Samba: A Unified Mamba-based Framework for General Salient Object Detection
Supplementary Material

Jiahao He[1] Keren Fu[1,3,*] Xiaohong Liu[2] Qijun Zhao[1,3]
[1]College of CS, Sichuan University [2]John Hopcroft Center, Shanghai Jiao Tong University
[3]National Key Lab of Fundamental Science on Synthetic Vision, Sichuan University

Table 1. Details about three versions of backbone.

Backbone	The Number of VSS Blocks				Embedded Dimension
	Stage 1	Stage 2	Stage 3	Stage 4	
VMamba-T	2	2	9	2	96
VMamba-S	2	2	27	2	96
VMamba-B	2	2	27	2	128

Abstract

This document provides supplementary materials for the Submission. Sec. 1 elaborates on additional implementation details, including encoder configuration, tri-modal convertor, VSS decoder layers, loss function and experimental details. In Sec. 2, we present two specific explanations of SNS for a better understanding. Further ablation studies on different encoder configuration are given in Sec. 3. More comparison results are reported in Sec .4, incorporating more methods and an additional evaluation metric. Lastly, visual comparison with SOTA methods are drawn in Sec. 5.

1. Additional Implementation Details

1.1. Encoder Configuration

Vmamba [30] provides three backbone versions pretrained on ImageNet [45], namely VMamba-T, VMamba-S, and VMamba-B. The detailed configurations of these backbones are presented in Table 1. For both efficiency and accuracy considerations, we adopt VMamba-S as the encoder of *Samba*. Additionally, ablation studies comparing different encoder configurations are provided in Table 2.

1.2. Tri-modal Convertor

Within the tri-modal convertor, input features, i.e., $f_4^r \in \mathbb{R}^{H_4 \times W_4 \times C_4}$, $f_4^d \in \mathbb{R}^{H_4 \times W_4 \times C_4}$ and $f_4^f \in \mathbb{R}^{H_4 \times W_4 \times C_4}$, are processed by a Linear and a DWConv, respectively. Then they are flattened to $\mathbb{R}^{L \times C_4}$, where $L = H_4 \times W_4$, and then concatenated on the L dimension. Next, we utilize a S6 block to process the concatenated sequence to conduct the interaction of multi-modal information. Finally, the sequence is split to recover three outputs and summed up, followed by a Linear projection. This process can be formulated as:

$$\begin{aligned} \bar{f}_4^r &= DWConv\left(Linear\left(f_4^r\right)\right), \\ \bar{f}_4^d &= DWConv\left(Linear\left(f_4^d\right)\right), \\ \bar{f}_4^f &= DWConv\left(Linear\left(f_4^f\right)\right), \\ \tilde{f}_4^r, \tilde{f}_4^d, \tilde{f}_4^f &= Split\left(S6\left(Cat\left(\bar{f}_4^r, \bar{f}_4^d, \bar{f}_4^f\right)\right)\right), \\ f_4 &= Linear\left(\tilde{f}_4^r + \tilde{f}_4^d + \tilde{f}_4^f\right). \end{aligned} \tag{1}$$

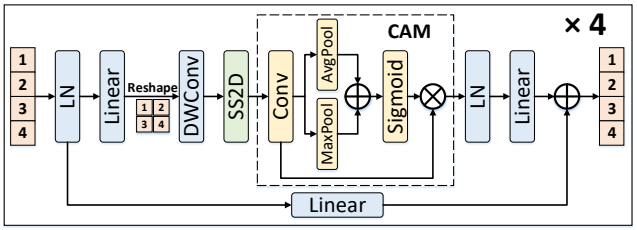

Figure 1. Diagram of a visual state space (VSS) decoder layer.

1.3. VSS Decoder Layers

As we mention in the *Submission*, we implement VSS decoder layers based on VSS blocks [30]. Fig. 1 illustrates the diagram of a VSS decoder layer. Specifically, we modify the VSS block by removing the SiLU activation functions [4], and integrate a channel attention mechanisms (CAM) [12] between the SS2D module and LN layer to explore inter-channel dependencies, resulting in the proposed VSS decoder block. Note that each VSS decoder layer is composed of four VSS decoder blocks.

1.4. Loss Function

We adopt a combination of widely used binary cross entropy (BCE) loss and intersection-over-union (IoU) loss for

*Corresponding author: Keren Fu *(fkrsuper@scu.edu.cn)*.

*The reference link of the PyTorch-based toolbox for evaluating all tasks is https://github.com/zzhanghub/eval-co-sod, and the link of MATLAB-based toolbox is https://github.com/DengPingFan/DAVSOD.

Table 2. Ablation studies of different encoder configurations on five RGB benchmark datasets. "↑" denotes that the larger value is better, and "↓" denotes that the smaller value is better. M represents mean absolute error (MAE) [1, 40]. The best results are stressed in **bold**.

Variant	Configuration	Params (M)	MACs (G)	DUTS[56]				DUT-O[63]				HKU-IS[22]				PASCAL-S[25]				ECSSD[61]			
				$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$
D1	*Samba*(VMamba-T)	33.48	34.03	.925	.921	.958	.023	.881	.847	.914	.041	.939	.952	.971	.021	.883	.885	.921	.052	.949	.961	.972	.022
D2	*Samba*(VMamba-S)	49.59	46.68	.932	.930	.966	**.020**	.889	**.859**	.922	**.037**	.945	**.956**	.978	**.018**	.892	.896	.931	**.047**	.953	.965	.978	.019
D3	*Samba*(VMamba-B)	87.91	82.42	**.934**	**.933**	**.967**	**.020**	**.890**	.857	**.926**	**.037**	**.947**	**.956**	**.979**	**.018**	**.896**	**.899**	**.935**	.048	**.956**	**.967**	**.979**	**.018**

Table 3. Quantitative comparison of our *Samba* against other SOTA RGB SOD methods on five benchmark datasets. "-" indicates the result is not available. "↑" denotes that the larger value is better. The best three results are stressed in red, blue and green.

Method	Params (M)	MACs (G)	DUTS[56]				DUT-O[63]				HKU-IS[22]				PASCAL-S[25]				ECSSD[61]			
			$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$
CNN-based																						
ITSD-R[71]	26.47	15.96	.885	.867	.929	.041	.840	.792	.880	.061	.917	.926	.960	.031	.861	.839	.889	.071	.925	.939	.959	.035
MINet-R[38]	162.38	87.11	.884	.864	.926	.037	.833	.769	.869	.056	.919	.926	.960	.029	.856	.831	.883	.071	.925	.938	.957	.034
LDF-R[58]	25.15	15.51	.892	.877	.930	.034	.839	.782	.870	.052	.920	.929	.958	.028	.861	.839	.888	.067	.925	.938	.954	.034
GateNet-R[70]	128.63	162.22	.891	.874	.932	.038	.840	.782	.878	.055	.921	.926	.959	.031	.863	.836	.886	.071	.924	.935	.955	.038
EDN[59]	42.85	20.41	.892	.893	.933	.035	.849	.821	.884	.049	.924	.940	.963	.027	.864	.879	.907	.062	.927	.950	.957	.032
CSF-R2[9]	36.53	18.96	.890	.869	.929	.037	.838	.775	.869	.055	-	-	-	-	.863	.839	.885	.073	.931	.942	.960	.033
ICON-R[74]	33.09	20.91	.890	.876	.931	.037	.845	.799	.884	.057	.920	.931	.960	.029	.862	.844	.888	.064	.928	.943	.960	.032
MENet[57]	27.83	94.62	.905	.895	.943	.028	.850	.792	.879	.045	.927	.939	.965	.023	.871	.848	.892	.062	.927	.938	.956	.031
Transformer-based																						
VST[28]	44.48	41.36	.896	.877	.939	.037	.850	.800	.888	.058	.928	.937	.968	.030	.873	.850	.900	.067	.932	.944	.964	.034
EBM[65]	118.96	53.38	.909	.900	.949	.029	.858	.817	.900	.051	.930	.943	.971	.023	.877	.856	.899	.061	.941	.954	.972	.024
ICON-S[74]	94.30	52.59	.917	.911	.960	.025	.869	.830	.906	.043	.936	.947	.974	.022	.885	.860	.903	.048	.941	.954	.971	.023
BBRF[33]	74.40	46.00	.908	.905	.951	.025	.855	.820	.898	.044	.935	.946	.936	.020	.871	.884	.925	.049	.939	.957	.972	.021
EVP[29]	-	-	.917	.910	.956	.027	.864	.822	.902	.047	.935	.945	.971	.024	.880	.859	.902	.061	.936	.949	.965	.029
VST-S++ [26]	74.90	32.73	.909	.897	.947	.029	.859	.813	.890	.050	.932	.941	.969	.025	.880	.859	.901	.062	.939	.951	.969	.027
VSCode-T[32]	54.09	72.77	.917	.910	.954	.027	.869	.830	.910	.045	.935	.946	.970	.024	.878	.852	.900	.062	.945	.957	.971	.024
VSCode-S[32]	74.72	93.76	.926	.922	.960	.024	.877	.840	.912	.043	.940	.951	.974	.021	.887	.864	.904	.058	.949	.959	.974	.022
Samba	49.59	46.68	.932	.930	.966	.020	.889	.859	.922	.037	.945	.956	.978	.018	.892	.896	.931	.047	.953	.965	.978	.019

training our *Samba*, which is formulated as:

$$\mathcal{L} = \mathcal{L}_{bce} + \mathcal{L}_{iou}. \tag{2}$$

Our total loss is defined as:

$$\mathcal{L}_{total} = \mathcal{L}\left(S_c, GT\right) + \mathcal{L}\left(S_f, GT\right), \tag{3}$$

where GT represents ground truth, S_c represents the coarse saliency map predicted by f_4^r and S_f represents the final saliency map output by our *Samba*.

1.5. Experimental Details

For VSOD and RGB-D VSOD tasks, we employ RAFT [50] as the optical flow extractor, given its consistently strong performance. Notably, our results for the VSOD task may differ from those reported in previous studies. This discrepancy is due to our adoption of a PyTorch-based toolbox for evaluating all tasks, whereas previous VSOD methods utilize a MATLAB-based toolbox which has different implementation details*.

2. Deeper Discussions of SNS

To better understand the SNS algorithm, we present two specific explanations as follows:

Explanation 1. From the perspective of *Dijkstra's* algorithm, SNS can be regarded as a constrained version of *Dijkstra's* algorithm. The constraint is that all salient patches must be traversed by the algorithm. Thus, the core idea of the SNS algorithm is to identify a path that not only visits all salient patches but also closely approximates the shortest possible route. This distinguishes SNS from the traditional *Dijkstra's* algorithm.

Explanation 2. From another perspective, SNS can be seen as an improved version of the "S" pattern algorithm [62], which we term the "S+" algorithm. The primary improvement lies in the cross-row scanning: instead of rigidly following the conventional "S" shape, the "S+" algorithm calculates the distance between the last salient patch of the current row and the leftmost and rightmost salient patches of the next row, and then select the salient patch with smaller distance as the next scanning patch, thereby maintaining greater spatial continuity of salient patches.

3. Further Ablation Studies

3.1. Different Encoder Configuration

To verify the effectiveness of our selected encoder configuration (VMamba-S), we utilize VMamba-T and Vmamba-B to

Table 4. Quantitative comparison of our *Samba* against other SOTA RGB-D SOD methods on five benchmark datasets.

Method	Params (M)	MACs (G)	NJUD[18]				NLPR[39]				SIP[5]				STERE[35]				DUTLF-D[42]			
			$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$
CNN-based																						
HDFNet[37]	44.15	91.77	.908	.911	.944	.039	.923	.917	.963	.023	.886	.894	.930	.048	.900	.900	.943	.042	.908	.915	.945	.041
CoNet[17]	43.66	20.89	.896	.893	.937	.046	.912	.893	.948	.027	.860	.873	.917	.058	.905	.901	.947	.037	.923	.932	.959	.029
BBSNet[7]	49.77	31.20	.921	.919	.949	.035	.931	.918	.961	.023	.879	.884	.922	.055	.908	.903	.942	.041	.882	.870	.912	.058
JL-DCF[8]	143.52	211.06	.877	.892	.941	.066	.931	.918	.965	.022	.885	.894	.931	.049	.900	.895	.942	.044	.894	.891	.927	.048
SPNet[72]	67.88	175.29	.925	.928	.957	.029	.927	.919	.962	.021	.894	.904	.933	.043	.907	.906	.949	.037	.895	.899	.933	.045
CMINet[64]	188.12	213.00	.929	.934	.957	.029	.932	.922	.963	.021	.899	.910	.939	.040	.918	.916	.951	.032	.912	.913	.938	.038
DCF[16]	53.92	108.60	.904	.905	.943	.039	.922	.910	.957	.024	.874	.886	.922	.052	.906	.904	.948	.037	.925	.930	.956	.030
SPSN[19]	-	-	.918	.921	.952	.032	.923	.912	.960	.023	.892	.900	.936	.043	.907	.902	.945	.035	-	-	-	-
Transformer-based																						
VST[28]	53.83	51.33	.922	.920	.951	.035	.932	.920	.962	.024	.904	.915	.944	.040	.913	.907	.951	.038	.943	.948	.969	.024
SwinNet-B[31]	199.18	122.20	.920	.924	.956	.034	.941	.936	.974	.018	.911	.927	.950	.035	.919	.918	.956	.033	.918	.920	.949	.035
HRTransNet[49]	68.89	18.80	.908	.911	.945	.037	.926	.916	.964	.021	.859	.866	.909	.059	.917	.915	.955	.031	.925	.930	.958	.028
CATNet[48]	262.73	172.06	.932	.937	.960	.025	.938	.934	.971	.017	.910	.928	.951	.034	.920	.922	.958	.030	.952	.958	.975	.018
VST-S++ [26]	143.15	45.41	.928	.928	.957	.031	.935	.925	.964	.021	.904	.918	.946	.038	.921	.916	.954	.034	.945	.950	.969	.024
CPNet[13]	216.50	129.34	.935	.941	.963	.024	.940	.936	.971	.016	.907	.927	.946	.035	.920	.922	.960	.029	.951	.959	.974	.018
VSCode-T[32]	54.09	72.77	.941	.945	.967	.025	.938	.930	.966	.020	.917	.936	.955	.032	.928	.926	.957	.030	.952	.959	.974	.019
VSCode-S[32]	74.72	93.76	.944	.949	.970	.022	.941	.932	.968	.018	.924	.942	.958	.029	.931	.928	.958	.028	.960	.967	.980	.015
Samba	54.94	71.64	.949	.956	.975	.018	.947	.941	.976	.014	.931	.949	.966	.025	.935	.933	.963	.026	.956	.964	.976	.017

replace the encoder of our *Samba* respectively, and evaluate them on five RGB benchmark datasets. As shown in Table 2, the results reveal that:

(1) Compared to "D1", "D2" shows only a modest increase in Params (16.11M) and MACs (12.65G). However, it demonstrates significant performance improvements across various datasets, particularly on the PASCAL-S [25] dataset, with S_m increasing by 0.009, F_m increasing by 0.011, E_m increasing by 0.010, and M decreasing by 0.005.

(2) Compared to "D2", "D3" exhibits a considerable increase in Params (38.32M) and MACs (35.74G), yet its performance improvements across various datasets are marginal. The most notable gains are observed on the PASCAL-S [25] dataset, with S_m increasing by 0.004, F_m by 0.003, E_m by 0.004, and M decreasing by only 0.001.

These experimental results demonstrate that selecting VMamba-S as the encoder of *Samba* achieves an optimal balance between efficiency and accuracy.

4. More Comparison Results

To conserve the space, we only present only 10 state-of-the-art (SOTA) methods for each of RGB SOD, RGB-D SOD, RGB-T SOD and VSOD tasks in the *Submission*. For more comprehensive comparison, we include six additional methods for each of the four tasks, and introduce an additional evaluation metric, the mean absolute error (M) [1, 40], to assess the model performance, as shown in Table 3, 4, 6, 5. The experimental results consistently demonstrate that *Samba* outperforms existing SOTA CNN- and transformer-based SOD models, with a comparable number of Params and relatively low MACs. Notably, we also present the results from another version of VSCode [32], i.e., VSCode-T. Compared to VSCode-T on RGB benchmark datasets, our *Samba* exhibits lower Params and MACs, with MACs reduced by a notable 26.09G. Furthermore, *Samba* achieves significantly better performance across five datasets, particularly on the DUT-O [63] dataset, with S_m increasing by 0.02, F_m increasing by 0.029, and E_m decreasing by 0.008. Compared to VSCode-T on RGB-D, RGB-T and VSOD benchmark datasets, our *Samba* achieves comparable Params and MACs while delivering superior performance across all datasets. It is worth noting that RGB-D VSOD research remains in its early stage, with no additional methods available for comparison.

5. Visual Comparison with SOTA Methods

In this section, we provide extensive visual comparison results with SOTA methods across RGB SOD (Fig. 2), RGB-D SOD (Fig. 3), VSOD (Fig. 5), RGB-T SOD (Fig. 4) and RGB-D VSOD (Fig .6) tasks. These results demonstrate the superior performance and robustness of our *Samba* when tackling a variety of challenging scenarios, including extremely large or small salient objects, object occlusions, multiple objects, and complex backgrounds.

References

[1] Ali Borji, Ming-Ming Cheng, Huaizu Jiang, and Jia Li. Salient object detection: A benchmark. *IEEE TIP*, 24(12):5706–5722, 2015. 2, 3

[2] Gang Chen, Feng Shao, Xiongli Chai, Hangwei Chen, Qiuping Jiang, Xiangchao Meng, and Yo-Sung Ho. Cgmdrnet: Cross-guided modality difference reduction network for rgb-t salient object detection. *IEEE TCSVT*, 32(9):6308–6323, 2022. 4

Table 5. Quantitative comparison of our *Samba* against other SOTA VSOD methods on five benchmark datasets.

Method	Params (M)	MACs (G)	DAVIS[41]				DAVSOD-easy[6]				FBMS[36]				SegV2[20]				VOS[24]			
			$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$
CNN-based																						
PDB[46]	-	-	.880	.851	.949	.030	.705	.590	.803	.109	.850	.821	.882	.072	.871	.820	.867	.024	.818	.742	.845	.077
FGRNE[21]	-	-	.839	.786	.918	.043	.703	.590	.755	.092	.822	.783	.871	.084	.736	.659	.903	.036	.709	.643	.797	.088
RCRNet[60]	53.79	36.86	.884	.845	.947	.028	.726	.601	.773	.078	.873	.850	.902	.055	.829	.747	.901	.038	-	-	-	-
MGAN[23]	91.51	123.57	.913	.894	.965	.021	.740	.611	.778	.073	.909	.903	.946	.026	.902	.869	.950	.016	.797	.725	.829	.064
SSAV[6]	-	-	.891	.857	.945	.029	.751	.646	.783	.089	.880	.856	.922	.043	.850	.797	.922	.024	.783	.702	.839	.080
PCSA[10]	2.63	12.85	.900	.877	.960	.022	.725	.590	.759	.077	.872	.844	.917	.041	.886	.848	.938	.018	.802	.699	.816	.070
TENet[44]	-	-	.905	.881	.958	.016	.756	.638	.793	.066	.911	.904	.938	.027	-	-	-	-	-	-	-	-
FSNet[15]	83.41	35.32	.922	.909	.972	.019	.760	.637	.796	.063	.875	.867	.918	.047	.849	.773	.920	.022	.678	.621	.755	.104
DCFNet[66]	69.56	93.27	.914	.899	.970	.015	.729	.612	.781	.065	.883	.853	.910	.040	.903	.870	.953	.012	.838	.773	.861	.059
UGPL[43]	-	-	.911	.895	.968	.018	.732	.602	.771	.064	.897	.884	.939	.027	.867	.828	.938	.020	.751	.685	.811	.075
MMNet[69]	50.81	82.63	.897	.877	.958	.020	.777	.708	.813	.065	.894	.883	.929	.032	.886	.840	.944	.014	-	-	-	-
Transformer-based																						
MGTNet[34]	150.91	265.21	.925	.919	.976	.014	.765	.653	.800	.063	.900	.881	.929	.034	.903	.861	.946	.012	.814	.727	.819	.064
UFO[11]	55.92	248.80	.918	.906	.978	.015	.747	.626	.799	.063	.858	.868	.911	.051	.888	.850	.951	.015	-	-	-	-
CoSTFormer[27]	-	-	.923	.906	.978	.013	.779	.667	.819	.059	.869	.861	.913	.045	.874	.813	.943	.017	.791	.708	.811	.084
VSCode-T[32]	54.09	72.77	.930	.913	.970	.014	.792	.696	.831	.053	.891	.880	.923	.037	.943	.937	.984	.008	-	-	-	-
VSCode-S[32]	74.72	93.76	.936	.922	.973	.013	.800	.710	.835	.052	.905	.902	.939	.029	.946	.937	.984	.008	-	-	-	-
Samba	54.94	71.64	.943	.936	.985	.009	.813	.734	.856	.043	.925	.922	.954	.022	.943	.938	.987	.006	.870	.820	.898	.040

Table 6. Quantitative comparison of our *Samba* against other SOTA RGB-T SOD methods on three benchmark datasets.

Method	Params (M)	MACs (G)	VT821[54]				VT1000[53]				VT5000[52]			
			$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$	$S_m\uparrow$	$F_m\uparrow$	$E_m\uparrow$	$M\downarrow$
CNN-based														
FCMF[67]	-	-	.760	.667	.810	.081	.873	.851	.921	.037	.814	.758	.866	.055
MIDD[51]	52.43	217.13	.871	.847	.916	.044	.916	.904	.956	.030	.868	.834	.919	.045
ECFFNet[73]	-	-	.877	.835	.911	.034	.924	.919	.959	.021	.876	.850	.922	.037
CGFNet[55]	69.92	382.63	.881	.866	.920	.038	.923	.923	.959	.023	.883	.852	.926	.039
CSRNet[14]	1.01	5.76	.885	.855	.920	.037	.919	.901	.952	.027	.868	.821	.912	.045
ADF[52]	-	-	.808	.749	.841	.077	.909	.908	.950	.034	.863	.837	.911	.048
MGAI[47]	87.09	78.37	.891	.870	.933	.030	.929	.921	.965	.024	.884	.846	.930	.037
TNet[3]	87.04	54.90	.899	.885	.936	.030	.929	.921	.965	.024	.895	.864	.936	.036
CGMDR[2]	-	-	.894	.872	.932	.035	.931	.927	.966	.020	.896	.877	.939	.032
CAVER[48]	55.79	32.15	.891	.874	.933	.033	.936	.927	.970	.021	.892	.857	.935	.035
Transformer-based														
HRTransNet[49]	68.89	18.80	.906	.881	.944	.026	.938	.931	.969	.017	.912	.895	.948	.025
SwinNet-B[31]	199.18	122.20	.904	.877	.937	.029	.938	.933	.974	.020	.912	.885	.944	.028
SPNet[68]	104.03	67.59	.913	.900	.949	.022	.941	.943	.975	.014	.914	.905	.954	.024
VST-S++ [26]	143.15	45.41	.897	.868	.925	.033	.940	.931	.971	.020	.901	.861	.936	.034
VSCode-T[32]	54.09	72.77	.921	.906	.951	.021	.949	.944	.981	.017	.918	.892	.954	.028
VSCode-S[32]	74.72	93.76	.926	.910	.954	.021	.952	.947	.981	.016	.925	.900	.959	.026
Samba	54.94	71.64	.934	.927	.965	.017	.953	.956	.983	.012	.928	.919	.963	.021

[3] Runmin Cong, Kepu Zhang, Chen Zhang, Feng Zheng, Yao Zhao, Qingming Huang, and Sam Kwong. Does thermal really always matter for rgb-t salient object detection? *IEEE TMM*, 25:6971–6982, 2022. 4

[4] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107:3–11, 2018. 1

[5] 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. *IEEE TNNLS*, 32(5):2075–2089, 2020. 3

[6] Deng-Ping Fan, Wenguan Wang, Ming-Ming Cheng, and Jianbing Shen. Shifting more attention to video salient object detection. In *CVPR*, pages 8554–8564, 2019. 4

[7] 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. Springer, 2020. 3

[8] 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. 3

[9] 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

Figure 2. Qualitative comparison of our model against state-of-the-art RGB SOD methods. (GT: ground truth.)

ECCV, pages 702–721. Springer, 2020. 2

[10] Yuchao Gu, Lijuan Wang, Ziqin Wang, Yun Liu, Ming-Ming Cheng, and Shao-Ping Lu. Pyramid constrained self-attention network for fast video salient object detection. In *AAAI*, volume 34, pages 10869–10876, 2020. 4

[11] Ruohao Guo, Xianghua Ying, Yanyu Qi, and Liao Qu. Unitr: A unified transformer-based framework for co-object and multi-modal saliency detection. *IEEE TMM*, 2024. 4

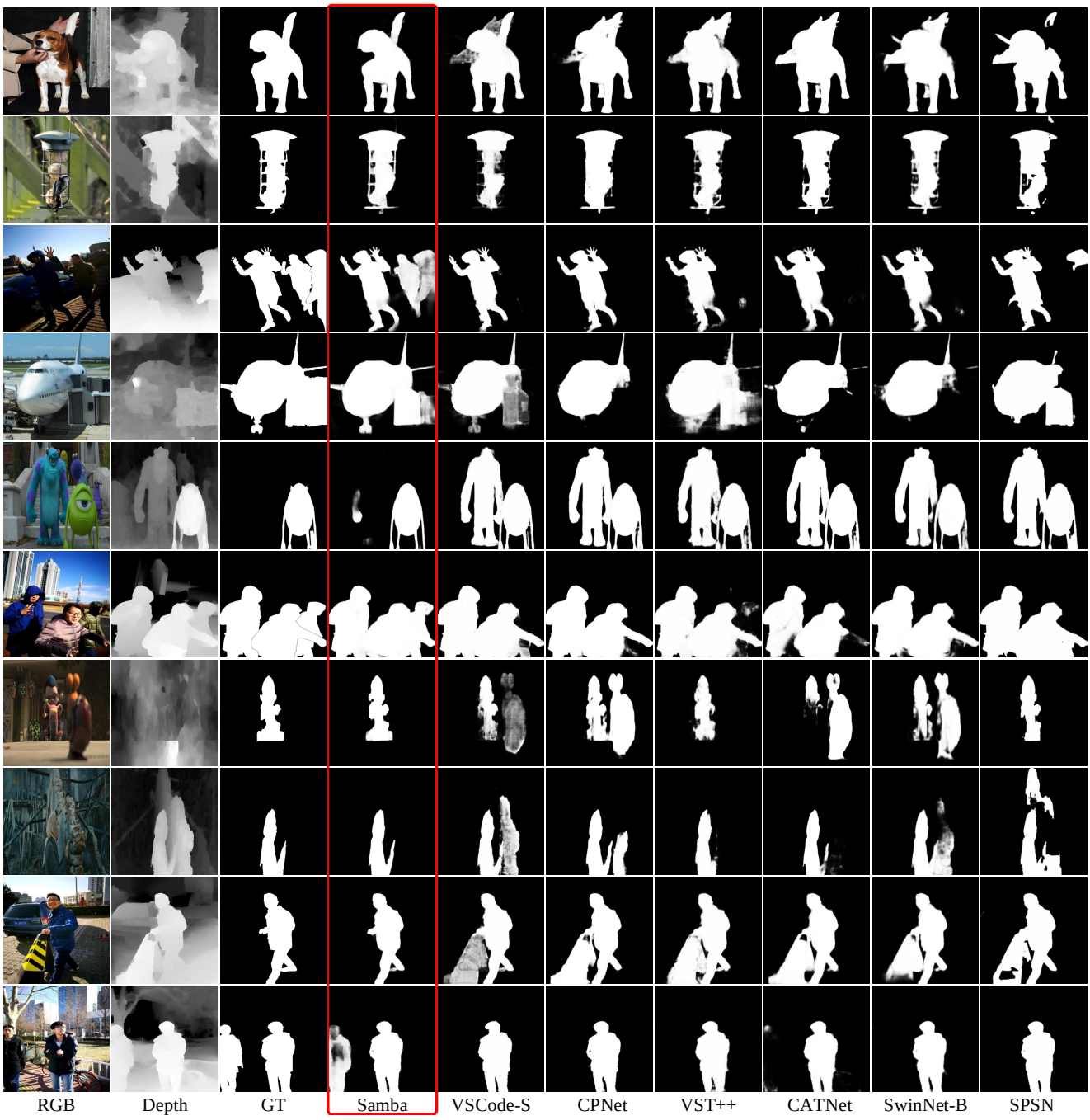

Figure 3. Qualitative comparison of our model against state-of-the-art RGB-D SOD methods. (GT: ground truth.)

[12] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, pages 7132–7141, 2018. 1

[13] Xihang Hu, Fuming Sun, Jing Sun, Fasheng Wang, and Haojie Li. Cross-modal fusion and progressive decoding network for rgb-d salient object detection. *IJCV*, pages 1–19, 2024. 3

[14] Fushuo Huo, Xuegui Zhu, Lei Zhang, Qifeng Liu, and Yu Shu. Efficient context-guided stacked refinement network for rgb-t salient object detection. *IEEE TCSVT*, 32(5):3111–3124, 2021. 4

[15] Ge-Peng Ji, Keren Fu, Zhe Wu, Deng-Ping Fan, Jianbing Shen, and Ling Shao. Full-duplex strategy for video object segmentation. In *ICCV*, pages 4922–4933, 2021. 4

[16] Wei Ji, Jingjing Li, Shuang Yu, Miao Zhang, Yongri Piao, Shunyu Yao, Qi Bi, Kai Ma, Yefeng Zheng, Huchuan Lu, et al. Calibrated rgb-d salient object detection. In *CVPR*, pages 9471–9481, 2021. 3

[17] Wei Ji, Jingjing Li, Miao Zhang, Yongri Piao, and Huchuan Lu. Accurate rgb-d salient object detection via collaborative

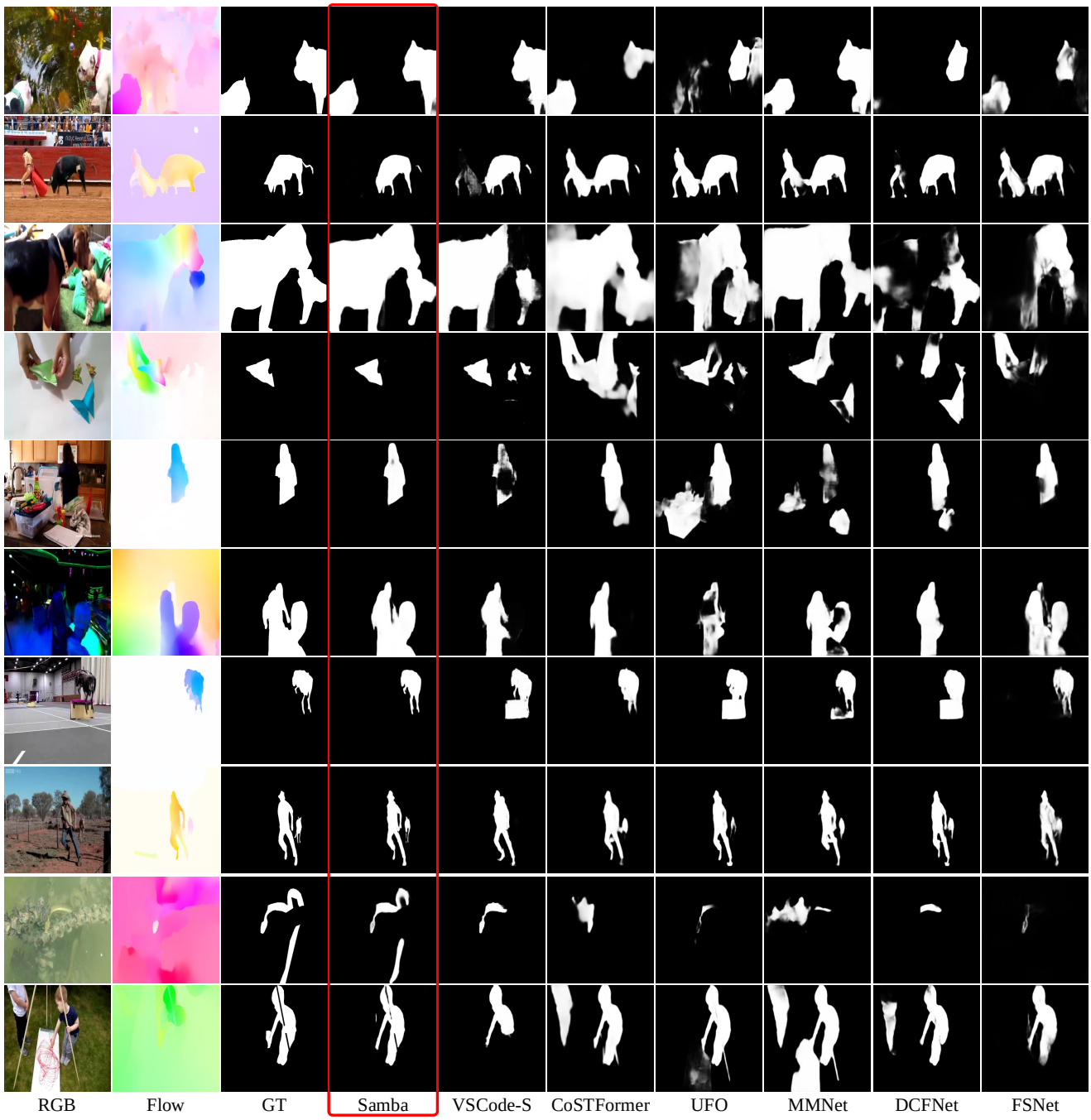

Figure 4. Qualitative comparison of our model against state-of-the-art VSOD methods. (GT: ground truth.)

learning. In *ECCV*, pages 52–69. Springer, 2020. 3

[18] Ran Ju, Ling Ge, Wenjing Geng, Tongwei Ren, and Gangshan Wu. Depth saliency based on anisotropic center-surround difference. In *ICIP*, pages 1115–1119. IEEE, 2014. 3

[19] Minhyeok Lee, Chaewon Park, Suhwan Cho, and Sangyoun Lee. Spsn: Superpixel prototype sampling network for rgb-d salient object detection. In *ECCV*, pages 630–647. Springer, 2022. 3

[20] Fuxin Li, Taeyoung Kim, Ahmad Humayun, David Tsai, and James M Rehg. Video segmentation by tracking many figure-ground segments. In *ICCV*, pages 2192–2199, 2013. 4

[21] Guanbin Li, Yuan Xie, Tianhao Wei, Keze Wang, and Liang Lin. Flow guided recurrent neural encoder for video salient object detection. In *CVPR*, pages 3243–3252, 2018. 4

[22] Guanbin Li and Yizhou Yu. Visual saliency based on multi-scale deep features. In *CVPR*, pages 5455–5463, 2015. 2

[23] Haofeng Li, Guanqi Chen, Guanbin Li, and Yizhou Yu. Motion guided attention for video salient object detection. In

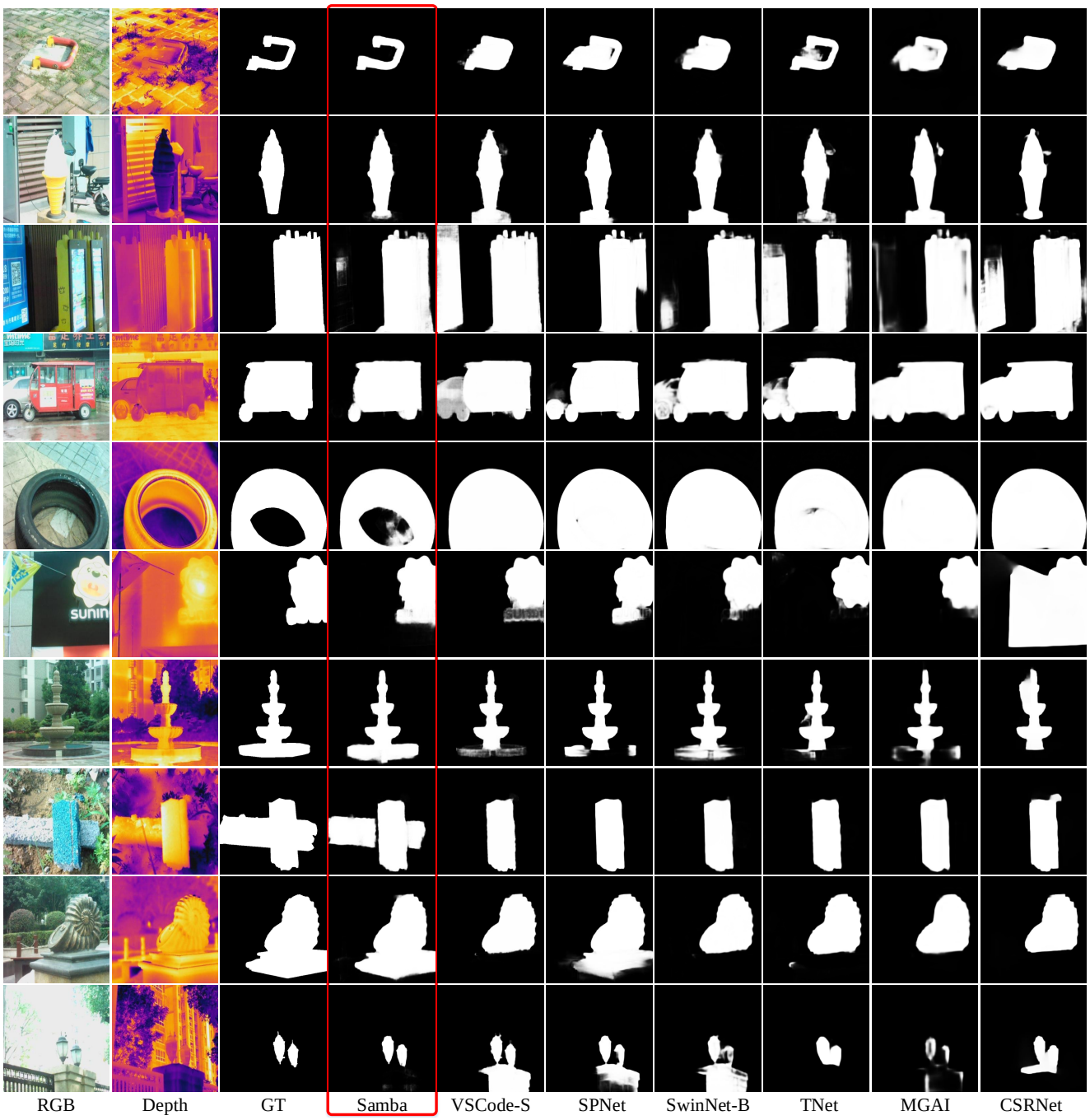

Figure 5. Qualitative comparison of our model against state-of-the-art RGB-T SOD methods. (GT: ground truth.)

ICCV, pages 7274–7283, 2019. 4

[24] Jia Li, Changqun Xia, and Xiaowu Chen. A benchmark dataset and saliency-guided stacked autoencoders for video-based salient object detection. *IEEE TIP*, 27(1):349–364, 2017. 4

[25] 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. 2, 3

[26] Nian Liu, Ziyang Luo, Ni Zhang, and Junwei Han. Vst++: Efficient and stronger visual saliency transformer. *IEEE TPAMI*, 2024. 2, 3, 4

[27] Nian Liu, Kepan Nan, Wangbo Zhao, Xiwen Yao, and Junwei Han. Learning complementary spatial–temporal transformer for video salient object detection. *IEEE TNNLS*, 2023. 4

[28] Nian Liu, Ni Zhang, Kaiyuan Wan, Ling Shao, and Junwei Han. Visual saliency transformer. In *ICCV*, pages 4722–4732, 2021. 2, 3

[29] Weihuang Liu, Xi Shen, Chi-Man Pun, and Xiaodong Cun.

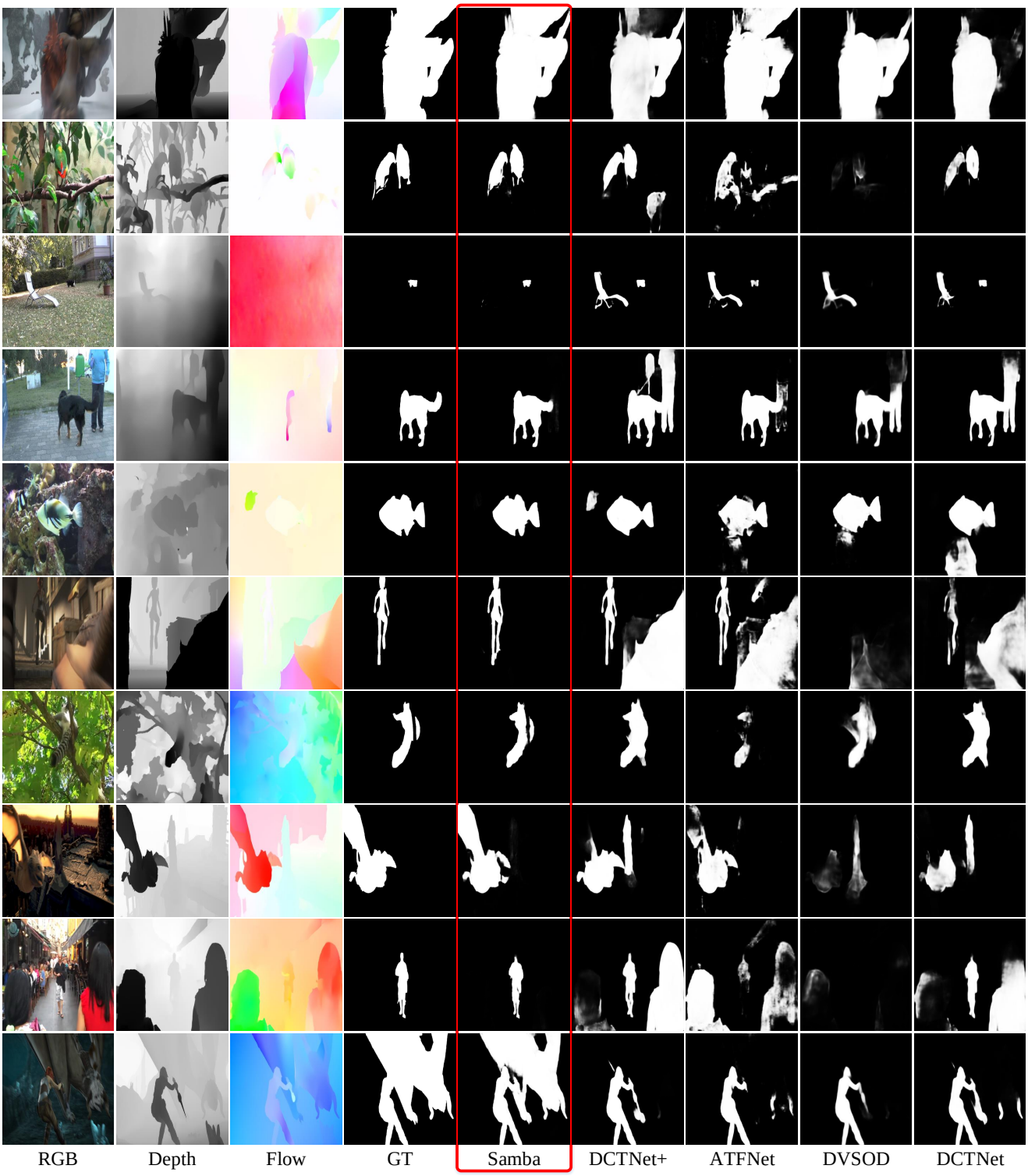

Figure 6. Qualitative comparison of our model against state-of-the-art RGB-D VSOD methods. (GT: ground truth.)

Explicit visual prompting for low-level structure segmentations. In *CVPR*, pages 19434–19445, 2023. 2

[30] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. Vmamba: Visual state space model. *arXiv preprint arXiv:2401.10166*, 2024. 1

[31] Zhengyi Liu, Yacheng Tan, Qian He, and Yun Xiao. Swinnet: Swin transformer drives edge-aware rgb-d and rgb-t salient object detection. *IEEE TCSVT*, 32(7):4486–4497, 2021. 3, 4

[32] Ziyang Luo, Nian Liu, Wangbo Zhao, Xuguang Yang, Dingwen Zhang, Deng-Ping Fan, Fahad Khan, and Junwei Han. Vscode: General visual salient and camouflaged object detection with 2d prompt learning. In *CVPR*, pages 17169–17180, 2024. 2, 3, 4

[33] Mingcan Ma, Changqun Xia, Chenxi Xie, Xiaowu Chen, and Jia Li. Boosting broader receptive fields for salient object detection. *IEEE TIP*, 32:1026–1038, 2023. 2

[34] Dingyao Min, Chao Zhang, Yukang Lu, Keren Fu, and Qijun Zhao. Mutual-guidance transformer-embedding network for video salient object detection. *IEEE SPL*, 29:1674–1678, 2022. 4

[35] Yuzhen Niu, Yujie Geng, Xueqing Li, and Feng Liu. Leveraging stereopsis for saliency analysis. In *CVPR*, pages 454–461. IEEE, 2012. 3

[36] Peter Ochs, Jitendra Malik, and Thomas Brox. Segmentation of moving objects by long term video analysis. *IEEE TPAMI*, 36(6):1187–1200, 2013. 4

[37] Youwei Pang, Lihe Zhang, Xiaoqi Zhao, and Huchuan Lu. Hierarchical dynamic filtering network for rgb-d salient object detection. In *ECCV*, pages 235–252. Springer, 2020. 3

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

[39] Houwen Peng, Bing Li, Weihua Xiong, Weiming Hu, and Rongrong Ji. Rgbd salient object detection: A benchmark and algorithms. In *ECCV*, pages 92–109. Springer, 2014. 3

[40] Federico Perazzi, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. Saliency filters: Contrast based filtering for salient region detection. In *CVPR*, pages 733–740. IEEE, 2012. 2, 3

[41] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, pages 724–732, 2016. 4

[42] 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. 3

[43] Yongri Piao, Chenyang Lu, Miao Zhang, and Huchuan Lu. Semi-supervised video salient object detection based on uncertainty-guided pseudo labels. In *NeurIPS*, volume 35, pages 5614–5627, 2022. 4

[44] Sucheng Ren, Chu Han, Xin Yang, Guoqiang Han, and Shengfeng He. Tenet: Triple excitation network for video salient object detection. In *ECCV*, pages 212–228. Springer, 2020. 4

[45] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115:211–252, 2015. 1

[46] Hongmei Song, Wenguan Wang, Sanyuan Zhao, Jianbing Shen, and Kin-Man Lam. Pyramid dilated deeper convlstm for video salient object detection. In *ECCV*, pages 715–731, 2018. 4

[47] Kechen Song, Liming Huang, Aojun Gong, and Yunhui Yan. Multiple graph affinity interactive network and a variable illumination dataset for rgbt image salient object detection. *IEEE TCSVT*, 33(7):3104–3118, 2022. 4

[48] Fuming Sun, Peng Ren, Bowen Yin, Fasheng Wang, and Haojie Li. Catnet: A cascaded and aggregated transformer network for rgb-d salient object detection. *IEEE TMM*, 2023. 3, 4

[49] Bin Tang, Zhengyi Liu, Yacheng Tan, and Qian He. Hrtransnet: Hrformer-driven two-modality salient object detection. *TCSVT*, 33(2):728–742, 2022. 3, 4

[50] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, pages 402–419. Springer, 2020. 2

[51] Zhengzheng Tu, Zhun Li, Chenglong Li, Yang Lang, and Jin Tang. Multi-interactive dual-decoder for rgb-thermal salient object detection. *IEEE TIP*, 30:5678–5691, 2021. 4

[52] Zhengzheng Tu, Yan Ma, Zhun Li, Chenglong Li, Jieming Xu, and Yongtao Liu. Rgbt salient object detection: A large-scale dataset and benchmark. *IEEE TMM*, 25:4163–4176, 2022. 4

[53] Zhengzheng Tu, Tian Xia, Chenglong Li, Xiaoxiao Wang, Yan Ma, and Jin Tang. Rgb-t image saliency detection via collaborative graph learning. *IEEE TMM*, 22(1):160–173, 2019. 4

[54] Guizhao Wang, Chenglong Li, Yunpeng Ma, Aihua Zheng, Jin Tang, and Bin Luo. Rgb-t saliency detection benchmark: Dataset, baselines, analysis and a novel approach. In *IGTA*, pages 359–369. Springer, 2018. 4

[55] Jie Wang, Kechen Song, Yanqi Bao, Liming Huang, and Yunhui Yan. Cgfnet: Cross-guided fusion network for rgb-t salient object detection. *IEEE TCSVT*, 32(5):2949–2961, 2021. 4

[56] 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. 2

[57] Yi Wang, Ruili Wang, Xin Fan, Tianzhu Wang, and Xiangjian He. Pixels, regions, and objects: Multiple enhancement for salient object detection. In *CVPR*, pages 10031–10040, 2023. 2

[58] 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. 2

[59] Yu-Huan Wu, Yun Liu, Le Zhang, Ming-Ming Cheng, and Bo Ren. Edn: Salient object detection via extremely-downsampled network. *IEEE TIP*, 31:3125–3136, 2022. 2

[60] Pengxiang Yan, Guanbin Li, Yuan Xie, Zhen Li, Chuan Wang, Tianshui Chen, and Liang Lin. Semi-supervised video salient object detection using pseudo-labels. In *ICCV*, pages 7284–7293, 2019. 4

[61] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. Hierarchical saliency detection. In *CVPR*, pages 1155–1162, 2013. 2

[62] Chenhongyi Yang, Zehui Chen, Miguel Espinosa, Linus Ericsson, Zhenyu Wang, Jiaming Liu, and Elliot J Crowley. Plainmamba: Improving non-hierarchical mamba in visual recognition. *arXiv preprint arXiv:2403.17695*, 2024. 2

[63] 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. 2, 3

[64] Jing Zhang, Deng-Ping Fan, Yuchao Dai, Xin Yu, Yiran Zhong, Nick Barnes, and Ling Shao. Rgb-d saliency de-

tection via cascaded mutual information minimization. In *ICCV*, pages 4338–4347, 2021. 3

[65] Jing Zhang, Jianwen Xie, Nick Barnes, and Ping Li. Learning generative vision transformer with energy-based latent space for saliency prediction. In *NeurIPS*, volume 34, pages 15448–15463, 2021. 2

[66] Miao Zhang, Jie Liu, Yifei Wang, Yongri Piao, Shunyu Yao, Wei Ji, Jingjing Li, Huchuan Lu, and Zhongxuan Luo. Dynamic context-sensitive filtering network for video salient object detection. In *ICCV*, pages 1553–1563, 2021. 4

[67] Qiang Zhang, Nianchang Huang, Lin Yao, Dingwen Zhang, Caifeng Shan, and Jungong Han. Rgb-t salient object detection via fusing multi-level cnn features. *IEEE TIP*, 29:3321–3335, 2019. 4

[68] Zihao Zhang, Jie Wang, and Yahong Han. Saliency prototype for rgb-d and rgb-t salient object detection. In *ACM MM*, pages 3696–3705, 2023. 4

[69] Xing Zhao, Haoran Liang, Peipei Li, Guodao Sun, Dongdong Zhao, Ronghua Liang, and Xiaofei He. Motion-aware memory network for fast video salient object detection. *IEEE TIP*, 2024. 4

[70] 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. Springer, 2020. 2

[71] 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

[72] Tao Zhou, Huazhu Fu, Geng Chen, Yi Zhou, Deng-Ping Fan, and Ling Shao. Specificity-preserving rgb-d saliency detection. In *ICCV*, pages 4681–4691, 2021. 3

[73] Wujie Zhou, Qinling Guo, Jingsheng Lei, Lu Yu, and Jenq-Neng Hwang. Ecffnet: Effective and consistent feature fusion network for rgb-t salient object detection. *IEEE TCSVT*, 32(3):1224–1235, 2021. 4

[74] Mingchen Zhuge, Deng-Ping Fan, Nian Liu, Dingwen Zhang, Dong Xu, and Ling Shao. Salient object detection via integrity learning. *IEEE TPAMI*, 45(3):3738–3752, 2022. 2