid architecture category. In contrast, our model is a pure transformer architecture and does not rely on any convolution operation and bilinear upsampling.

3. Visual Saliency Transformer

Figure 1 shows the overall architecture of our proposed VST model. The main components include a transformer encoder based on T2T-ViT, a transformer convertor to convert patch tokens from the encoder space to the decoder space, and a multi-task transformer decoder.

3.1. Transformer Encoder

Similar to other CNN-based SOD methods, which often utilize pretrained image classification models such as VGG [59] and ResNet [23] as the backbone of their encoders to extract image features, we adopt the pretrained T2T-ViT [74] model as our backbone, as detailed below.

3.1.1 Tokens to Token

Given a sequence of patch tokens $\boldsymbol{T}'$ with length l from the previous layer, T2T-ViT iteratively applies the T2T module, which is composed of a re-structurization step and a soft split step, to model the local structure information in $\boldsymbol{T}'$ and obtain a new sequence of tokens.

Re-structurization. As shown in Figure 2(a), the tokens $\boldsymbol{T}'$ is first transformed using a transformer layer to obtain new tokens $\boldsymbol{T} \in \mathbb{R}^{l \times c}$:

$$\boldsymbol{T} = \text{MLP}(\text{MSA}(\boldsymbol{T}')), \tag{1}$$

where MSA and MLP denote the multi-head self-attention and multilayer perceptron in the original Transformer [61], respectively. Note that layer normalization [1] is applied before each block. Then, $\boldsymbol{T}$ is reshaped to a 2D image $\boldsymbol{I} \in \mathbb{R}^{h \times w \times c}$, where $l = h \times w$, to recover spatial structures, as shown in Figure 2(a).

Soft split. After the re-structurization step, $\boldsymbol{I}$ is first split into $k \times k$ patches with s overlapping. p zero-padding is also utilized to pad image boundaries. Then, the image patches are unfolded to a sequence of tokens $\boldsymbol{T}_o \in \mathbb{R}^{l_o \times ck^2}$, where the sequence length l_o is computed as:

$$l_o = h_o \times w_o = \lfloor \frac{h + 2p - k}{k - s} + 1 \rfloor \times \lfloor \frac{w + 2p - k}{k - s} + 1 \rfloor. \tag{2}$$

Different from ViT [12], the overlapped patch splitting adopted in T2T-ViT introduces local correspondence within neighbouring patches, thus bringing spatial priors.

The T2T transformation can be conducted iteratively multiple times. In each time, the re-structurization step first transforms previous token embeddings to new embeddings

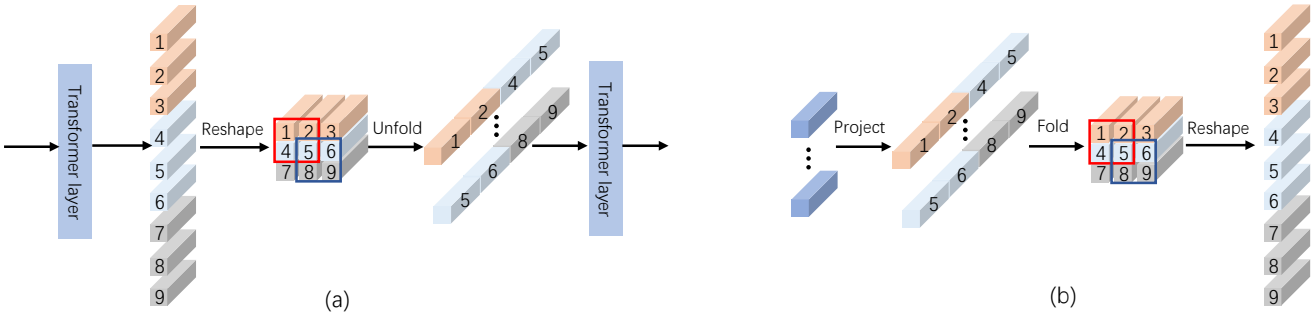

Figure 2. (a) T2T module merges neighbouring tokens into a new token, thus reducing the length of tokens. (b) Our proposed reverse T2T module upsamples tokens by expanding each token into multiple sub-tokens.

and also integrates long-range dependencies within all tokens. Then, the soft split operation aggregates the tokens in each $k \times k$ neighbour into a new token, which is ready to use for the next layer. Furthermore, when setting $s < k-1$, the length of tokens can be reduced progressively.

We follow [74] to first soft split the input image into patches and then adopt the T2T module twice. Among the three soft split steps, the patch sizes are set to $k = [7, 3, 3]$, the overlappings are set to $s = [3, 1, 1]$, and the padding sizes are set to $p = [2, 1, 1]$. As such, we can obtain multi-level tokens $\boldsymbol{T}_1 \in \mathbb{R}^{l_1 \times c}$, $\boldsymbol{T}_2 \in \mathbb{R}^{l_2 \times c}$, and $\boldsymbol{T}_3 \in \mathbb{R}^{l_3 \times c}$. Given the width and height of the input image as H and W, respectively, then $l_1 = \frac{H}{4} \times \frac{W}{4}$, $l_2 = \frac{H}{8} \times \frac{W}{8}$, and $l_3 = \frac{H}{16} \times \frac{W}{16}$. We follow [74] to set $c = 64$ and use a linear projection layer on $\boldsymbol{T}_3$ to transform its embedding dimension from c to $d = 384$.

3.1.2 Encoder with T2T-ViT Backbone

The final token sequence $\boldsymbol{T}_3$ is added with the sinusoidal position embedding [61] to encode 2D position information. Then, $L^{\mathcal{E}}$ transformer layers are used to model long-range dependencies among $\boldsymbol{T}_3$ to extract powerful patch token embeddings $\boldsymbol{T}^{\mathcal{E}} \in \mathbb{R}^{l_3 \times d}$.

For RGB SOD, we adopt a single transformer encoder to obtain RGB encoder patch tokens $\boldsymbol{T}_r^{\mathcal{E}} \in \mathbb{R}^{l_3 \times d}$ from each input RGB image. For RGB-D SOD, we follow two-stream architectures to further use another transformer encoder to extract the depth encoder patch tokens $\boldsymbol{T}_d^{\mathcal{E}}$ from the input depth map in a similar way, as shown in Figure 1.

3.2. Transformer Convertor

We insert a convertor module between the transformer encoder and decoder to convert the encoder patch tokens $\boldsymbol{T}_*^{\mathcal{E}}$ from the encoder space to the decoder space, thus obtaining the converted patch tokens $\boldsymbol{T}^{\mathcal{C}} \in \mathbb{R}^{l_3 \times d}$.

3.2.1 RGB-D Convertor

We fuse $\boldsymbol{T}_r^{\mathcal{E}}$ and $\boldsymbol{T}_d^{\mathcal{E}}$ in the RGB-D converter to integrate the complementary information between the RGB and depth data. To this end, we design a Cross Modality Transformer (CMT), which consists of $L^{\mathcal{C}}$ alternating cross-modality-attention layers and self-attention layers.

Cross-modality-attention. Under the pure transformer architecture, we modify the standard self-attention layer to propagate long-range cross-modal dependencies between the image and depth data, thus obtaining the cross-modality-attention, which is detailed as follows.

First, similar with the self-attention in [61], $\boldsymbol{T}_r^{\mathcal{E}}$ is embedded to queries $\boldsymbol{Q}_r \in \mathbb{R}^{l_3 \times d}$, keys $\boldsymbol{K}_r \in \mathbb{R}^{l_3 \times d}$, and values $\boldsymbol{V}_r \in \mathbb{R}^{l_3 \times d}$ through three linear projections. Similarly, we can obtain the depth queries $\boldsymbol{Q}_d$, keys $\boldsymbol{K}_d$, and values $\boldsymbol{V}_d$ from $\boldsymbol{T}_d^{\mathcal{E}}$.

Next, we compute the "Scaled Dot-Product Attention" [61] between the queries from one modality with the keys from the other modality. Then, the output is computed as a weighted sum of the values, formulated as:

$$\begin{aligned} \text{Attention}(\boldsymbol{Q}_r, \boldsymbol{K}_d, \boldsymbol{V}_d) &= \text{softmax}(\boldsymbol{Q}_r \boldsymbol{K}_d^\top / \sqrt{d}) \boldsymbol{V}_d, \\ \text{Attention}(\boldsymbol{Q}_d, \boldsymbol{K}_r, \boldsymbol{V}_r) &= \text{softmax}(\boldsymbol{Q}_d \boldsymbol{K}_r^\top / \sqrt{d}) \boldsymbol{V}_r. \end{aligned} \tag{3}$$

We follow the standard Transformer architecture in [61] and adopt the multi-head attention mechanism in the cross-modality-attention. The same positionwise feed-forward network, residual connections, and layer normalization [1] are also used, forming our CMT layer.

After each adoption of the proposed CMT layer, we use one standard transformer layer on each RGB and depth patch token sequence, further enhancing their token embeddings. After alternately using CMT and transformer for $L^{\mathcal{C}}$ times, we fuse the obtained RGB tokens and depth tokens by concatenation and then project them to the final converted tokens $\boldsymbol{T}^{\mathcal{C}}$, as shown in Figure 1.

3.2.2 RGB Convertor

To align with our RGB-D SOD model, for RGB SOD, we simply use $L^{\mathcal{C}}$ standard transformer layers on $\boldsymbol{T}_r^{\mathcal{E}}$ to obtain the converted patch token sequence $\boldsymbol{T}^{\mathcal{C}}$.

3.3. Multi-task Transformer Decoder

Our decoder aims to decode the patch tokens $\boldsymbol{T}^{\mathcal{C}}$ to saliency maps. Hence, we propose a novel token upsam-

pling method with multi-level token fusion and a token-based multi-task decoder.

3.3.1 Token Upsampling and Multi-level Token Fusion

We argue that directly predicting saliency maps from $\boldsymbol{T}^{\mathcal{C}}$ can not obtain high-quality results since the length of $\boldsymbol{T}^{\mathcal{C}}$ is relatively small, *i.e.*, $l_3 = \frac{H}{16} \times \frac{W}{16}$, which is limited for dense prediction. Thus, we propose to upsample patch tokens first and then conduct dense prediction. Most CNN-based methods [84, 82, 38, 18] adopt bilinear upsampling to recover large scale feature maps. Alternatively, we propose a new token upsampling method under the transformer framework. Inspired by the T2T module [74] that aggregates neighbour tokens to reduce the length of tokens progressively, we propose a reverse T2T (RT2T) transformation to upsample tokens by expanding each token into multiple sub-tokens, as shown in Figure 2(b).

Specifically, we first project the input patch tokens to reduce their embedding dimension from $d = 384$ to $c = 64$. Then, we use another linear projection to expand the embedding dimension from c to ck^2. Next, similar to the soft split step in T2T, each token is seen as a $k \times k$ image patch and neighbouring patches have s overlapping. Then, we can fold the tokens as an image using p zero-padding. The output image size can be computed using (2) reversely, *i.e.*, given the length of the input patch tokens as $h_o \times w_o$, the spatial size of the out image is $h \times w$. Finally, we reshape the image back to the upsampled tokens with size $l_o \times c$, where $l_o = h \times w$. By setting $s < k - 1$, the RT2T transformation can increase the length of the tokens. Motivated by T2T-ViT, we use RT2T three times and set $k = [3, 3, 7]$, $s = [1, 1, 3]$, and $p = [1, 1, 3]$. Thus, the length of the patch tokens can be gradually upsampled to $H \times W$, equaling to the original size of the input image.

Furthermore, motivated by the widely proved successes of multi-level feature fusion in existing SOD methods [24, 49, 84, 16, 43], we leverage low-level tokens with larger lengths from the T2T-ViT encoder, *i.e.*, $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$, to provide accurate local structural information. For both RGB and RGB-D SOD, we only use the low-level tokens from the RGB transformer encoder. Concretely, we progressively fuse $\boldsymbol{T}_2$ and $\boldsymbol{T}_1$ with the upsampled patch tokens via concatenation and linear projection. Then, we adopt one transformer layer to obtain the decoder tokens $\boldsymbol{T}_i^{\mathcal{D}}$ at each level i, where $i = 2, 1$. The whole process is formulated as:

$$\boldsymbol{T}_i^{\mathcal{D}} = \text{MLP}(\text{MSA}(\text{Linear}([\text{RT2T}(\boldsymbol{T}_{i+1}^{\mathcal{D}}), \boldsymbol{T}_i])), \quad (4)$$

where $[,]$ means concatenation along the token embedding dimension. "Linear" means linear projection to reduce the embedding dimension after the concatenation to c. Finally, we use another linear projection to recover the embedding dimension of $\boldsymbol{T}_i^{\mathcal{D}}$ back to d.

3.3.2 Token Based Multi-task Prediction

Inspired by existing pure transformer methods [74, 12], which add a class token on the patch token sequence for image classification, we also leverage task-related tokens to predict results. However, we can not obtain dense prediction results by directly using MLP on the task token embedding, as done in [74, 12]. Hence, we propose to perform patch-task-attention between the patch tokens and the task-related token to perform SOD.

In addition, motivated by the widely used boundary detection in SOD models [82, 69, 79, 25], we also adopt the multi-task learning strategy to jointly perform saliency and boundary detection, thus using the latter to help boost the performance of the former.

To this end, we design two task-related tokens, *i.e.*, a saliency token $\boldsymbol{t}_s \in \mathbb{R}^{1\times d}$ and a boundary token $\boldsymbol{t}_b \in \mathbb{R}^{1\times d}$. At each decoder level i, we add the saliency and boundary tokens $\boldsymbol{t}_s$ and $\boldsymbol{t}_b$ on the patch token sequence $\boldsymbol{T}_i^{\mathcal{D}}$, and then process them using $L_i^{\mathcal{D}}$ transformer layers. As such, the two task tokens can learn image-dependent task-related embeddings from the interaction with the patch tokens. After this, we take the updated patch tokens as input and perform the token upsampling and multi-level fusion process in (4) to obtain upsampled patch tokens $\boldsymbol{T}_{i-1}^{\mathcal{D}}$. Next, we reuse the updated $\boldsymbol{t}_s$ and $\boldsymbol{t}_b$ in the next level $i - 1$ to further update them and $\boldsymbol{T}_{i-1}^{\mathcal{D}}$. We repeat this process until we reach the last decoder level with the $\frac{1}{4}$ scale.

For saliency and boundary prediction, we perform patch-task-attention between the final decoder patch tokens $\boldsymbol{T}_1^{\mathcal{D}}$ and the saliency and boundary tokens $\boldsymbol{t}_s$ and $\boldsymbol{t}_b$. For saliency prediction, we first embed $\boldsymbol{T}_1^{\mathcal{D}}$ to queries $\boldsymbol{Q}_s^{\mathcal{D}} \in \mathbb{R}^{l_1\times d}$ and embed $\boldsymbol{t}_s$ to a key $\boldsymbol{K}_s \in \mathbb{R}^{1\times d}$ and a value $\boldsymbol{V}_s \in \mathbb{R}^{1\times d}$. Similarly, for boundary prediction, we embed $\boldsymbol{T}_1^{\mathcal{D}}$ to $\boldsymbol{Q}_b^{\mathcal{D}}$ and embed $\boldsymbol{t}_b$ to $\boldsymbol{K}_b$ and $\boldsymbol{V}_b$. Then, we adopt the patch-task-attention to obtain the task-related patch tokens:

$$\begin{aligned} \boldsymbol{T}_s^{\mathcal{D}} &= \text{sigmoid}(\boldsymbol{Q}_s^{\mathcal{D}} \boldsymbol{K}_s^{\top} / \sqrt{d}) \boldsymbol{V}_s + \boldsymbol{T}_1^{\mathcal{D}}, \\ \boldsymbol{T}_b^{\mathcal{D}} &= \text{sigmoid}(\boldsymbol{Q}_b^{\mathcal{D}} \boldsymbol{K}_b^{\top} / \sqrt{d}) \boldsymbol{V}_b + \boldsymbol{T}_1^{\mathcal{D}}. \end{aligned} \quad (5)$$

Here we use the sigmoid activation for the attention computation since in each equation we only have one key.

Since $\boldsymbol{T}_s^{\mathcal{D}}$ and $\boldsymbol{T}_b^{\mathcal{D}}$ are at the $\frac{1}{4}$ scale, we adopt the third RT2T transformation to upsample them to the full resolution. Finally, we apply two linear transformations with the sigmoid activation to project them to scalars in $[0, 1]$, and then reshape them to a 2D saliency map and a 2D boundary map, respectively. The whole process is given in Figure 1.

4. Experiments

4.1. Datasets and Evaluation Metrics

For RGB SOD, we evaluate our VST model on six widely used benchmark datasets, including **ECSSD** [72]

Table 1. Ablation studies of our proposed model. "Bili" denotes bilinear upsampling. "F" means multi-level token fusion. "TMD" denotes our proposed token-based multi-task decoder, while "C2D" means using conventional two-stream decoder to perform saliency and boundary detection without using task-related tokens. The best results are labeled in **blue**.

Settings	NJUD [26]				DUTLF-Depth [52]				STERE [46]				LFSD [33]			
	$S_m \uparrow$	maxF $\uparrow$	$E_\xi^{\max} \uparrow$	MAE $\downarrow$	$S_m \uparrow$	maxF $\uparrow$	$E_\xi^{\max} \uparrow$	MAE $\downarrow$	$S_m \uparrow$	maxF $\uparrow$	$E_\xi^{\max} \uparrow$	MAE $\downarrow$	$S_m \uparrow$	maxF $\uparrow$	$E_\xi^{\max} \uparrow$	MAE $\downarrow$
Baseline	0.869	0.862	0.931	0.073	0.889	0.887	0.942	0.062	0.868	0.853	0.927	0.075	0.842	0.845	0.893	0.103
+CMT	0.873	0.867	0.934	0.072	0.889	0.890	0.942	0.063	0.869	0.854	0.928	0.075	0.849	0.855	0.900	0.100
+CMT+Bili	0.906	0.902	0.944	0.045	0.926	0.930	0.961	0.032	0.889	0.877	0.939	0.051	0.856	0.858	0.895	0.081
+CMT+RT2T	0.915	0.915	0.951	0.039	0.934	0.940	0.964	0.028	0.896	0.889	0.943	0.046	0.867	0.873	0.903	0.073
+CMT+RT2T+F	**0.923**	**0.923**	**0.954**	**0.035**	0.936	0.943	0.963	0.028	0.910	0.903	0.947	0.040	0.876	0.880	0.909	0.067
+CMT+RT2T+F+TMD	0.922	0.920	0.951	**0.035**	**0.943**	**0.948**	**0.969**	**0.024**	**0.913**	**0.907**	**0.951**	**0.038**	**0.882**	**0.889**	**0.921**	**0.061**
+CMT+RT2T+F+C2D	0.922	0.921	**0.954**	0.036	0.941	0.947	0.968	0.026	0.911	0.906	0.949	0.040	0.874	0.878	0.909	0.069

(1,000 images), **HKU-IS** [32] (4,447 images), **PASCAL-S** [34] (850 images), **DUT-O** [73] (5,168 images), **SOD** [45] (300 images), and **DUTS** [63] (10,553 training images and 5,019 testing images). For RGB-D SOD, we use nine widely used benchmark datasets: **STERE** [46] (1,000 image pairs), **LFSD** [33] (100 image pairs), **RGBD135** [9] (135 image pairs), **SSD** [90] (80 image pairs), **NJUD** [26] (1,985 image pairs), **NLPR** [51] (1,000 image pairs), **DUTLF-Depth** [52] (1,200 image pairs), **SIP** [15] (929 image pairs), and **ReDWeb-S** [39] (3,179 image pairs).

We adopt four widely used evaluation metrics to evaluate our model performance comprehensively. Specifically, Structure-measure S_m [13] evaluates region-aware and object-aware structural similarity. Maximum F-measure (maxF) jointly considers precision and recall under the optimal threshold. Maximum enhanced-alignment measure $E_\xi^{\max}$ [14] simultaneously considers pixel-level errors and image-level errors. Mean Absolute Error (MAE) computes pixel-wise average absolute error. To evaluate the model complexity, we also report the multiply accumulate operations (MACs) and the number of parameters (Params).

4.2. Implementation Details

For fair comparisons, we follow most previous methods to use the training set of DUTS to train our VST for RGB SOD and use 1,485 images from NJUD, 700 images from NLPR, and 800 images from DUTLF-Depth to train our VST for RGB-D SOD. We follow [82] to use a sober operator to generate the boundary ground truth from GT saliency maps. For depth data preprocessing, we normalize the depth maps to [0,1] and duplicate them to three channels. Finally, we resize each image or depth map to 256×256 pixels and then randomly crop 224×224 image regions as the model input and use random flipping as data augmentation.

We use the pre-trained T2T-ViT$_t$-14 [74] model as our backbone since it has similar computational complexity as ResNet50 [23] does. This model uses the efficient Performer [10] and $c = 64$ in T2T modules, and sets $L^{\mathcal{E}} = 14$. In our convertor and decoder, we set $L^{\mathcal{C}} = L_3^{\mathcal{D}} = 4$ and $L_2^{\mathcal{D}} = L_1^{\mathcal{D}} = 2$ according to experimental results. We set the batchsizes as 11 and 8, and the total training steps as 40,000 and 60,000, for RGB and RGB-D SOD, respectively. For both of them, Adam [27] is adopted as the optimizer and the binary cross entropy loss is used for both saliency and boundary prediction. The initial learning rate is set to 0.0001 and reduced by a factor of 10 at half and three-quarters of the total step, respectively. Deep supervision is also used to facilitate the model training, where we use the patch-task attention to predict saliency and boundary at each decoder level. We implemented our model using Pytorch [50] and trained it on a GTX 1080 Ti GPU.

4.3. Ablation Study

Since our RGB-D VST is built by adding one more transformer encoder and additional CMT based on our RGB VST, while the other parts of the two models are the same, we conduct ablation studies based on our RGB-D VST to verify all of our proposed model components. The experimental results on four RGB-D SOD datasets, *i.e.*, NJUD, DUTLF-Depth, STERE, and LFSD, are given in Table 1. We remove the transformer convertor and the decoder from our RGB-D VST as the baseline model. Specifically, it uses the two-stream transformer encoder to extract RGB encoder patch tokens $\boldsymbol{T}_r^{\mathcal{E}}$ and the depth encoder patch tokens $\boldsymbol{T}_d^{\mathcal{E}}$, and then directly concatenate them and predict the saliency map with 1/16 scale by using MLP on each patch token.

Effectiveness of CMT. For cross-modal information fusion, we deploy our proposed CMT right after the transformer encoder to substitute the concatenation fusion method in the baseline model, shown as "+CMT" in Table 1. Compared to the baseline, CMT brings performance gain especially on the NJUD and LFSD datasets, hence demonstrating its effectiveness.

Effectiveness of RT2T. Based on "+CMT" model, we further simply use bilinear upsampling ("+CMT+Bili") to progressively upsample tokens to the full resolution and then predict the saliency map. The results show using bilinear upsampling to increase the resolution of the saliency map can largely improve the model performance. Then, we replace bilinear upsampling with our proposed RT2T token upsampling method ("+CMT+RT2T"). We find that RT2T leads to obvious performance improvement compared with using bilinear upsampling, which verifies its effectiveness.

Effectiveness of multi-level token fusion. We progressively fuse $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ in our decoder ("+CMT+RT2T+F") to

Table 2. Quantitative comparison of our proposed VST with other 14 SOTA RGB-D SOD methods on 9 benchmark datasets. **Red** and **blue** denote the best and the second-best results, respectively. '-' indicates the code or result is not available.

Dataset	Metric	A2dele [53]	JL-DCF [18]	SSF-RGBD [79]	UC-Net [76]	S^2MA [38]	PGAR [6]	DANet [85]	cmMS [29]	ATST [78]	CMW [31]	Cas-Gnn [43]	HDFNet [48]	CoNet [25]	BBS-Net [16]	VST
	MACs (G)	41.86	211.06	46.56	16.16	141.19	44.65	66.25	134.77	42.17	208.03	-	91.77	20.89	31.2	30.99
	Params (M)	30.34	143.52	32.93	31.26	86.65	16.2	26.68	92.02	32.17	85.65	-	44.15	43.66	49.77	83.83
NJUD [26]	$S_m\uparrow$	0.871	0.902	0.899	0.897	0.894	0.909	0.899	0.900	0.885	0.870	0.911	0.908	0.896	**0.921**	**0.922**
	maxF$\uparrow$	0.874	0.904	0.896	0.895	0.889	0.907	0.898	0.897	0.893	0.871	0.916	0.911	0.893	**0.919**	**0.920**
	$E_\xi^{\max}\uparrow$	0.916	0.944	0.935	0.936	0.930	0.940	0.935	0.936	0.930	0.927	0.948	0.944	0.937	**0.949**	**0.951**
	MAE$\downarrow$	0.051	0.041	0.043	0.043	0.054	0.042	0.046	0.044	0.047	0.061	**0.036**	0.039	0.046	**0.035**	**0.035**
NLPR [51]	$S_m\uparrow$	0.899	0.925	0.915	0.920	0.916	0.917	0.920	0.919	0.909	0.917	0.919	0.923	0.912	**0.931**	**0.932**
	maxF$\uparrow$	0.882	**0.918**	0.896	0.903	0.902	0.897	0.909	0.904	0.898	0.903	0.906	0.917	0.893	**0.918**	**0.920**
	$E_\xi^{\max}\uparrow$	0.944	**0.963**	0.953	0.956	0.953	0.950	0.955	0.955	0.951	0.951	0.955	**0.963**	0.948	0.961	**0.962**
	MAE$\downarrow$	0.029	**0.022**	0.027	0.025	0.030	0.027	0.027	0.028	0.027	0.029	0.025	**0.023**	0.027	0.023	0.024
DUTLF -Depth [52]	$S_m\uparrow$	0.885	0.906	0.915	0.871	0.904	0.899	0.899	0.912	0.916	0.797	0.920	0.908	**0.923**	0.882	**0.943**
	maxF$\uparrow$	0.891	0.910	0.923	0.864	0.899	0.898	0.904	0.913	0.928	0.779	0.926	0.915	**0.932**	0.870	**0.948**
	$E_\xi^{\max}\uparrow$	0.928	0.941	0.950	0.908	0.935	0.933	0.939	0.940	0.953	0.864	0.953	0.945	**0.959**	0.912	**0.969**
	MAE$\downarrow$	0.043	0.042	0.033	0.059	0.043	0.041	0.042	0.036	0.033	0.098	0.030	0.041	**0.029**	0.058	**0.024**
ReDWeb-S [39]	$S_m\uparrow$	0.641	**0.734**	0.595	0.713	0.711	0.656	-	0.699	0.679	0.634	-	0.728	0.696	0.693	**0.759**
	maxF$\uparrow$	0.603	**0.727**	0.558	0.710	0.696	0.632	-	0.677	0.673	0.607	-	0.717	0.693	0.680	**0.763**
	$E_\xi^{\max}\uparrow$	0.674	**0.805**	0.710	0.794	0.781	0.749	-	0.767	0.758	0.714	-	0.804	0.782	0.763	**0.826**
	MAE$\downarrow$	0.160	**0.128**	0.189	0.130	0.139	0.161	-	0.143	0.155	0.195	-	0.129	0.147	0.150	**0.113**
STERE [46]	$S_m\uparrow$	0.879	0.903	0.837	0.903	0.890	0.894	0.901	0.894	0.896	0.852	0.899	0.900	0.905	**0.908**	**0.913**
	maxF$\uparrow$	0.880	**0.904**	0.840	0.899	0.882	0.880	0.892	0.887	0.901	0.837	0.901	0.900	0.901	0.903	**0.907**
	$E_\xi^{\max}\uparrow$	0.928	**0.947**	0.912	0.944	0.932	0.929	0.937	0.935	0.942	0.907	0.944	0.943	**0.947**	0.942	**0.951**
	MAE$\downarrow$	0.045	0.040	0.065	0.039	0.051	0.045	0.044	0.045	**0.038**	0.067	0.039	0.042	**0.037**	0.041	**0.038**
SSD [90]	$S_m\uparrow$	0.803	0.860	0.790	0.865	0.868	0.832	0.864	0.857	0.850	0.798	0.872	**0.879**	0.851	0.863	**0.889**
	maxF$\uparrow$	0.777	0.833	0.762	0.855	0.848	0.798	0.843	0.839	0.853	0.771	0.863	**0.870**	0.837	0.843	**0.876**
	$E_\xi^{\max}\uparrow$	0.862	0.902	0.867	0.907	0.909	0.872	0.914	0.900	0.920	0.871	0.923	**0.925**	0.917	0.914	**0.935**
	MAE$\downarrow$	0.070	0.053	0.084	0.049	0.053	0.068	0.050	0.053	0.052	0.085	0.047	**0.046**	0.056	0.052	**0.045**
RGBD135 [9]	$S_m\uparrow$	0.886	0.931	0.904	0.934	**0.941**	0.886	0.924	0.934	0.917	0.934	0.894	0.926	0.914	0.934	**0.943**
	maxF$\uparrow$	0.872	0.923	0.885	0.930	**0.935**	0.864	0.914	0.928	0.916	0.931	0.894	0.921	0.902	0.928	**0.940**
	$E_\xi^{\max}\uparrow$	0.921	0.968	0.940	**0.976**	0.973	0.924	0.966	0.969	0.961	0.969	0.937	0.970	0.948	0.966	**0.978**
	MAE$\downarrow$	0.029	0.021	0.026	0.019	0.021	0.032	0.023	**0.018**	0.022	0.022	0.028	0.022	0.024	0.021	**0.017**
LFSD [33]	$S_m\uparrow$	0.825	0.853	0.851	**0.856**	0.829	0.808	0.841	0.845	0.845	0.776	0.838	0.846	0.848	0.835	**0.882**
	maxF$\uparrow$	0.828	**0.863**	**0.863**	0.860	0.831	0.794	0.840	0.858	0.859	0.779	0.843	0.858	0.852	0.828	**0.889**
	$E_\xi^{\max}\uparrow$	0.866	0.894	0.892	**0.898**	0.865	0.853	0.874	0.886	0.893	0.834	0.880	0.889	0.895	0.870	**0.921**
	MAE$\downarrow$	0.084	0.077	**0.074**	**0.074**	0.102	0.099	0.087	0.082	0.078	0.130	0.081	0.085	0.076	0.092	**0.061**
SIP [15]	$S_m\uparrow$	0.829	0.880	0.799	0.875	0.872	0.838	0.875	0.872	0.849	0.705	-	**0.886**	0.860	0.879	**0.904**
	maxF$\uparrow$	0.834	0.889	0.786	0.879	0.877	0.827	0.876	0.876	0.861	0.677	-	**0.894**	0.873	0.884	**0.915**
	$E_\xi^{\max}\uparrow$	0.890	0.925	0.870	0.919	0.919	0.886	0.918	0.911	0.901	0.804	-	**0.930**	0.917	0.922	**0.944**
	MAE$\downarrow$	0.070	0.049	0.091	0.051	0.058	0.073	0.055	0.058	0.063	0.141	-	**0.048**	0.058	0.055	**0.040**

supply low-level fine-grained information. We find that this strategy further improves the model performance. Hence, leveraging low-level tokens in transformer is as important as fusing low-level features in CNN-based models.

Effectiveness of the multi-task transformer decoder. Based on "+CMT+RT2T+F", we further use our token-based multi-task decoder (TMD) to jointly perform saliency and boundary detection ("+CMT+RT2T+F+TMD"). It shows that using boundary detection can bring further performance gain for SOD on three out of four datasets. To very the effectiveness of our token-based prediction scheme, we try to directly use a conventional two-stream decoder (C2D) by using the "+RT2T+F" architecture twice to predict the saliency map and boundary map via MLP, without using task-related tokens. This model is denoted as "+CMT+RT2T+F+C2D" in Table 1. The parameters and MACs of TMD vs. C2D are 17.22 M vs. 20.35 M and 17.70 G vs. 28.27 G, respectively. The results show that using our TMD can achieve better results than using C2D on three out of four datasets, and also with much less computational costs. This clearly demonstrates the superiority of our proposed token-based transformer decoder.

4.4. Comparison with State-of-the-Art Methods

For RGB-D SOD, we compare our VST with 14 state-of-the-art RGB-D SOD methods, *i.e.*, A2dele [53], JL-DCF [18], SSF-RGBD [79], UC-Net [76], S^2MA [38], PGAR [6], DANet [85], cmMS [29], ATSA [78], CMW [31], Cas-Gnn [43], HDFNet [48], CoNet [25], and BBS-Net [16]. For RGB SOD, we compare our VST with 12 state-of-the-art RGB SOD models, including GateNet [84], CSF [20], LDF [69], MINet [49], ITSD [87], EGNet [82], TSPOANet [41], AFNet [17], PoolNet [35], CPD [70], BASNet [55], and PiCANet [37]. Table 2 and Table 3 show the quantitative comparison results for RGB-D and RGB SOD, respec-

Table 3. Quantitative comparison of our proposed VST with other 12 SOTA RGB SOD methods on 6 benchmark datasets. "-R" and "-R2" means the ResNet50 and Res2Net backbone, respectively.

Dataset	Metric	PiCANet [37]	BASNet [55]	CPD-R [70]	PoolNet [35]	AFNet [17]	TSPOANet [41]	EGNet-R [82]	ITSD-R [87]	MINet-R [49]	LDF-R [69]	CSF-R2 [20]	GateNet-R [84]	VST
	MACs (G)	54.05	127.36	17.77	88.89	21.66	-	157.21	15.96	87.11	15.51	18.96	162.13	23.16
	Params (M)	47.22	87.06	47.85	68.26	35.95	-	111.64	26.47	162.38	25.15	36.53	128.63	44.48
DUTS [63]	$S_m \uparrow$	0.863	0.866	0.869	0.879	0.867	0.860	0.887	0.885	0.884	**0.892**	0.890	0.891	**0.896**
	maxF↑	0.840	0.838	0.840	0.853	0.838	0.828	0.866	0.867	0.864	**0.877**	0.869	**0.874**	**0.877**
	$E_\xi^{\max} \uparrow$	0.915	0.902	0.913	0.917	0.910	0.907	0.926	0.929	0.926	0.930	0.929	**0.932**	**0.939**
	MAE↓	0.040	0.047	0.043	0.041	0.045	0.049	0.039	0.041	**0.037**	**0.034**	**0.037**	0.038	**0.037**
ECSSD [72]	$S_m \uparrow$	0.916	0.916	0.918	0.917	0.914	0.907	0.925	0.925	0.925	0.925	**0.931**	0.924	**0.932**
	maxF↑	0.929	0.931	0.926	0.929	0.924	0.919	0.936	0.939	0.938	0.938	**0.942**	0.935	**0.944**
	$E_\xi^{\max} \uparrow$	0.953	0.951	0.951	0.948	0.947	0.942	0.955	0.959	0.957	0.954	**0.960**	0.955	**0.964**
	MAE↓	0.035	0.037	0.037	0.042	0.042	0.047	0.037	0.035	**0.034**	**0.034**	**0.033**	0.038	**0.034**
HKU-IS [32]	$S_m \uparrow$	0.905	0.909	0.906	0.916	0.905	0.902	0.918	0.917	0.919	0.920	-	**0.921**	**0.928**
	maxF↑	0.913	0.919	0.911	0.920	0.910	0.909	0.923	0.926	0.926	**0.929**	-	0.926	**0.937**
	$E_\xi^{\max} \uparrow$	0.951	0.952	0.950	0.955	0.949	0.950	0.956	**0.960**	**0.960**	0.958	-	0.959	**0.968**
	MAE↓	0.031	0.032	0.034	0.032	0.036	0.039	0.031	0.031	**0.029**	**0.028**	-	0.031	0.030
PASCAL-S [34]	$S_m \uparrow$	0.846	0.837	0.847	0.852	0.849	0.841	0.852	0.861	0.856	0.861	**0.863**	**0.863**	**0.873**
	maxF↑	0.824	0.819	0.817	0.830	0.824	0.817	0.825	**0.839**	0.831	**0.839**	**0.839**	0.836	**0.850**
	$E_\xi^{\max} \uparrow$	0.882	0.868	0.872	0.880	0.877	0.871	0.874	**0.889**	0.883	0.888	0.885	0.886	**0.900**
	MAE↓	0.072	0.083	0.077	0.076	0.076	0.082	0.080	**0.071**	**0.071**	**0.067**	0.073	**0.071**	**0.067**
DUT-O [73]	$S_m \uparrow$	0.826	0.836	0.825	0.832	0.826	0.818	**0.841**	0.840	0.833	0.839	0.838	0.840	**0.850**
	maxF↑	0.767	0.779	0.754	0.769	0.759	0.750	0.778	**0.792**	0.769	0.782	0.775	0.782	**0.800**
	$E_\xi^{\max} \uparrow$	0.865	0.872	0.868	0.869	0.861	0.858	0.878	**0.880**	0.869	0.870	0.869	0.878	**0.888**
	MAE↓	0.054	0.057	0.056	0.056	0.057	0.062	**0.053**	0.061	0.056	**0.052**	0.055	0.055	0.058
SOD [45]	$S_m \uparrow$	0.813	0.799	0.797	0.823	0.811	0.802	0.824	**0.835**	0.830	0.831	0.826	0.827	**0.854**
	maxF↑	0.824	0.808	0.804	0.832	0.819	0.809	0.831	**0.849**	0.835	0.841	0.832	0.835	**0.866**
	$E_\xi^{\max} \uparrow$	0.871	0.846	0.860	0.873	0.867	0.852	0.875	**0.889**	0.878	0.878	0.883	0.877	**0.902**
	MAE↓	0.073	0.091	0.089	0.085	0.085	0.094	0.080	0.075	0.074	**0.071**	0.079	0.079	**0.065**

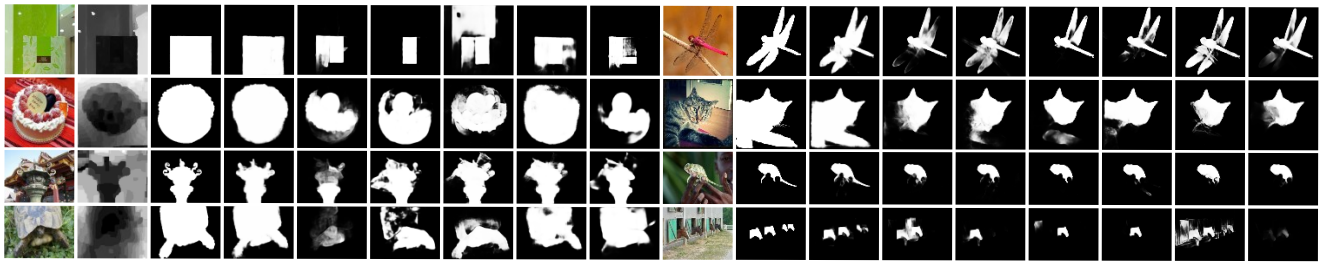

Figure 3. Qualitative comparison against state-of-the-art RGB-D (left) and RGB (right) SOD methods. (GT: ground truth)

tively. The results show that our VST outperforms all previous state-of-the-art CNN-based SOD models on both RGB and RGB-D benchmark datasets, with comparable number of parameters and relatively small MACs, hence demonstrating the great effectiveness of our VST. We also show visual comparison results among best-performed models in Figure 3. It shows our proposed VST can accurately detect salient objects in very challenging scenarios, *e.g.*, big salient objects, cluttered backgrounds, foreground and background having similar appearances, etc.

5. Conclusion

In this paper, we are the first to rethink SOD from a sequence-to-sequence perspective and develop a novel unified model based on a pure transformer, for both RGB and RGB-D SOD. To handle the difficulty of applying transformers in dense prediction tasks, we propose a new token upsampling method under the transformer framework and fuse multi-level patch tokens. We also design a multi-task decoder by introducing task-related tokens and a novel patch-task-attention mechanism to jointly perform saliency and boundary detection. Our VST model achieves state-of-the-art results for both RGB and RGB-D SOD without relying on heavy computational costs, thus showing its great effectiveness. We also set a new paradigm for the open question of how to use transformer in dense prediction tasks.

Acknowledgments: This work was supported in part by the National Key R&D Program of China under Grant 2020AAA0105702, the National Science Foundation of China under Grant 62027813, 62036005, U20B2065, U20B2068.

References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 3, 4

[2] Ali Borji and Laurent Itti. Exploiting local and global patch rarities for saliency detection. In *CVPR*, pages 478–485, 2012. 1

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229, 2020. 2

[4] Hao Chen and Youfu Li. Progressively complementarity-aware fusion network for rgb-d salient object detection. In *CVPR*, pages 3051–3060, 2018. 1, 2

[5] Hao Chen and Youfu Li. Three-stream attention-aware network for rgb-d salient object detection. *TIP*, 28(6):2825–2835, 2019. 2

[6] Shuhan Chen and Yun Fu. Progressively guided alternate refinement network for rgb-d salient object detection. In *ECCV*, pages 520–538, 2020. 2, 7, 14

[7] Zuyao Chen, Runmin Cong, Qianqian Xu, and Qingming Huang. Dpanet: Depth potentiality-aware gated attention network for rgb-d salient object detection. *TIP*, 2020. 1, 2

[8] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. Global contrast based salient region detection. *TPAMI*, 37(3):569–582, 2014. 1

[9] Yupeng Cheng, Huazhu Fu, Xingxing Wei, Jiangjian Xiao, and Xiaochun Cao. Depth enhanced saliency detection method. In *Conference on Internet Multimedia Computing and Service*, pages 23–27, 2014. 6, 7

[10] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. In *ICLR*, 2020. 6

[11] Zijun Deng, Xiaowei Hu, Lei Zhu, Xuemiao Xu, Jing Qin, Guoqiang Han, and Pheng-Ann Heng. R3net: Recurrent residual refinement network for saliency detection. In *IJCAI*, pages 684–690, 2018. 2

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020. 1, 2, 3, 5

[13] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *ICCV*, pages 4548–4557, 2017. 6

[14] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment Measure for Binary Foreground Map Evaluation. In *IJCAI*, pages 698–704, 2018. 6

[15] Deng-Ping Fan, Zheng Lin, Zhao Zhang, Menglong Zhu, and Ming-Ming Cheng. Rethinking rgb-d salient object detection: Models, data sets, and large-scale benchmarks. *TNNLS*, 32(5):2075–2089, 2020. 6, 7

[16] Deng-Ping Fan, Yingjie Zhai, Ali Borji, Jufeng Yang, and Ling Shao. Bbs-net: Rgb-d salient object detection with a bifurcated backbone strategy network. In *ECCV*, pages 275–292, 2020. 1, 2, 5, 7, 8, 14

[17] Mengyang Feng, Huchuan Lu, and Errui Ding. Attentive feedback network for boundary-aware salient object detection. In *CVPR*, pages 1623–1632, 2019. 7, 8, 13

[18] Keren Fu, Deng-Ping Fan, Ge-Peng Ji, and Qijun Zhao. Jl-dcf: Joint learning and densely-cooperative fusion framework for rgb-d salient object detection. In *CVPR*, pages 3052–3062, 2020. 1, 2, 5, 7, 8, 14

[19] Chuang Gan, Naiyan Wang, Yi Yang, Dit-Yan Yeung, and Alex G Hauptmann. Devnet: A deep event network for multimedia event detection and evidence recounting. In *CVPR*, pages 2568–2577, 2015. 1

[20] Shang-Hua Gao, Yong-Qiang Tan, Ming-Ming Cheng, Chengze Lu, Yunpeng Chen, and Shuicheng Yan. Highly efficient salient object detection with 100k parameters. In *ECCV*, pages 702–721, 2020. 7, 8, 13

[21] Stas Goferman, Lihi Zelnik-Manor, and Ayellet Tal. Context-aware saliency detection. *TPAMI*, 34(10):1915–1926, 2011. 1

[22] Junwei Han, Hao Chen, Nian Liu, Chenggang Yan, and Xuelong Li. Cnns-based rgb-d saliency detection via cross-view transfer and multiview fusion. *IEEE transactions on cybernetics*, 48(11):3171–3183, 2017. 1

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 3, 6

[24] Q Hou, MM Cheng, X Hu, A Borji, Z Tu, and PHS Torr. Deeply supervised salient object detection with short connections. *TPAMI*, 41(4):815–828, 2018. 1, 2, 5

[25] Wei Ji, Jingjing Li, Miao Zhang, Yongri Piao, and Huchuan Lu. Accurate rgb-d salient object detection via collaborative learning. In *ECCV*, pages 52–69, 2020. 1, 2, 5, 7, 8, 14

[26] Ran Ju, Ling Ge, Wenjing Geng, Tongwei Ren, and Gangshan Wu. Depth saliency based on anisotropic center-surround difference. In *ICIP*, pages 1115–1119, 2014. 6, 7, 13

[27] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[28] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 1

[29] Chongyi Li, Runmin Cong, Yongri Piao, Qianqian Xu, and Chen Change Loy. Rgb-d salient object detection with cross-modality modulation and selection. In *ECCV*, pages 225–241, 2020. 7, 14

[30] Gongyang Li, Zhi Liu, and Haibin Ling. Icnet: Information conversion network for rgb-d based salient object detection. *TIP*, 29:4873–4884, 2020. 2

[31] Gongyang Li, Zhi Liu, Linwei Ye, Yang Wang, and Haibin Ling. Cross-modal weighting network for rgb-d salient object detection. In *ECCV*, pages 665–681, 2020. 1, 2, 7, 14

[32] Guanbin Li and Yizhou Yu. Visual saliency based on multiscale deep features. In *CVPR*, pages 5455–5463, 2015. 6, 8, 13

[33] Nianyi Li, Jinwei Ye, Yu Ji, Haibin Ling, and Jingyi Yu. Saliency detection on light field. In *CVPR*, pages 2806–2813, 2014. 6, 7, 13

[34] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In *CVPR*, pages 280–287, 2014. 6, 8, 13

[35] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. A simple pooling-based design for real-time salient object detection. In *CVPR*, pages 3917–3926, 2019. 7, 8, 13

[36] Nian Liu and Junwei Han. Dhsnet: Deep hierarchical saliency network for salient object detection. In *CVPR*, pages 678–686, 2016. 1, 2

[37] Nian Liu, Junwei Han, and Ming-Hsuan Yang. Picanet: Learning pixel-wise contextual attention for saliency detection. In *CVPR*, pages 3089–3098, 2018. 1, 2, 7, 8, 13

[38] Nian Liu, Ni Zhang, and Junwei Han. Learning selective self-mutual attention for rgb-d saliency detection. In *CVPR*, pages 13756–13765, 2020. 1, 2, 5, 7, 14

[39] Nian Liu, Ni Zhang, Ling Shao, and Junwei Han. Learning selective mutual attention and contrast for rgb-d saliency detection. *arXiv preprint arXiv:2010.05537*, 2020. 2, 6, 7

[40] Ruijin Liu, Zejian Yuan, Tie Liu, and Zhiliang Xiong. End-to-end lane shape prediction with transformers. In *IEEE Winter Conference on Applications of Computer Vision*, pages 3694–3702, 2021. 2

[41] Yi Liu, Qiang Zhang, Dingwen Zhang, and Jungong Han. Employing deep part-object relationships for salient object detection. In *ICCV*, pages 1232–1241, 2019. 7, 8, 13

[42] Zhengyi Liu, Song Shi, Quntao Duan, Wei Zhang, and Peng Zhao. Salient object detection for rgb-d image by single stream recurrent convolution neural network. *Neurocomputing*, 363:46–57, 2019. 2

[43] Ao Luo, Xin Li, Fan Yang, Zhicheng Jiao, Hong Cheng, and Siwei Lyu. Cascade graph neural networks for rgb-d salient object detection. In *ECCV*, pages 346–364, 2020. 1, 2, 5, 7, 14

[44] Zhiming Luo, Akshaya Mishra, Andrew Achkar, Justin Eichel, Shaozi Li, and Pierre-Marc Jodoin. Non-local deep features for salient object detection. In *CVPR*, pages 6609–6617, 2017. 1

[45] Vida Movahedi and James H Elder. Design and perceptual validation of performance measures for salient object segmentation. In *CVPR Workshops*, pages 49–56, 2010. 6, 8, 13

[46] Yuzhen Niu, Yujie Geng, Xueqing Li, and Feng Liu. Leveraging stereopsis for saliency analysis. In *CVPR*, pages 454–461, 2012. 6, 7, 13

[47] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *ICCV*, pages 1520–1528, 2015. 1

[48] Youwei Pang, Lihe Zhang, Xiaoqi Zhao, and Huchuan Lu. Hierarchical dynamic filtering network for rgb-d salient object detection. In *ECCV*, pages 235–252, 2020. 1, 2, 7, 8, 14

[49] Youwei Pang, Xiaoqi Zhao, Lihe Zhang, and Huchuan Lu. Multi-scale interactive network for salient object detection. In *CVPR*, pages 9413–9422, 2020. 1, 2, 5, 7, 8, 13

[50] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NIPS*, 32:8026–8037, 2019. 6

[51] Houwen Peng, Bing Li, Weihua Xiong, Weiming Hu, and Rongrong Ji. Rgbd salient object detection: A benchmark and algorithms. In *ECCV*, pages 92–109, 2014. 6, 7

[52] Yongri Piao, Wei Ji, Jingjing Li, Miao Zhang, and Huchuan Lu. Depth-induced multi-scale recurrent attention network for saliency detection. In *ICCV*, pages 7254–7263, 2019. 2, 6, 7, 13

[53] Yongri Piao, Zhengkun Rong, Miao Zhang, Weisong Ren, and Huchuan Lu. A2dele: Adaptive and attentive depth distiller for efficient rgb-d salient object detection. In *CVPR*, pages 9060–9069, 2020. 1, 2, 7, 14

[54] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: Boundary-aware salient object detection. In *CVPR*, pages 7479–7489, 2019. 2

[55] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: Boundary-aware salient object detection. In *CVPR*, pages 7479–7489, 2019. 7, 8, 13

[56] Jianqiang Ren, Xiaojin Gong, Lu Yu, Wenhui Zhou, and Michael Ying Yang. Exploiting global priors for rgb-d saliency detection. In *CVPR workshops*, pages 25–32, 2015. 1

[57] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241, 2015. 1, 2

[58] Wataru Shimoda and Keiji Yanai. Distinct class-specific saliency maps for weakly supervised semantic segmentation. In *ECCV*, pages 218–234, 2016. 1

[59] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 3

[60] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, pages 10347–10357, 2021. 2, 3

[61] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017. 1, 2, 3, 4

[62] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In *CVPR*, pages 5463–5474, 2021. 2

[63] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *CVPR*, pages 136–145, 2017. 6, 8, 13

[64] Linzhao Wang, Lijun Wang, Huchuan Lu, Pingping Zhang, and Xiang Ruan. Salient object detection with recurrent fully convolutional networks. *TPAMI*, 41(7):1734–1746, 2018. 2

[65] Tiantian Wang, Ali Borji, Lihe Zhang, Pingping Zhang, and Huchuan Lu. A stagewise refinement model for detecting salient objects in images. In *ICCV*, pages 4019–4028, 2017. 1, 2

[66] Wenguan Wang, Qiuxia Lai, Huazhu Fu, Jianbing Shen, and Haibin Ling. Salient object detection in the deep learning era: An in-depth survey. *TPAMI*, 2021. 1

[67] Wenguan Wang, Jianbing Shen, Xingping Dong, and Ali Borji. Salient object detection driven by fixation prediction. In *CVPR*, pages 1711–1720, 2018. 1, 2

[68] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. *arXiv preprint arXiv:2102.12122*, 2021. 2

[69] Jun Wei, Shuhui Wang, Zhe Wu, Chi Su, Qingming Huang, and Qi Tian. Label decoupling framework for salient object detection. In *CVPR*, pages 13025–13034, 2020. 1, 2, 5, 7, 8, 13

[70] Zhe Wu, Li Su, and Qingming Huang. Cascaded partial decoder for fast and accurate salient object detection. In *CVPR*, pages 3907–3916, 2019. 7, 8, 13

[71] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *CVPR*, pages 1395–1403, 2015. 2

[72] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. Hierarchical saliency detection. In *CVPR*, pages 1155–1162, 2013. 5, 8

[73] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *CVPR*, pages 3166–3173, 2013. 6, 8

[74] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *ICCV*, 2021. 1, 2, 3, 4, 5, 6

[75] Yun Zhai and Mubarak Shah. Visual attention detection in video sequences using spatiotemporal cues. In *ACM International Conference on Multimedia*, pages 815–824, 2006. 1

[76] Jing Zhang, Deng-Ping Fan, Yuchao Dai, Saeed Anwar, Fatemeh Sadat Saleh, Tong Zhang, and Nick Barnes. Uc-net: Uncertainty inspired rgb-d saliency detection via conditional variational autoencoders. In *CVPR*, pages 8582–8591, 2020. 7, 8, 14

[77] Lu Zhang, Jianming Zhang, Zhe Lin, Huchuan Lu, and You He. Capsal: Leveraging captioning to boost semantics for salient object detection. In *CVPR*, pages 6024–6033, 2019. 1, 2

[78] Miao Zhang, Sun Xiao Fei, Jie Liu, Shuang Xu, Yongri Piao, and Huchuan Lu. Asymmetric two-stream architecture for accurate rgb-d saliency detection. In *ECCV*, pages 374–390, 2020. 1, 7, 14

[79] Miao Zhang, Weisong Ren, Yongri Piao, Zhengkun Rong, and Huchuan Lu. Select, supplement and focus for rgb-d saliency detection. In *CVPR*, pages 3472–3481, 2020. 2, 5, 7, 14

[80] Xiaoning Zhang, Tiantian Wang, Jinqing Qi, Huchuan Lu, and Gang Wang. Progressive attention guided recurrent network for salient object detection. In *CVPR*, pages 714–722, 2018. 1, 2

[81] Jia-Xing Zhao, Yang Cao, Deng-Ping Fan, Ming-Ming Cheng, Xuan-Yi Li, and Le Zhang. Contrast prior and fluid pyramid integration for rgbd salient object detection. In *CVPR*, pages 3927–3936, 2019. 2

[82] Jia-Xing Zhao, Jiang-Jiang Liu, Deng-Ping Fan, Yang Cao, Jufeng Yang, and Ming-Ming Cheng. Egnet:edge guidance network for salient object detection. In *ICCV*, pages 8779–8788, 2019. 1, 2, 5, 6, 7, 8, 13

[83] Rui Zhao, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Saliency detection by multi-context deep learning. In *CVPR*, pages 1265–1274, 2015. 1

[84] Xiaoqi Zhao, Youwei Pang, Lihe Zhang, Huchuan Lu, and Lei Zhang. Suppress and balance: A simple gated network for salient object detection. In *ECCV*, pages 35–51, 2020. 2, 5, 7, 8, 13

[85] Xiaoqi Zhao, Lihe Zhang, Youwei Pang, Huchuan Lu, and Lei Zhang. A single stream network for robust and real-time rgb-d salient object detection. In *ECCV*, pages 646–662, 2020. 7, 14

[86] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *CVPR*, pages 6881–6890, 2021. 3

[87] Huajun Zhou, Xiaohua Xie, Jian-Huang Lai, Zixuan Chen, and Lingxiao Yang. Interactive two-stream decoder for accurate and fast saliency detection. In *CVPR*, pages 9141–9150, 2020. 2, 7, 8, 13

[88] Tao Zhou, Deng-Ping Fan, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Rgb-d salient object detection: A survey. *Computational Visual Media*, pages 1–33, 2021. 1

[89] Tao Zhou, Huazhu Fu, Geng Chen, Yi Zhou, Deng-Ping Fan, and Ling Shao. Specificity-preserving rgb-d saliency detection. In *ICCV*, 2021. 1

[90] Chunbiao Zhu and Ge Li. A three-pathway psychobiological framework of salient object detection using stereoscopic technology. In *ICCV Workshops*, pages 3008–3014, 2017. 6, 7

[91] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2020. 2

6. Supplementary materials

6.1. Ablation Study on RGB SOD Datasets

We further report the results of ablation studies on four RGB SOD datasets, *i.e.*, DUTS, HKU-IS, PASCAL-S, and SOD, in Table 4 to demonstrate the effectiveness of our VST model components.

The baseline model is using transformer encoder to extract patch tokens $\boldsymbol{T}_r^{\mathcal{E}}$ and then directly using $\boldsymbol{T}_r^{\mathcal{E}}$ to predict the saliency map with 1/16 scale by using MLP on each patch token. Based on the baseline, we insert RGB convertor right after the transformer encoder, shown as "+RC" in Table 4. Compared to the baseline, RC brings performance gains especially on the DUTS and PASCAL-S datasets, which demonstrates its effectiveness. For other components, *i.e.*, RT2T, multi-level token fusion, and multi-task transformer decoder, we get consistent conclusions with the ablation studies on RGB-D SOD datasets as follows.

First, using bilinear upsampling ("+RC+Bili") can significantly improve the model performance while using our proposed RT2T ("+RC+RT2T") can further bring performance gains, hence demonstrating the effectiveness of our proposed RT2T. Second, based on "+RC+RT2T", multi-level token fusion ("+RC+RT2T+F") can lead to better performance on all four datasets, which verifies its effectiveness. Third, using multi-task transformer decoder ("+RC+RT2T+F+TMD") can improve the model performance on all four datasets and it is also superior to the conventional two-stream decoder ("+RC+RT2T+F+C2D").

To this end, the results of ablation studies on both RGB and RGB-D SOD datasets strongly demonstrate the effectiveness of our proposed VST components.

6.2. Layer Number Study

We conduct experiments to study the optimal numbers of different transformer layers, *i.e.*, $L^{\mathcal{C}}$ in the transformer convertor and $L^{\mathcal{D}}$ in the multi-task transformer decoder, jointly considering computational costs and model performance. Note that there are three decoder modules at three scales in the multi-task transformer decoder, thus we set different transformer layer numbers for them, *i.e.*, $L_3^{\mathcal{D}}$ for 1/16 scale, $L_2^{\mathcal{D}}$ for 1/8 scale, and $L_1^{\mathcal{D}}$ for 1/4 scale. The experimental results on four RGB-D SOD datasets, *i.e.*, NJUD, DUTLF-Depth, STERE, and LFSD, are given in Table 5.

In our initial model setting, we set $L^{\mathcal{C}} = L_3^{\mathcal{D}} = 8$. Since $L_2^{\mathcal{D}}$ and $L_1^{\mathcal{D}}$ are used at relatively large scales, we initially set both of them to 4, as shown in row I in Table 5. Then, we start to change the numbers of different layers.

We first reduce $L_2^{\mathcal{D}}$ and $L_1^{\mathcal{D}}$ from 4 to 2 to save computational costs. The experimental results on row II show that it can get comparable performance with less computational costs compared with row I. Hence, we set $L_2^{\mathcal{D}} = L_1^{\mathcal{D}} = 2$ and start to change $L_3^{\mathcal{D}}$ from 8 to 6, 4, 2, respectively, which are shown in row III, IV, V in Table 5. We find that as $L_3^{\mathcal{D}}$ decreases, the computation costs decrease gradually while the results are generally comparable. However, the model performance on row IV is better than that on row V on DUTLF-Depth and LFSD datasets. Thus, we set $L_3^{\mathcal{D}} = 4$ and start to change $L^{\mathcal{C}}$ from 8 to 6, 4, 2, respectively, which are shown in row VI, VII, VIII. It can be seen that the performance on row VII is the best and the model has acceptable computational costs. Hence, we set $L^{\mathcal{C}} = L_3^{\mathcal{D}} = 4$ and $L_2^{\mathcal{D}} = L_1^{\mathcal{D}} = 2$ as our final model setting.

6.3. More Visual Comparison with State-of-the-art Methods

We give more visual comparison results with the state-of-the-art RGB and RGB-D SOD methods in Figure 4 and Figure 5, respectively. It shows that our VST model can handle well in many challenging scenarios, *i.e.*, big salient objects, cluttered backgrounds, foregrounds and backgrounds with very similar appearance, etc, while existing methods are heavily disturbed in these scenarios. Besides, we also show the boundary maps predicted by our RGB VST and RGB-D VST models in Figure 4 and Figure 5, respectively. It can be seen that our models can predict clear boundaries for salient objects.

Table 4. Ablation studies of our proposed model on RGB SOD datasets. "RC" means RGB Convertor. "Bili" denotes bilinear upsampling and "F" means multi-level token fusion. "TMD" denotes our proposed token-based multi-task decoder, while "C2D" means using conventional two-stream decoder to perform saliency and boundary detection without using task-related tokens. The best results are labeled in **blue**.

Settings	DUTS [63]				HKU-IS [32]				PASCAL-S [34]				SOD [45]			
	S_m	maxF	$E_\xi^{\max}$	MAE	S_m	maxF	$E_\xi^{\max}$	MAE	S_m	maxF	$E_\xi^{\max}$	MAE	S_m	maxF	$E_\xi^{\max}$	MAE
Baseline	0.824	0.780	0.909	0.071	0.858	0.854	0.938	0.075	0.826	0.795	0.878	0.096	0.802	0.803	0.880	0.100
+RC	0.827	0.785	0.913	0.070	0.860	0.856	0.939	0.074	0.830	0.797	0.879	0.095	0.804	0.805	0.880	0.100
+RC+Bili	0.867	0.835	0.929	0.048	0.901	0.901	0.956	0.044	0.856	0.827	0.891	0.074	0.833	0.836	0.891	0.077
+RC+RT2T	0.881	0.856	0.934	0.043	0.914	0.918	0.961	0.037	0.864	0.838	0.896	0.070	0.844	0.850	0.894	0.069
+RC+RT2T+F	0.895	0.874	**0.939**	0.039	0.925	0.932	0.966	0.032	0.871	0.845	0.897	0.068	0.851	0.861	0.899	0.068
+RC+RT2T+F+TMD	**0.896**	**0.877**	**0.939**	**0.037**	**0.928**	**0.937**	**0.968**	**0.030**	**0.873**	**0.850**	**0.900**	**0.067**	**0.854**	**0.866**	**0.902**	**0.065**
+RC+RT2T+F+C2D	0.891	0.870	0.937	0.040	0.924	0.931	0.966	0.033	0.869	0.844	0.896	0.069	0.852	0.860	0.898	0.067

Table 5. Comparison of using different numbers of transformer layers in our VST model. The final model setting is labeled in **blue**.

ID	Layer Num				MACs	Params	NJUD [26]				DUTLF-Depth [52]				STERE [46]				LFSD [33]			
	$L^{\mathcal{C}}$	$L_3^{\mathcal{D}}$	$L_2^{\mathcal{D}}$	$L_1^{\mathcal{D}}$	(G)	(M)	S_m	maxF	$E_\xi^{\max}$	MAE	S_m	maxF	$E_\xi^{\max}$	MAE	S_m	maxF	$E_\xi^{\max}$	MAE	S_m	maxF	$E_\xi^{\max}$	MAE
I	8	8	4	4	48.35	119.30	0.925	0.925	0.955	0.033	0.940	0.947	0.966	0.026	0.910	0.902	0.948	0.039	0.878	0.884	0.914	0.066
II	8	8	2	2	36.78	113.39	0.923	0.922	0.955	0.035	0.943	0.947	0.968	0.025	0.911	0.904	0.948	0.039	0.874	0.878	0.908	0.069
III	8	6	2	2	36.20	110.43	0.921	0.920	0.952	0.036	0.940	0.945	0.966	0.026	0.910	0.904	0.948	0.040	0.875	0.883	0.911	0.067
IV	8	4	2	2	35.61	107.47	0.921	0.920	0.951	0.036	0.942	0.947	0.968	0.026	0.911	0.904	0.949	0.040	0.876	0.880	0.912	0.068
V	8	2	2	2	35.03	104.52	0.922	0.921	0.952	0.036	0.940	0.944	0.965	0.026	0.912	0.906	0.949	0.039	0.873	0.875	0.908	0.068
VI	6	4	2	2	33.30	95.65	0.923	0.921	0.952	0.036	0.943	0.948	0.968	0.024	0.913	0.906	0.949	0.039	0.875	0.878	0.912	0.067
VII	**4**	**4**	**2**	**2**	30.99	83.83	0.922	0.920	0.951	0.035	0.943	0.948	0.969	0.024	0.913	0.907	0.951	0.038	0.882	0.889	0.921	0.061
VIII	2	4	2	2	28.68	72.00	0.923	0.921	0.953	0.036	0.938	0.943	0.963	0.028	0.912	0.906	0.950	0.039	0.881	0.887	0.917	0.062

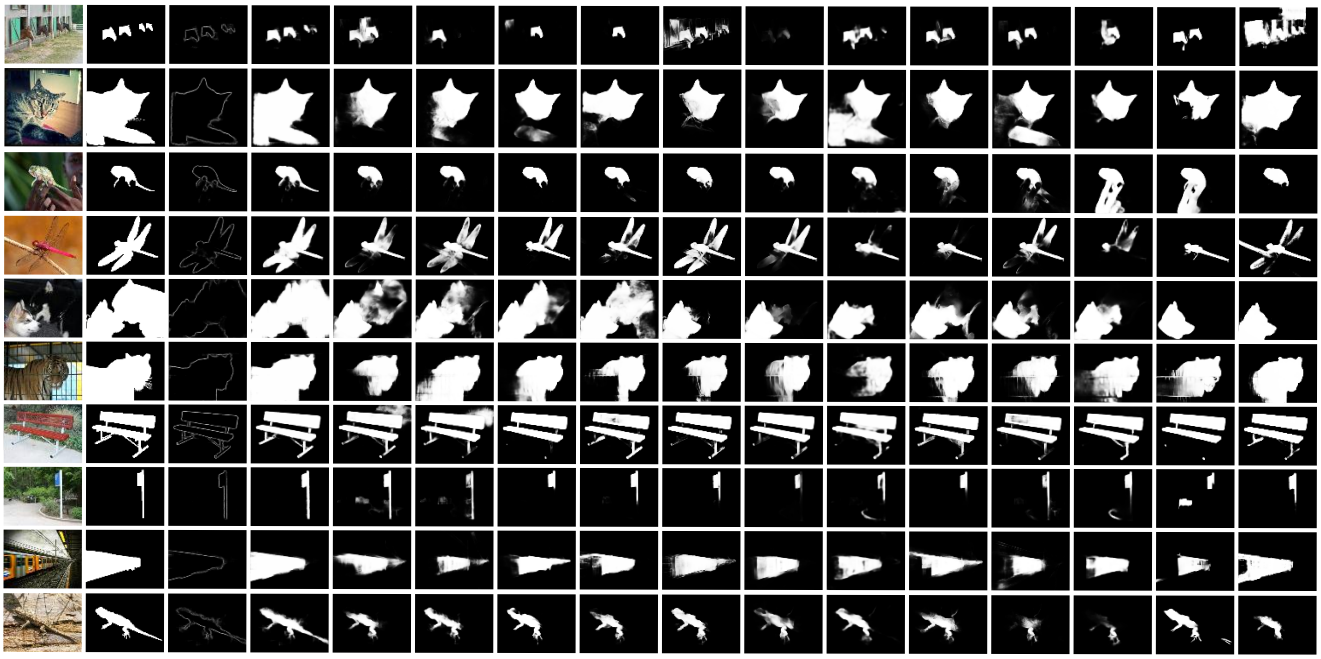

Figure 4. Qualitative comparison against state-of-the-art RGB SOD methods. (GT: ground truth)

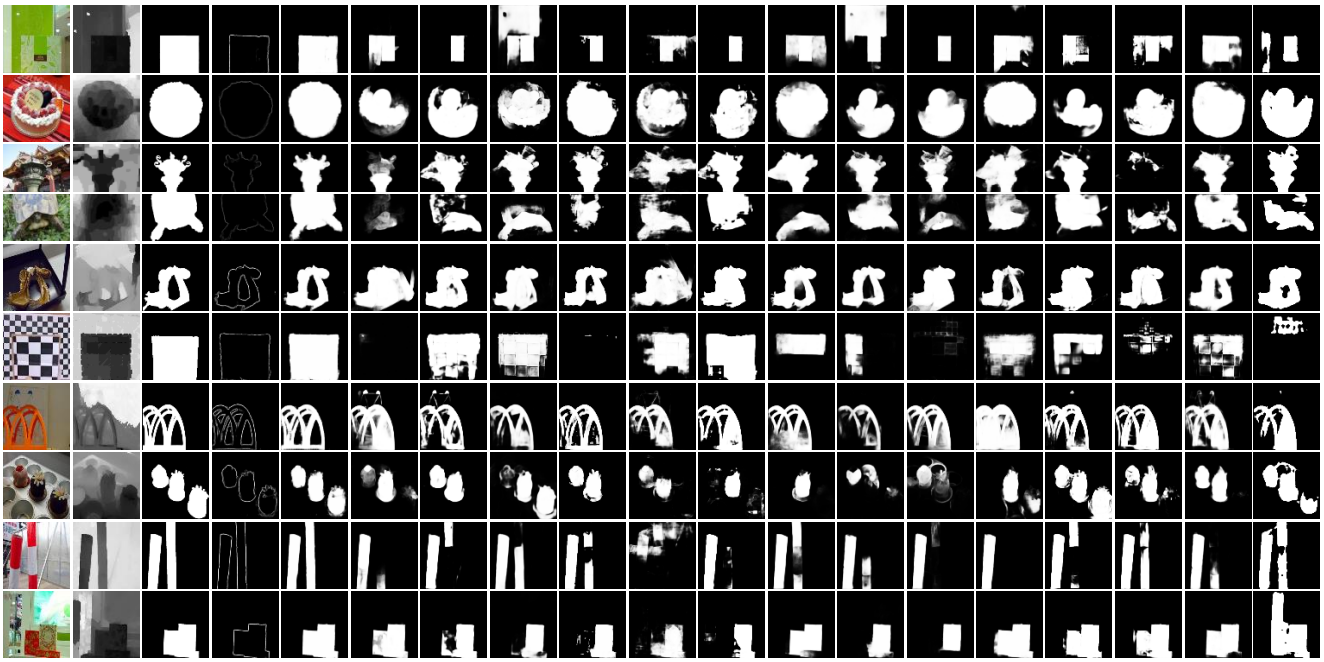

Figure 5. Qualitative comparison against state-of-the-art RGB-D methods. (GT: ground truth)