Applied Soft Computing Journal 170 (2025) 112685

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

TFGNet: Frequency-guided saliency detection for complex scenes

Yi Wang [a], Ruili Wang [b,c,*], Juncheng Liu [b], Rui Xu [a], Tianzhu Wang [b], Feng Hou [b], Bin Liu [a], Na Lei [a]

[a] School of Software, Dalian University of Technology, Dalian, China
[b] School of Mathematical and Computational Sciences, Massey University, Auckland, New Zealand
[c] School of Computer Science, University of Nottingham Ningbo China, Ningbo, China

HIGHLIGHTS

- Saliency detection struggles with boundary identification in complex scenes due to errors in multiscale feature fusion.
- TFGNet enhances boundary and inner region detection by learning high and low spatial frequency features separately.
- TFGNet employs pixel and mask-level decoders to obtain more comprehensive saliency features.
- A histogram dissimilarity loss ensures frequency distribution consistency between ground truth and predicted saliency maps.
- TFGNet surpasses leading methods with more accurate and complete boundaries in complex scenes.

ARTICLE INFO

Dataset link: Predicted maps

Keywords:
Salient object detection
Spatial frequency
Convolutional neural network
Transformer

ABSTRACT

Salient object detection (SOD) with accurate boundaries in complex and chaotic natural or social scenes remains a significant challenge. Many edge-aware or/and two-branch models rely on exchanging global and local information between multistage features, which can propagate errors and lead to incorrect predictions. To address this issue, this work explores the fundamental problems in current U-Net architecture-based SOD models from the perspective of image spatial frequency decomposition and synthesis. A concise and efficient Frequency-Guided Network (TFGNet) is proposed that simultaneously learns the boundary details (high-spatial frequency) and inner regions (low-spatial frequency) of salient regions in two separate branches. Each branch utilizes a Multiscale Frequency Feature Enhancement (FFE) module to learn pixel-wise frequency features and a Transformer-based decoder to learn mask-wise frequency features, improving a comprehensive understanding of salient regions. TFGNet eliminates the need to exchange global and local features at intermediate layers of the two branches, thereby reducing interference from erroneous information. A hybrid loss function is also proposed to combine BCE, IoU, and Histogram dissimilarity to ensure pixel accuracy, structural integrity, and frequency distribution consistency between ground truth and predicted saliency maps. Comprehensive evaluations have been conducted on five widely used SOD datasets and one underwater SOD dataset, demonstrating the superior performance of TFGNet compared to state-of-the-art methods. The codes and results are available at https://github.com/yiwangtz/TFGNet.

1. Introduction

Salient object detection (SOD) is a computer vision task that aims to identify and locate the most visually distinctive objects or regions within an image [1,2]. SOD serves as a foundational step for many high-level vision tasks, enabling effective analysis and interpretation of visual content [3]. The primary goal of SOD is to classify image elements into foreground (salient) and background (non-salient) categories. Unlike tasks that focus on specific object categories or individual instances, SOD emphasizes the contrast between the target and the background [2]. Achieving this requires SOD to balance two opposing objectives in salient feature learning: capturing global contexts and preserving local details [1]. Global contexts are essential for determining salient regions, requiring globally consistent features that exhibit invariance to various

* Corresponding author at: School of Mathematical and Computational Sciences, Massey University, Auckland, New Zealand.
E-mail addresses: dlutwangyi@dlut.edu.cn (Y. Wang), ruili.wang@massey.ac.nz (R. Wang), ljc91122@icloud.com (J. Liu), xurui@dlut.edu.cn (R. Xu), wangtz@126.com (T. Wang), f.hou@massey.ac.nz (F. Hou), liubin@dlut.edu.cn (B. Liu), nalei@dlut.edu.cn (N. Lei).

https://doi.org/10.1016/j.asoc.2024.112685
Received 22 April 2024; Received in revised form 26 November 2024; Accepted 26 December 2024
Available online 27 December 2024

Fig. 1. Visual examples of challenging real-world scenes, such as multiple connected objects and geometrically complex structures, reveal that state-of-the-art (SOTA) models produce less precise boundaries than ours. The high-spatial frequency (HSF) components of ground-truth (GT) maps represent the boundaries of salient regions, which can be used as auxiliary supervision for boundary feature learning.

background interferences. Local details, especially boundary details, are crucial for precisely segmenting salient regions; however, they often exhibit imbalanced distributions and poor consistency, particularly for small, intricate, or multi-connected objects with significant variations in appearance or geometry, as demonstrated in Fig. 1.

A U-Net-like encoder-decoder architecture [4] has been widely adopted to address the challenge of salient feature learning and has become a cornerstone of SOD models [1,2]. This structure employs a top-down encoder and a bottom-up decoder to locate salient regions and refine details. The encoder uses convolution and down-sampling to identify the approximate location of salient regions and extract their global semantic features. These features are progressively refined in the decoder path by combining information from different layers, producing a fine-grained saliency map at the highest resolution. Skip connections between corresponding encoder and decoder layers preserve global and detailed information. While many models based on this architecture employ various refinement strategies to improve performance, detection results still need improvement in complex scenes. These include cases where the salient region contains multiple objects with significant differences in appearance (Row 1, Fig. 1), objects with complex geometric shapes (Row 2, Fig. 1), or low contrast between the salient object and the background in terms of appearance (Row 3, Fig. 1) or shape (Rows 4 and 5, Fig. 1).

As signals, images can be understood in terms of various spatial frequencies. High-spatial frequency (HSF) components represent fine details, such as edges and textures, while low-spatial frequency (LSF) components capture broader structures, such as object shapes and spatial context. By examining U-Net-based SOD models through the lens of frequency theory, we can gain deeper insights into the existing issues associated with this architecture. In U-Net-based SOD models, the encoder performs a form of frequency decomposition through down-sampling and convolutional operations. Each down-sampling layer progressively reduces spatial resolution, removing HSF details (e.g., edges) while retaining LSF components. The convolution layers also

contribute to this by aggregating information over local neighborhoods, further reducing fine detail. The decoder, through up-sampling and skip connections, functions as a frequency synthesis process. Up-sampling gradually restores spatial resolution, while skip connections feed higher-frequency information from early layers back into the synthesis process, helping preserve sharp edges and textures. However, skip connections alone may not be sufficient to fully recover lost high-frequency details, especially after extensive down-sampling and convolutions.

A known issue with this architecture is the degradation of spatial frequency information during down-sampling [1,2]. Many SOD models use Convolutional Neural Networks (CNN) [5]-based backbones, such as ResNet [6] or VGG [7], originally designed for classification tasks [1,2]. These networks reduce feature map resolution through multiple down-sampling stages, which can lead to the loss or corruption of fine spatial details. Although skip connections provide some frequency information from earlier layers, it is often inadequate for fully restoring sharp boundaries and fine textures. To address this, multiscale feature enhancement strategies, such as Atrous Spatial Pyramid Pooling (ASPP) [8] and attention mechanisms [9] adopted in many SOD methods [10,11,12,13]. These approaches help capture details at different spatial scales, compensating for the frequency loss introduced by down-sampling. By processing features at multiple receptive fields, such techniques can preserve both high-frequency and low-frequency components, enhancing the model's ability to detect salient objects across varying spatial resolutions.

Another issue is caused by the interaction of HSF and LSF features in boundary-wise two-branch SOD models. In the SOD task, we want a binary/grayscale saliency map to indicate the salient regions, as shown in Fig. 1. This map usually has a unified distribution in the foreground without background noise. A salient map's HSF components are the boundary of the foreground and can be computed by edge detectors [5]. For this reason, many boundary-wise SOD methods explicitly use auxiliary boundary information—specifically, the HSF of ground truth (GT) saliency maps—as an additional supervisory signal to improve overall segmentation quality [1,2]. These models typically employ a two-branch decoder structure: one branch focuses on boundary features, while the other captures the entire salient region [14,15,16] or the inner regions [10,11,17]. However, many models, such as SFENet [10], MENet [11], DCN[16], and LDF [17], require the exchange of information between the HSF and LSF components of two branches at one or more stages. This can lead to inaccuracies in intermediate features propagating through the network, ultimately compromising the final detection performance.

We present a concise and compelling Transformer (TF)-based Frequency-Guided Network (TFGNet) to address the above issues. TFGNet utilizes a two-branch structure, with one branch refining high-spatial frequency (HSF) boundary details and the other handling low-spatial frequency (LSF) inner regions, through pixel-level and mask-level decoders in each branch. Crucially, it avoids any intermediate information exchange between the HSF and LSF features of the two branches. In contrast, models like MENet and SFENet also employ a two-branch architecture but differ in their approach to feature refinement. SFENet adopts a two-stage decoding process, while MENet uses four stages to enhance salient features. In both models, HSF and LSF information from previous stages is fused multiple times, allowing global and detailed information to complement each other at each stage. TFGNet, however, requires only a single-stage decoding and fusion process, making the architecture more concise and efficient.

The key to TFGNet's efficiency lies in the design of its feature optimization modules within each branch. Each branch of TFGNet incorporates a pixel-wise decoder alongside a TF-decoder, inspired by MaskFormer [18], to optimize features by combining per-pixel embeddings and per-mask embeddings in the HSF and LSF branches. Unlike MaskFormer, TFGNet's pixel-wise decoder employs a Multiscale Frequency Feature Enhancement (FFE) strategy, progressively enhancing, up-sampling, and aggregating multiscale features. This strategy leverages a global-local attention mechanism [9] to capture vital contextual information for extracting per-pixel embeddings. Unlike SFENet and MENet, which rely solely on pixel-level feature extraction, TFGNet integrates a TF-decoder to learn mask-level discriminative features and generate per-mask embeddings. By combining FFE with a TF-decoder, TFGNet achieves a comprehensive understanding of salient regions, significantly improving its local representation and overall performance.

In loss design, we also introduce the general pixel-intensity frequency distribution, the histogram dissimilarity measurement, into the loss functions to enhance overall distribution consistency. Then, by combining pixel-wise Binary Cross-Entropy (BCE) [19] and structure-wise Intersection Over Union (IoU) [20] losses, we create a hybrid loss that ensures a comprehensive and effective training process for TFGNet. As far as we are aware, this is the first instance of a histogram-based loss being utilized in an SOD model.

The streamlined design of TFGNet across the encoder, feature optimization, and decoder components results in a more effective and efficient model for detecting salient objects, capitalizing on the strengths of both frequency decomposition and transformer architectures.

In short, this paper presents the following contributions.

- We present TFGNet, a novel and efficient Transformer-based frequency-guided framework designed for SOD tasks. TFGNet is based on the image spatial frequency decomposition and synthesis principle, allowing the model to learn salient features through separate branches for high- and low-frequency components. By combining a pixel-level Multiscale Frequency Feature Enhancement (FFE) decoder with a mask-level Transformer decoder, each branch produces comprehensive embedding that strengthens the model's capacity for precise and reliable predictions.
- For the first time, we propose to use histogram dissimilarity measurement in the loss function. A comprehensive hybrid loss that integrates BCE, IoU, and Histogram-based loss functions is applied in TFGNet to ensure the predicted salient maps align with the ground truth maps regarding pixel accuracy, structural integrity, and frequency distribution consistency.
- A comprehensive evaluation of TFGNet with 23 SOD methods on five widely used SOD datasets and an underwater SOD dataset. The results demonstrate that TFGNet can accurately localize salient objects with more complete and precise boundaries on various complex backgrounds.

The rest of this paper is structured as shown below: Section 2 briefly reviews related SOD approaches to this work. Section 3 explains the details of TFGNet. Section 4 demonstrates and discusses the proposed method through quantitative and qualitative experiments. Section 5 outlines the primary contributions of this work.

2. Related work

Recently, various SOD approaches have been proposed successively [1,2]. The following brief overview provides a summary of recent strategies in SOD research.

2.1. CNN-based SOD model

Convolutional Neural Network (CNN)-based SOD methods demonstrate promising results across various benchmark datasets [1,2]. Most of these models adopt a framework similar to U-Net. However, more than semantic global features in top layers are needed to localize salient objects accurately, so it is possible to guide detail learning incorrectly. Hence, the decoder design is crucial for generating accurate salient maps. CPDNet [21] proposes a partial decoder to refine high-level features to generate precise saliency maps. EGNet [16] proposes an edge guidance network using complementary information about salient edges

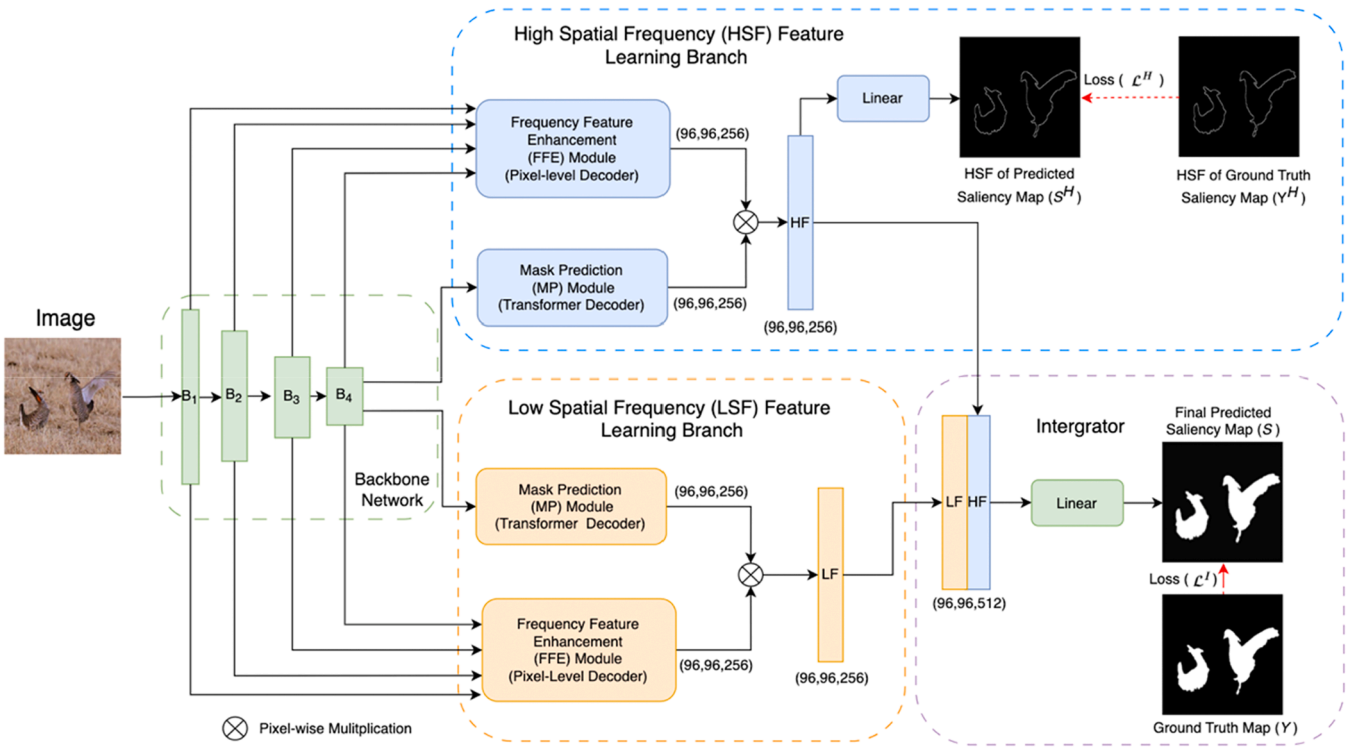

Fig. 2. Illustration of the proposed TFGNet. Multiscale features are initially extracted from a backbone network. Subsequently, a two-branch frequency learning process refines and optimizes high- and low-frequency salient features separately. Finally, these salient features are integrated to produce the final full-band saliency maps.

and objects. BASNet [14] sequentially stacks two U-Nets with different configurations. AADFNet [12] uses an attentional dense ASPP-based network to selectively use small and large dilated rate convolutions to obtain local and global saliency information. GateNet [22] designs a gated dual-branch structure to establish a cooperative relationship between features of different levels to increase network discriminability. MINet [23] proposes to enhance the feed-forward neural network by adopting a refinement mechanism for multiple stages. LDF [17] divides a GT map into a body map and a detailed map, allowing collaborative supervision of body parts and details of the saliency regions. U2Net [24] proposes a novel ReSidual U-block (RSU), which can obtain multi-resolution features in the intrastate without reducing the resolution of the feature map. PA-KRN [25] performs intermediate edge supervision on its five feature layers in its exemplary segmentation module to ensure that the boundaries provided by the encoding process are clear. HQSOD [26] divides the process into two networks: a low-resolution saliency classification network (LRSCN) and a high-resolution refinement network (HRRN). The LRSCN determines salient, background, and ambiguous regions at a lower resolution. HRRN enhances pixel saliency, particularly in ambiguous regions, ensuring sharp boundaries at high resolution with minimal GPU consumption. DCN [15] uses a multitasking network to simultaneously predict salient maps, edges, and skeleton maps. Then, cross-task aggregation and cross-layer aggregation modules are used to integrate multi-level and multitasking features for the final results. SAC [13] implements a spatial attenuation context module to propagate and aggregate salient features through two rounds of recurrent translations. UDNet [27] locates pixels within and within the contours of the surrounding region using internal contour uncertainty maps, whole saliency maps, and external contour uncertainty maps. ICON-R [28] introduces three different aggregations of features, enhancement of the integrity channel, and verification of the entire SOD. EDN [29] uses an extreme down-sampling method to effectively learn the decoder's global features and Scale-Correlated Pyramid Convolution to recover local details. TSNet [30] proposes a novel bi-stream network to take full advantage of a small training set consisting of two feature backbones with different structures, achieving complementary semantical saliency fusion via the proposed gate control unit. Our previous work, SFENet [11], uses a spatial frequency enhancement (SFE) module to refine saliency features by extracting and exchanging frequency information among multiple in-stage and cross-stage feature maps. MENet [10] integrates multiple human visual systems (HVS) operations into the network structure and the loss function. Specifically, SFENet and MENet feature a two-branch decoding process that progressively refines the boundary and adaptive features.

While these techniques leverage supplementary edge information to enhance detection accuracy, predicting precise boundaries or contours in intricate scenes remains challenging. The scarcity of boundary data compared to inner region data leads to sub-optimal outcomes when edges are directly employed for supervision [17]. Moreover, boundary labels contain limited information and are prone to interference from similar textures inside large objects [1], particularly when the distinction between the background and foreground is minimal, or the objects have indistinct boundaries. Some approaches implement a strategic communication design across various levels of different branches to provide mutual reinforcement. Furthermore, error information can be shared concurrently.

2.2. *Transformer-based SOD model*

Transformers show promise for computer vision tasks, such as SOD, due to their effective modeling of long-range dependencies using self-attention [18,31]. Several transformer-based models [28,32,33,34] have been proposed for SOD that have shown promising results on benchmark datasets and outperformed CNN-based methods. Visual Saliency Transformer (VST) [32] introduced a token-based multitask structure. The decoder detects salient regions and boundaries simultaneously using task-related tokens. EBMG [33] uses the vision

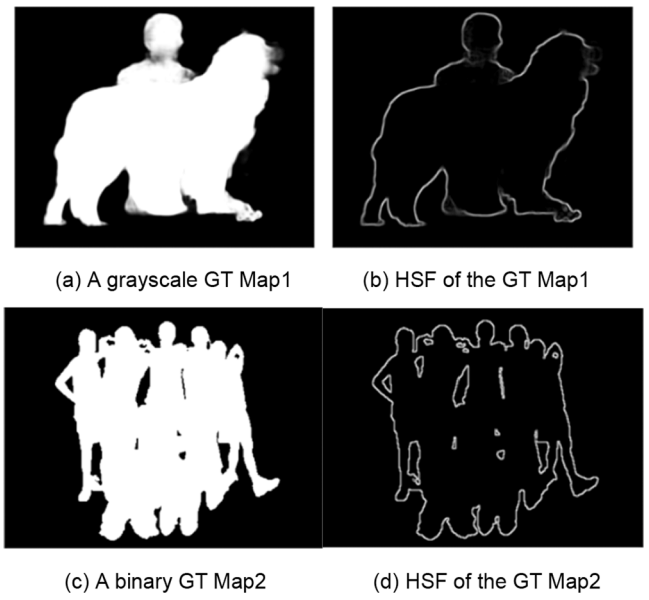

Fig. 3. Two types of GT maps and the HSF components of the GT maps computed by the Sobel edge detector.

Transformer that generates latent variables to detect salient objects depending on an informative energy-driven prior. Using global and local context generated by the Context Refinement Module, the Self-Refined Transformer (SelfReformer) [34] can guide and correct itself, and predictions are reshaped to ground truth using the pixel shift from Super-Resolution (SR). ICON-P/ICON-S [28] (based on the PVT [35] and Swin [36] backbones, respectively) incorporate three critical components to achieve integral SOD: diverse feature aggregation, enhancement of the integrity channel, and verification of the whole package.

In this work, the design of our TFGNet, encompassing the Transformer encoder, pixel-level and mask-level feature optimization, and frequency decomposition decoder components, creates a more effective and efficient model for salient object detection.

3. Methodology

This section presents a concise and efficient framework for salient object detection (SOD). Section 3.1 introduces the overall architecture of the framework, providing a detailed explanation of the salient prediction process. Section 3.2 analyzes the decomposition of the ground truth map from a frequency perspective. Section 3.3 details the structure and functionality of the network's key components. Finally, Section 3.4 introduces the novel hybrid loss function, highlighting its key characteristics.

3.1. Overall architecture

Fig. 2 shows the overall architecture of the proposed Transformer-based frequency-guided network (TFGNet). TFGNet adopts a Transformer encoder as the backbone network to extract multiscale features and a parallel two-branch decoder, which gradually refines the high-frequency boundary details and low-frequency inner regions of salient objects under the guidance of decomposed frequency supervisions.

The backbone network generates multistage multiscale feature blocks labeled B_i. Common Transformer backbone models, including Pyramid Vision Transformer (PVT) [35], Swin [36], and SwinV2 [37], each create four stages of feature blocks, denoted B_1, B_2, B_3, and B_4.

The decoding process is split into a high spatial frequency (HSF) feature learning branch and a low spatial frequency (LSF) feature learning branch. Each part includes a spatial frequency-feature enhancement (FFE) module, a pixel decoder, and a mask prediction (MP) module employing a Transformer decoder. In the FFE module, the network incrementally propagates and combines multiscale spatial frequency features derived from the backbone network. This aids in calculating per-pixel embedding and capturing fine details and local information. In contrast, the MP module functions at the mask level, segmenting the input image or feature maps into masks. These masks are categorized as salient or non-salient via the TF-decoder layer, resulting in per-mask embedding. This TF-decoder is designed to capture global

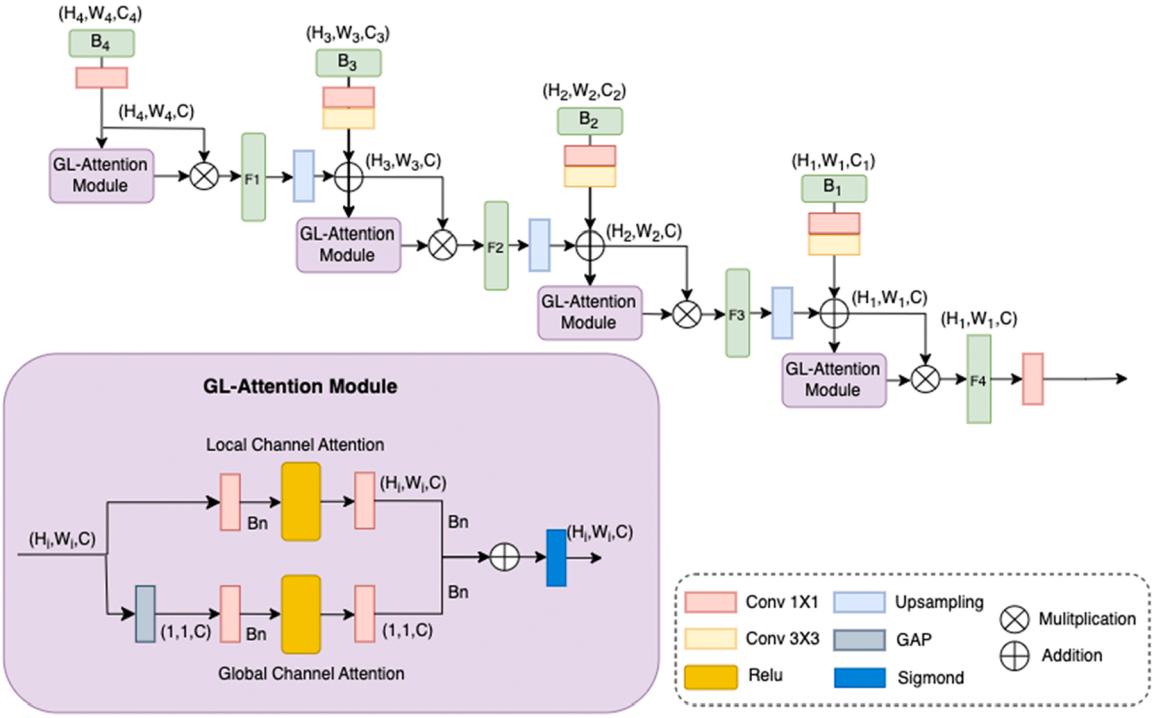

Fig. 4. FFE module structure.

Table 1
Quantitative performance comparison on the HKU-IS dataset. The top three results are red, green, and blue. 'Params' denote a model's parameter size.

No.	Method	Pub.	Backbone	Params (M)	HKU-IS				
					MAE	*MF*	*mF*	E_m	S_m
1	CPD [21]	CVPR19	Resnet-50	47.85	0.0342	0.9250	0.9047	0.9503	0.9056
2	EGNet [16]	ICCV19	Resnet-50	111.65	0.0309	0.9352	0.9122	0.9564	0.9180
3	BASNet [14]	CVPR19	Resnet-34	87.06	0.0322	0.9284	0.9113	0.9458	0.9090
4	AADFNet [12]	TCSVT20	Resnet-50	26.46	0.0255	0.9415	0.9339	0.9592	0.9190
5	GateNet [22]	ECCV20	Resnet-101	130.02	0.0320	0.9375	0.9136	0.9567	0.9195
6	MINet [23]	Tip20	VGG-16	162.37	0.0292	0.9349	0.9166	0.9600	0.9189
7	LDF [17]	CVPR20	Resnet-50	25.15	0.0275	0.9394	0.9224	0.9597	0.9196
8	U2Net [24]	PR20	RSU	87.06	0.0312	0.9352	0.9133	0.9484	0.9161
9	SGL-KRN [25]	AAAI21	Resnet-50	68.69	0.0280	0.9301	0.9154	0.9539	0.9206
10	PA-KRN [25]	AAAI21	Resnet-50	72.37	0.0271	0.9349	0.9198	0.9561	0.9230
11	HQSOD [26]	ICCV21	Resnet-50	-	0.0252	0.9428	0.9351	0.9639	0.9235
12	DCN [15]	TIP21	Resnet-50	37.95	0.0268	0.9394	0.9226	0.9624	0.9217
13	SAC [13]	TCSVT21	Resnet-101	-	0.0257	0.9416	0.9260	0.9636	0.9253
14	EDN [29]	TIP22	Resnet-50	42.84	0.0264	0.9325	0.9196	0.9548	0.9241
15	TSNet [30]	TMM22	Resnet-50+Vgg16	-	0.0266	0.9417	0.9220	0.9622	0.9223
16	SFENet [11]	IF23	Resnet-101	58.10	0.0264	0.9369	0.9283	0.9622	0.9178
17	ICON-R [28]	TPAMI23	Resnet-50	33.09	0.0289	0.9395	0.9196	0.9585	0.9202
18	MENet [10]	CVPR23	Resnet-50	27.83	0.0234	0.9483	0.9319	0.9657	0.9274
19	VST [32]	ICCV21	T2T-ViTt-14	44.63	0.0297	0.9424	0.9129	0.9597	0.9283
20	EBMG [33]	ANIPS21	Swin	118.96	0.0229	0.9466	0.9288	0.9673	0.9304
21	SelfReformer [34]	ArXiV22	PVT	90.70	0.0241	0.9474	0.9265	0.9606	0.9310
22	ICON-S [28]	TPAMI23	Swin	94.30	0.0216	0.9512	0.9331	0.9717	0.9355
23	ICON-P [28]	TPAMI23	PVT	65.68	0.0216	0.9521	0.9325	0.9698	0.9353
24	**TFGNet-B256**		**SwinV2**	**80.74**	**0.0200**	**0.9548**	**0.9382**	**0.9738**	**0.9387**
25	**TFGNet-B384**		**SwinV2**	**80.74**	**0.0179**	**0.9596**	**0.9456**	**0.9769**	**0.9441**
26	**TFGNet-L384**		**SwinV2**	**162.86**	**0.0176**	**0.9603**	**0.9472**	**0.9774**	**0.9449**

information and high-level features at the mask level. The optimized frequency feature embedding is obtained by multiplying the two embeddings together. This integration of per-pixel and per-mask embedding ensures a thorough understanding of the salient object regions, enhancing the precision and robustness of the salient features.

In the last part of TFGNet, the enhanced frequency salient features are concatenated into an integrator, which outputs a full-band salient map, denoted S. During training, S is supervised by the GT map (denote Y) by the proposed hybrid loss function. Using the hybrid loss function and integrating the HSF and LSF features, TFGNet generates accurate and robust saliency predictions for complex scenes.

3.2. Ground truth map decomposition

SOD is a binary classification task. Given a GT map Y, and we denote it by $Y = \{y_i | y_i \in [0,1]\}$. A GT map is typically grayscale, which can be binary or non-binary, as illustrated in Fig. 3.

A GT map (Y) can be decomposed into an HSF GT map (Y^H) and an LSF GT map (Y^L): $Y=Y^H+Y^L$. HSF information in a GT map exhibits the object's details, edges, and other fine-grained features. The LSF part can be computed by $Y^L=Y-Y^H$, which is the inner region of the salient object. Since a GT map is a simple binary or grayscale image with no background or foreground noise, a gradient-based edge detector [5], such as Prewitt, Roberts, and Sobel, is usually used to extract Y^H by convolving the image with different size kernels. Specifically, Sobel is known to be more effective in detecting accurate edges than the Prewitt and Roberts operators. We use Sobel in our implementation, as shown in Fig. 3.

In contrast to some edge-aware methods, such as EGNet and MENet, which utilize edge detectors to extract salient edges from GT maps as prior information, our approach takes a different perspective based on the frequency decomposition of the GT maps. This aligns with the core ideas of our frequency decomposition and ensures theoretical consistency with the network structure.

3.3. Salient features learning

We combine a pixel-level decoder and a Transformer decoder to optimize salient features.

Pixel Decoder: It is a multiscale frequency feature enhancement (FFE) module, as shown in Fig. 4. Enhancing features begins with feature B_4. Then, the three remaining feature blocks (i.e., B_3, B_2, and B_1) are aggregated sequentially, proceeding from small to large resolutions. We introduce a Global and Local Attention (GL-Attention) module to the FFE to tackle the challenge of recognizing and detecting objects in extreme scale variations. GL-Attention is based on multiscale channel attention (MS-CAM) [9]. By incorporating GL-Attention into the FFE, the network becomes more adept at handling objects at different scales and capturing essential contextual information, leading to accurate pixel-wise classification. Starting from the feature block B_4, a $[1 \times 1]$ convolutional layer compresses the number of channels to C. In the implementation, we let $C= 256$. Next, we apply $[1 \times 1]$ followed by $[3 \times 3]$ convolutional layers to the feature block B_i to make these feature blocks the same channel value C, ensuring consistency in channel dimensions. Then, in the GL-Attention module, a feature block B_i is refined synchronously by a global and local-channel attention processes.

Then, in the GL-Attention module (denoted *glAtten* () in the following

Table 2
Quantitative performance comparison on the DUT-OMRON and the DUTS-TE datasets. The top three results are highlighted in bold red, green, and blue.

No.	Method	Pub.	DUT-OMRON					DUTS-TE				
			MAE	*MF*	*mF*	E_m	S_m	*MAE*	*MF*	*mF*	E_m	S_m
1	CPD [21]	CVPR19	0.0560	0.7966	0.7807	0.8726	0.8248	0.0429	0.8649	0.8431	0.9009	0.8691
2	EGNet [16]	ICCV19	0.0528	0.8155	0.7942	0.8738	0.8412	0.0386	0.8880	0.8597	0.9040	0.8873
3	BASNet [14]	CVPR19	0.0565	0.8053	0.7906	0.8691	0.8362	0.0472	0.8589	0.8416	0.8790	0.8660
4	AADFNet [12]	TCSVT20	0.0488	0.8143	0.8050	0.8744	0.8389	0.0314	0.8993	0.8911	0.9225	0.8914
5	GateNet [22]	ECCV20	0.0547	0.8210	0.7944	0.8736	0.8449	0.0380	0.8919	0.8615	0.9075	0.8910
6	MINet [23]	TIP20	0.0559	0.8098	0.7911	0.8734	0.8329	0.0373	0.8833	0.8597	0.9132	0.8842
7	LDF [17]	CVPR20	0.0517	0.8199	0.8015	0.8814	0.8392	0.0336	0.8968	0.8779	0.9232	0.8924
8	U2Net [24]	PR20	0.0544	0.8226	0.8023	0.8716	0.8467	0.0443	0.8719	0.8479	0.8840	0.8738
9	SGL-KRN [25]	AAAI21	0.0492	0.7961	0.7830	0.8783	0.8464	0.0337	0.8833	0.8649	0.9311	0.8929
10	PA-KRN [25]	AAAI21	0.0496	0.8101	0.7956	0.8880	0.8533	0.0328	0.8954	0.8761	0.9353	0.9005
11	HQSOD [26]	ICCV21	0.0509	0.8160	0.8091	0.8805	0.8404	0.0326	0.8932	0.8867	0.9271	0.8920
12	DCN [15]	TIP21	0.0511	0.8230	0.8056	0.8853	0.8455	0.0348	0.8935	0.8742	0.9180	0.8917
13	SAC [13]	TCSVT21	0.0523	0.8287	0.8092	0.8833	0.8487	0.0339	0.8944	0.8732	0.9208	0.8957
14	EDN [29]	TIP22	0.0497	0.7992	0.7880	0.8854	0.8496	0.0353	0.894	0.8775	0.9222	0.8926
15	TSNet [30]	TMM22	0.0510	0.8312	0.8083	0.8871	0.8503	0.0319	0.9025	0.8775	0.9249	0.8995
16	SFENet [11]	IF23	0.0490	0.8180	0.8098	0.8831	0.8396	0.0334	0.8856	0.8768	0.9265	0.8847
17	ICON [28]	TPAMI23	0.0569	0.8254	0.8013	0.8791	0.8445	0.0370	0.8917	0.8665	0.9142	0.8889
18	MENet [10]	CVPR23	0.0450	0.8337	0.8178	0.8911	0.8496	0.0281	0.9123	0.8930	0.9368	0.9049
19	VST [32]	ICCV21	0.0582	0.8245	0.7967	0.8718	0.8503	0.0372	0.8898	0.8579	0.9153	0.8963
20	EBMG [33]	ANIPS21	0.0505	0.8386	0.8179	0.8951	0.8584	0.0288	0.9091	0.8863	0.9331	0.9088
21	SelfReformer [34]	ArXiV22	**0.0433**	0.8367	0.8189	0.8928	0.8608	0.0266	0.9155	0.8921	0.9210	0.9111
22	ICON-S [28]	TPAMI23	**0.0426**	0.8546	0.8350	0.9073	0.8693	0.0242	0.9196	0.8998	0.9470	0.9171
23	ICON-P [28]	TPAMI23	0.0468	0.8519	0.8228	0.8951	0.8654	0.0255	0.9218	0.8932	0.9386	0.9173
24	**TFGNet-B256**		0.0438	**0.8557**	**0.8388**	**0.9074**	**0.8717**	**0.0228**	**0.9242**	**0.9062**	**0.9500**	**0.9212**
25	**TFGNet-B38**		0.0441	**0.8605**	**0.8462**	**0.9087**	**0.8758**	**0.0208**	**0.9340**	**0.9198**	**0.9545**	**0.9312**
26	**TFGNet-L384**		**0.0402**	**0.8614**	**0.8488**	**0.9118**	**0.8799**	**0.0193**	**0.9375**	**0.9238**	**0.9566**	**0.9340**

equations), a feature block B_i is refined by a global channel attention process (denoted *gAtten* ()) and a local channel attention process (denoted *lAtten*()), synchronously, in the following equations.

$$glAtten(B_i) = \sigma[gAtten(B_i) \oplus lAtten(B_i)]$$

$$= \sigma[Conv_1(ReLU(Bn(Conv_1(GAP(B_i))))) \oplus Conv_1], \quad (1)$$

where *ReLU* () is the Rectified Linear Unit, GAP() is the global average pooling, σ[] is the Sigmond Activation function, *Bn*() is the Batch Normalization, and $\oplus$ is the element-wise addition.

Let

$$\widehat{B}_i = Up(F_{i+1}) \oplus Conv_3(Conv_1(B_i)) \quad (i = 3, 2, 1), \quad (2)$$

where *Up*() is the up-sampling operation, then the FFE refine process can be concluded by the following four steps.

$$F_4 = glAtten(Conv_1(B_4))Conv_1(B_4), \quad (3)$$

$$F_3 = glAtten(\widehat{B}_3) \otimes \widehat{B}_3, \quad (4)$$

$$F_2 = glAtten(\widehat{B}_2) \otimes \widehat{B}_2, \quad (5)$$

$$F_1 = glAtten(\widehat{B}_1) \otimes \widehat{B}_1. \quad (6)$$

Here, the symbol $\otimes$ represents element-wise multiplication.

Transformer Decoder:

It is a mask prediction (MP) module based on a Transformer (TF). The MP divides the input into segments, also known as masks, and classifies them into salient and non-salient categories to produce segment embeddings by a TF-decoder layer. These segment embeddings are then processed by a Multi-Layer Perception (MLP) with two hidden layers to convert into N mask embeddings. Segments with the same category label are merged, yielding the final mask embeddings.

Next, the optimized HSF and LSF salient features are obtained through a product operation between the feature embeddings of the pixel decoder and the TF-decoder. The outputs of the HSF and LSF feature learning branches are HF and LF, the same size as B_1. This Pixel-Decoder and TF-decoder optimization process enables TFGNet to learn and efficiently generate accurate saliency predictions for complex scenes.

Learning Strategy:

We use different loss settings and learning strategies for HSF and LSF feature learning branches. Considering the complexity of the network architecture in the HSF feature learning branch, we apply the BCE loss to *HF*. Regarding the LSF feature learning branch, both the pixel-decoder and the TF-decoder employ an adaptive learning strategy. Consequently, the *LF* is not supervised. This strategy has been experimentally proven optimal, helping the network refine high- and low-frequency features more effectively.

Integration:

Since the output of HSF and LSF feature learning branches (i.e., *HF* and *LF*) share the exact resolution and channel number, we concatenate them. Next, a [3 × 3] convolutional layer is applied to this concatenated map, obtaining the full-band saliency map. Then a [1 × 1] convolutional layer and an up-sampling operation are used to generate the final saliency map S supervised by the proposed hybrid loss in Section 3.4.

Table 3
Quantitative performance comparison on the PASCAL-S the ECSSD datasets. The top three results are highlighted in bold red, green, and blue.

No.	Method	Pub.	PASCAL-S					ECSSD				
			MAE	*MF*	*mF*	E_m	S_m	*MAE*	*MF*	*mF*	E_m	S_m
1	CPD [21]	CVPR19	0.0706	0.8595	0.8414	0.8873	0.8484	0.0371	0.9393	0.9244	0.9494	0.9182
2	EGNet [16]	ICCV19	0.0740	0.8653	0.8437	0.8772	0.8521	0.0374	0.9474	0.9288	0.9469	0.9246
3	BASNet [14]	CVPR19	0.0758	0.8539	0.8344	0.8527	0.8380	0.0370	0.9425	0.9274	0.9210	0.9163
4	AADFNet [12]	TCSVT20	0.0550	0.8797	0.8677	0.9051	0.8658	0.0280	0.9543	0.9478	0.9529	0.9299
5	GateNet [22]	ECCV20	0.0668	0.8702	0.8468	0.8924	0.8622	0.0357	0.9508	0.9301	0.9501	0.9302
6	MINet [23]	TIP20	0.0643	0.8665	0.8461	0.8981	0.8563	0.0342	0.9475	0.9309	0.9532	0.9250
7	LDF [17]	CVPR20	0.0596	0.8741	0.8577	0.9048	0.863	0.0335	0.9501	0.9379	0.9509	0.9245
8	U2Net [24]	PR20	0.0740	0.8592	0.8386	0.8500	0.8444	0.0330	0.9510	0.9325	0.9251	0.9276
9	SGL-KRN [25]	AAAI21	0.0678	0.8502	0.8373	0.8941	0.8556	0.0360	0.9368	0.9241	0.9462	0.9231
10	PA-KRN [25]	AAAI21	0.0665	0.8530	0.8388	0.8964	0.8578	0.0323	0.9425	0.9301	0.9503	0.9278
11	HQSOD [26]	ICCV21	0.0597	0.8798	0.8698	0.9074	0.8603	0.0294	0.9520	0.9456	0.9600	0.9276
12	DCN [15]	TIP21	0.0618	0.8723	0.8543	0.9017	0.8612	0.0315	0.9524	0.9396	0.9575	0.9282
13	SAC [13]	TCSVT21	0.0622	0.8772	0.8585	0.9022	0.8656	0.0309	0.9512	0.9376	0.9586	0.9312
14	EDN [29]	TIP22	0.0617	0.8600	0.8489	0.9015	0.8646	0.0320	0.9410	0.9304	0.9508	0.9267
15	TSNet [30]	TMM22	0.0573	0.8800	0.8599	0.9064	0.8684	0.0297	0.9532	0.9391	0.9561	0.9322
16	SFENet [11]	IF23	0.0594	0.8715	0.8604	0.9092	0.8567	0.0314	0.9457	0.9388	0.9577	0.9234
17	ICON [28]	TPAMI23	0.0644	0.8757	0.8514	0.8931	0.8611	0.0318	0.9503	0.9336	0.9543	0.9290
18	MENet [10]	CVPR23	0.0535	0.8896	0.8701	0.9132	0.8721	0.0307	0.9549	0.9422	0.9544	0.9279
19	VST [32]	ICCV21	0.0620	0.8755	0.8457	0.9024	0.8716	0.0337	0.9507	0.9258	0.9571	0.9323
20	EBMG [33]	ANIPS21	0.0542	0.8866	0.8659	0.9070	0.8765	0.0232	0.9591	0.9452	0.9632	0.9416
21	SelfReformer [34]	ArXiV22	0.0510	0.8943	0.8736	0.8825	0.8809	0.0273	0.9577	0.9414	0.9361	0.9356
22	ICON-S [28]	TPAMI23	0.0484	0.8961	0.8767	0.9237	0.8849	0.0235	0.9608	0.9458	0.9669	0.9414
23	ICON-P [28]	TPAMI23	0.0510	0.8927	0.8690	0.9145	0.8819	0.0240	0.9594	0.9432	0.9624	0.9401
24	**TFGNet-B256**		**0.0468**	**0.9000**	**0.8806**	**0.9284**	**0.8874**	**0.0214**	**0.9633**	**0.9496**	**0.9714**	**0.9452**
25	**TFGNet-B38**		**0.0471**	**0.9018**	**0.8843**	**0.9304**	**0.8887**	**0.1980**	**0.9673**	**0.9559**	**0.9728**	**0.9493**
26	**TFGNet-L384**		**0.0442**	**0.9038**	**0.8874**	**0.9332**	**0.8919**	**0.0186**	**0.9679**	**0.9568**	**0.9741**	**0.9510**

3.4. Hybrid loss

There are two supervisions within the network. One ($\mathscr{L}^H$) is for the predicted HSF feature, and the other ($\mathscr{L}^I$) is applied to the final full-band predicted map (S). No supervision is applied to the LSF feature learning branch. Thus, the training loss $\mathscr{L}$ can be represented by Eq. (7).

$$\mathscr{L} = \alpha_1 \mathscr{L}^H + \alpha_2 \mathscr{L}^I \tag{7}$$

We set α_1=α_2= 0.5 in implementation.

We use the BCE loss [19] in $\mathscr{L}^H$ defined by Eq. (8).

$$\mathscr{L}^H = -\sum \left(Y^H \log S^H + (1 - Y^H) \log(1 - S^H) \right), \tag{8}$$

where S^H and Y^H are the HSF predicted and GT maps, respectively.

We propose to use a hybrid loss $\mathscr{L}^I$, defined by Eq. (9).

$$\mathscr{L}^I = \theta_1 \mathscr{L}^I_{BCE} + \theta_2 \mathscr{L}^I_{IoU} + \theta_3 \mathscr{L}^I_{Hist}, \tag{9}$$

where $\mathscr{L}^I_{BCE}$, $\mathscr{L}^I_{IoU}$, and $\mathscr{L}^I_{Hist}$ denote the BCE, the IoU [20], and the proposed Hist losses, respectively. Empirically, we set θ_1=θ_2=θ_3= 1.

The BCE loss is defined by Eq. (10).

$$\mathscr{L}^I_{BCE} = -\sum \left(Y \log S + (1 - Y) \log(1 - S) \right), \tag{10}$$

where Y is the GT map.

IoU loss is defined in Eq. (11). It measures global structural similarity by evaluating the overlap between predicted and ground-truth saliency regions.

$$\mathscr{L}^I_{IoU} = 1 - \frac{\sum (Y * S)}{\sum (Y + S - Y * S)}. \tag{11}$$

Here, we suggest using a histogram to evaluate the similarity of the global pixel-intensity frequency distributions of Y and S. The histogram is the first-order data statistic representing the global signal intensity distribution. The histogram loss is represented by Eq. (12).

$$\mathscr{L}^I_{Hist} = \frac{\sum_{i=1}^{N_{bin}} |H_s(i) - H_y(i)|}{N_{bin}}, \tag{12}$$

where $H_s()$ and $H_y()$ are the histograms of S and Y, and N_{bin} is the total bin number in the histogram.

The histogram lacks spatial information. It remains zero when Y and S have different shapes but identical histograms. Incorporating a structure-ware loss function like IoU can complement such limitations. Therefore, combining the BCE, IoU, and the histogram-based losses can provide a comprehensive training process for the TFGNet model.

Section 4 explores how these losses impact the overall network performance.

4. Experiments and discussion

This section provides an evaluation of TFGNet using pre-trained Transformer backbones across various scales. We first outline the experimental setup in Section 4.1, detailing the training strategies and offering a brief overview of the test datasets. Next, in Section 4.2, we

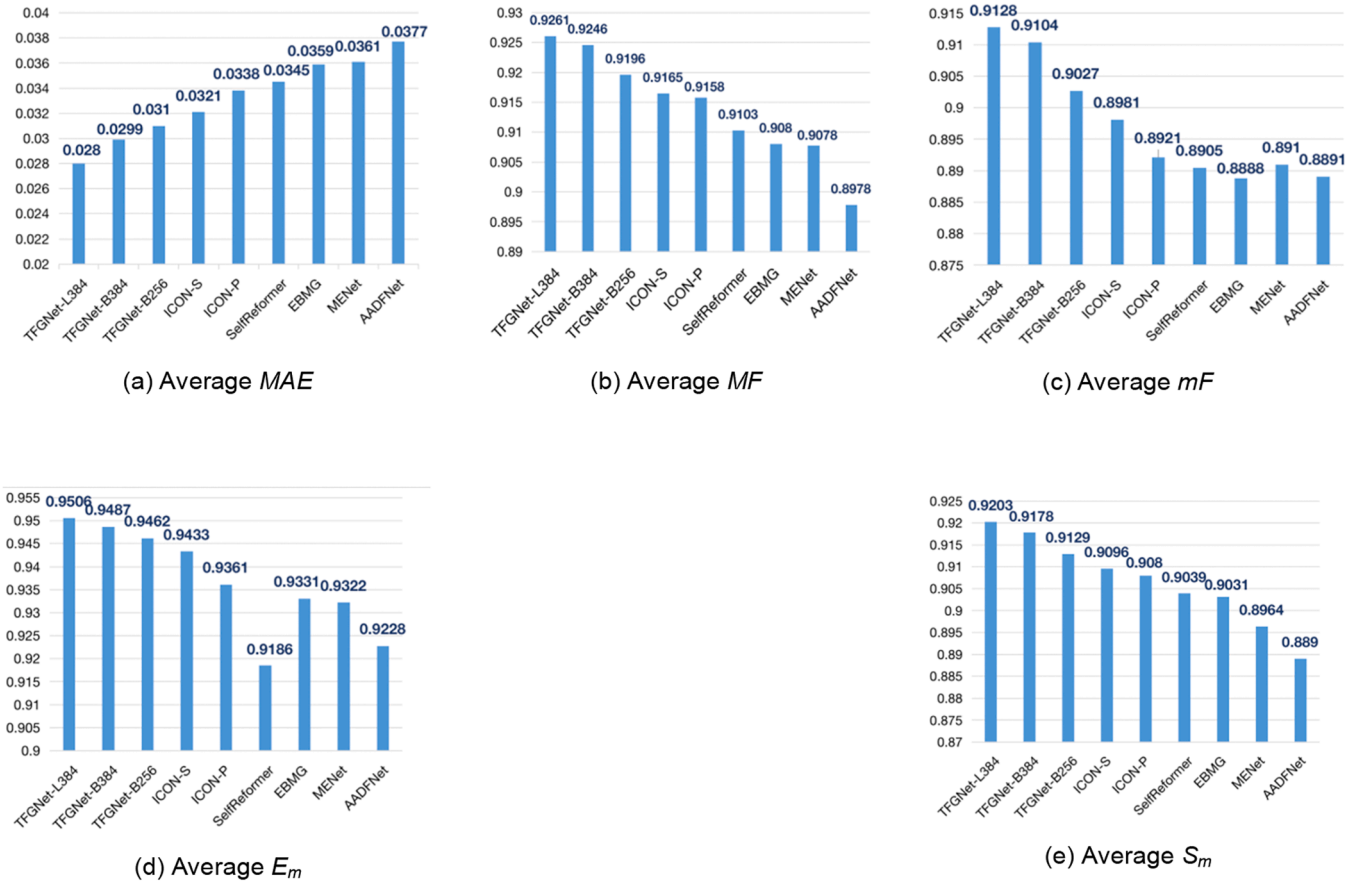

Fig. 5. Average scores of five metrics for five datasets.

present the evaluation metrics. Section 4.3 compares our proposed model with other approaches, discussing our advantages and limitations. Finally, we conduct a systematic analysis of how different design choices influence the performance of our model.

4.1. Training and testing strategies

We use the ImageNet [38] pre-train and the DUTS-TR [39] dataset to fine-tune the proposed TFGNet. The data augmentation includes horizontal flips and random crops. In training, TFGNet uses SwinV2 [37] as the backbone network. The maximum learning rate of the backbone is 0.0001, and 0.001 for other parts. The momentum is 0.9, and the weight decay is 0.001. The 'poly' learning rate strategy is also adopted. The maximum iteration number is defaulted to 99. The batch size is 12.

Our network is built on PyTorch 1.12. Both training and testing are conducted a computer server with AMD EPYC 7742 (2.25 GHz) and an NVIDIA A100 GPU (with 40 GB of memory).

Five popular pixel-wise annotated SOD benchmark datasets are used for evaluation.

DUTS-TE [39] is the subset of the DUTS dataset [39], comprising 5019 images with highly challenging scenarios.

DUT-OMRON [40] comprises 5168 images of diverse objects against complex backgrounds, ranging from 89 to 401. The objects in this dataset are diverse and numerous, and the backgrounds of most samples are complex.

HKU-IS [41] is composed of 4447 images low-contrast images. Each image satisfies at least one of the three following criteria: (*i*) multiple scattered salient objects, (*ii*) at least one prominent object in the image boundary, and (*iii*) apparent similarity to backgrounds.

ECSSD [42] comprises 1000 semantically meaningful images with complicated structures and complex backgrounds.

PASCAL-S [43] has 850 images. It is recognized for having less bias compared to many other salient datasets.

4.2. Evaluation criteria

Mean Absolute Error [44], maximum F-measure, mean F-measure [45], Enhanced-alignment Measure [46], and S-measure [47] are used to evaluate SOD models.

Mean Absolute Error (*MAE*) evaluates the absolute error between the normalized ground truth map (G) and the predicted map (S) at the average pixel level. It can be formulated in Eq.(13).

$$MAE = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} (G - S), \tag{13}$$

where W and H denote input image dimensions. A smaller *MAE* value is better.

F-measure (F_β) is the weighted harmonic mean of Precision and Recall; it is a comprehensive measurement, with a larger value indicating better performance. It is defined in Eq. (14).

$$F_\beta = \frac{(1 + \beta^2) \times Precision \times Recall}{\beta^2 \times Precision + Recall}, \tag{14}$$

We report maximal F_β (denoted *MF*) or mean F_β (denoted *mF*) in experiments.

Enhanced-alignment Measure (E_m) is a comprehensive evaluation index that combines image-level statistics with local-pixel matching information by Eq. (15).

$$E_m = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} S_\varnothing(i, j), \tag{15}$$

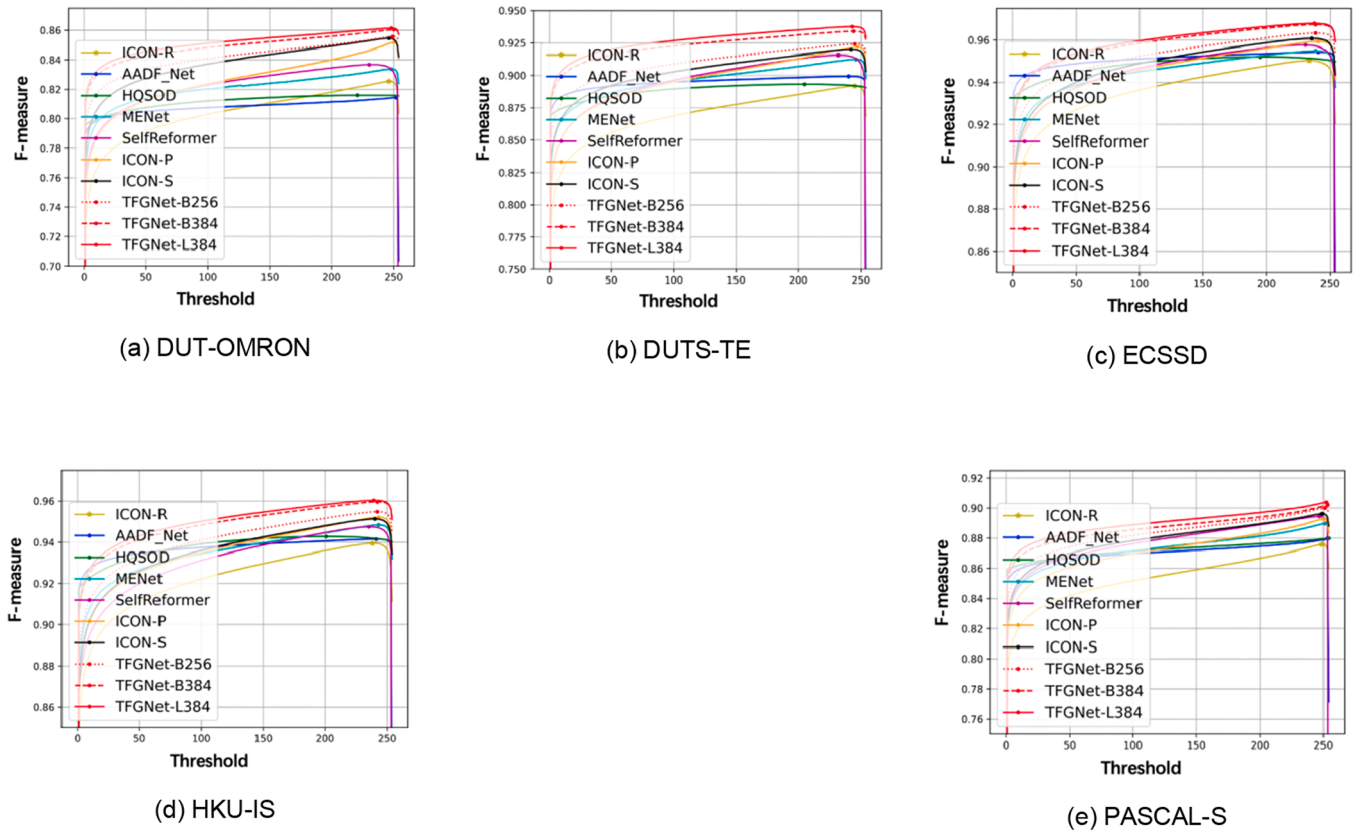

Fig. 6. Fm-curves comparison.

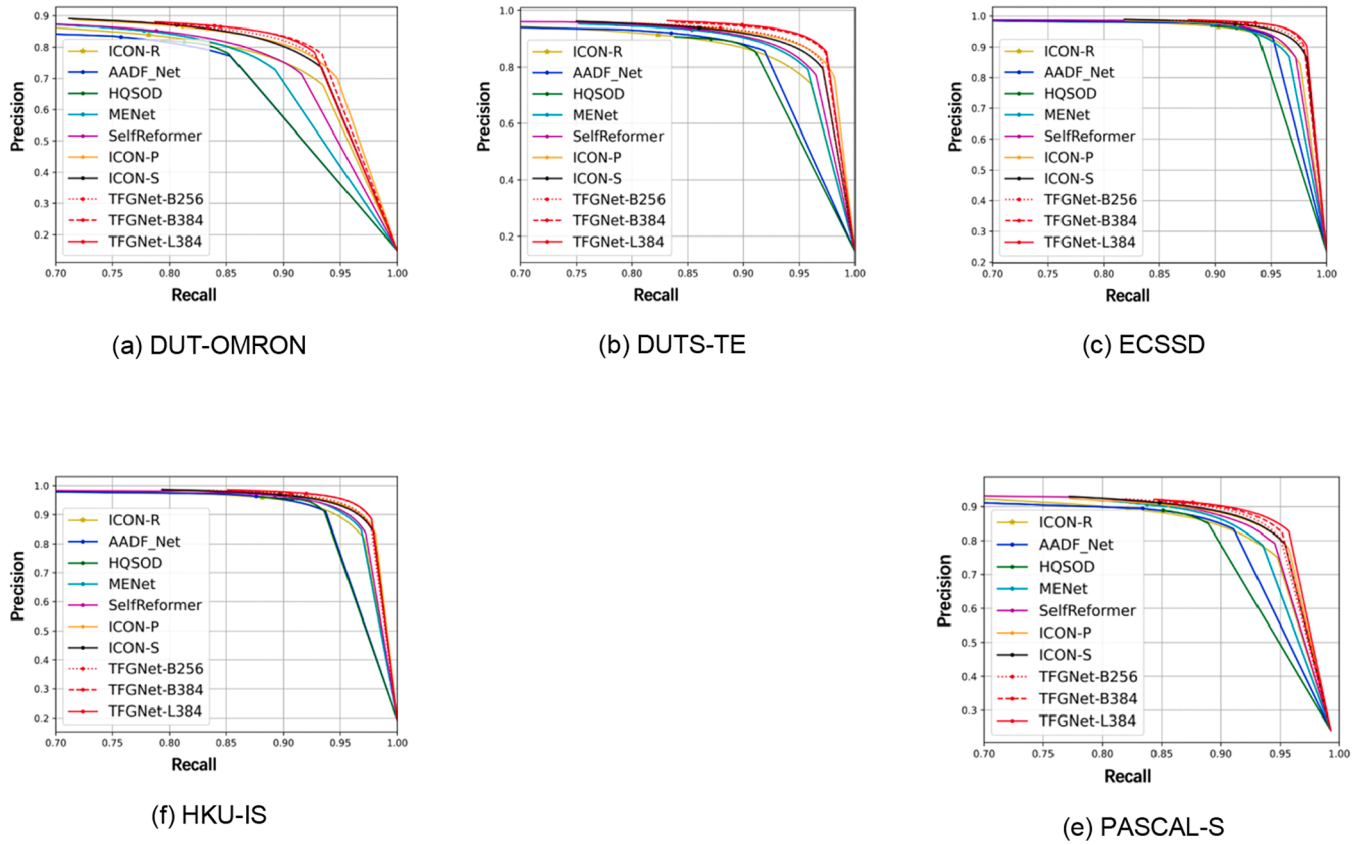

Fig. 7. PR-curves comparison.

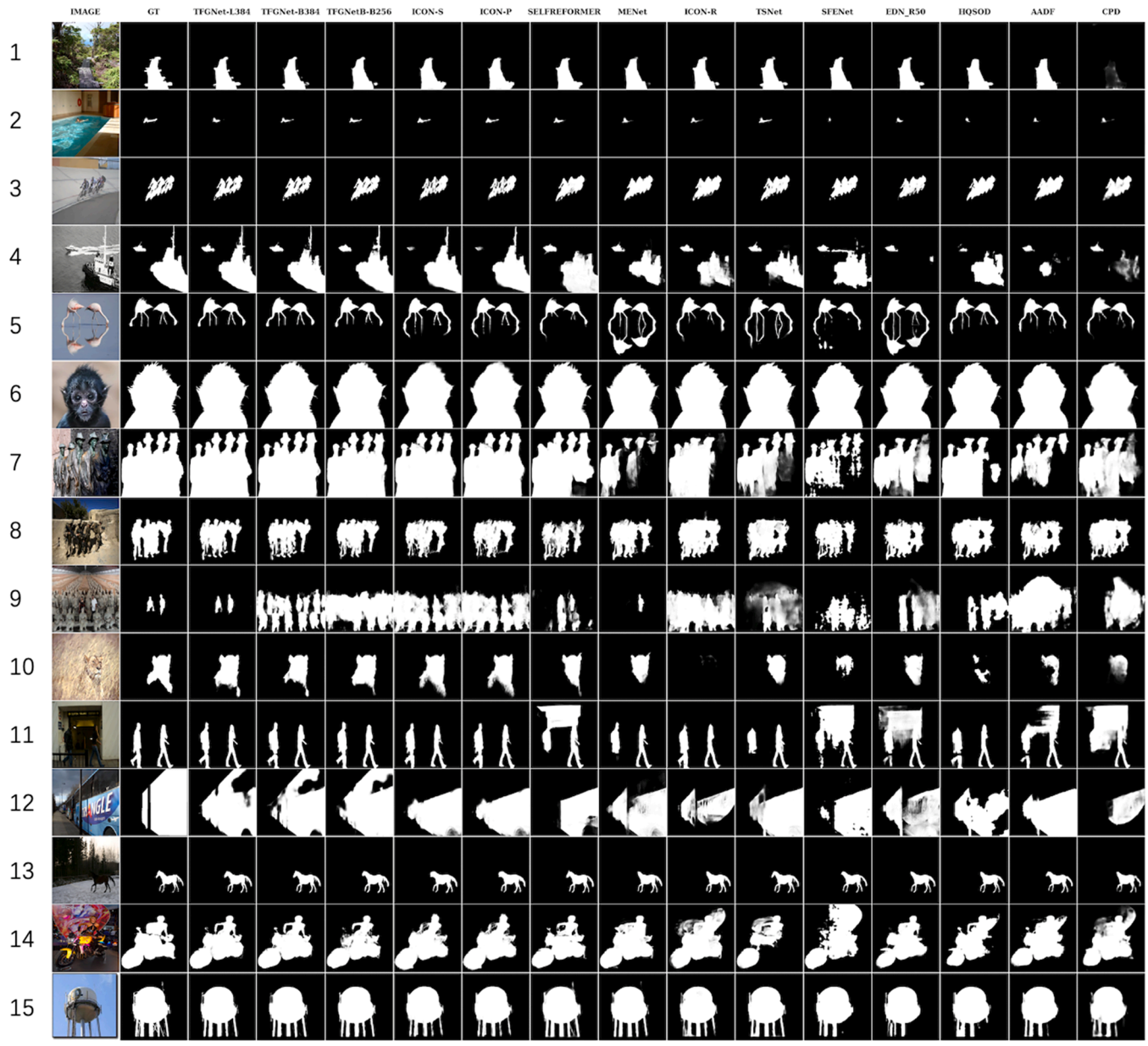

Fig. 8. Qualitative performance comparison for complex scene (Part 1).

where $S_{\varnothing}$ the enhanced alignment matrix reflects the correlation between S and G after subtracting their global means, respectively. E_m considers global means of the image and local pixel matching simultaneously.

Structure-measure (E_m) measures how similar the predicted map is to the GT map by Eq. (16).

$$S_m = \gamma S_o + (1-\gamma) S_\gamma. \tag{16}$$

Here, S_o and S_γ denote region-specific and object-specific structural similarities, respectively, with γ typically set to 0.5. S_m as a structure-wise measure complements pixel-wise error.

Precision-recall (PR) curves and F-measure curves are plotted to demonstrate overall performance [1]. A Precision-Recall (PR) curve is generated from a series of PR pairs by adjusting the threshold in the range [0,1]. The F-measure is a harmonic mean of precision and recall to assess the performance of the classification model.

4.3. Quantitative and qualitative comparison

We report three TFGNet configurations using three SwinV2 backbones, denoted as TFGNet-B256, TFGNet-B384, and TFGNet-L384, to show the effectiveness of TFGNet in different settings. Here, `B′ represents the SwinV2-Base model, 'L' means the SwinV2-Large model, and '256' and '384' indicate the input image size [256 × 256] and [384 × 384].

The comparison models include: (*i*) CNN-based models: CPD [21], EGNet [16], BASNet [14], AADFNet [12], GateNet [22], MINet [23], LDF [17], U2Net [24], PA-KRN [25], SGL-KRN [25], HQSOD [26], DCN [15], SAC [13], EDN [29], TSNet [30], SFENet[11], ICON-R [28], UDNet [27], and MENet [10]; (*ii*) Transformer-based models: VST [32], EBMG [33], SelfReformer [34], ICON-P [28], and ICON-S [28].

4.3.1. Quantitative performance comparison

Tables 1–3 illustrate the quantitative comparison results. Fig. 5 illustrates the average values of five metrics in the five datasets. Figs. 6–7 compare the Fm and PR curves, respectively. For a more comprehensive

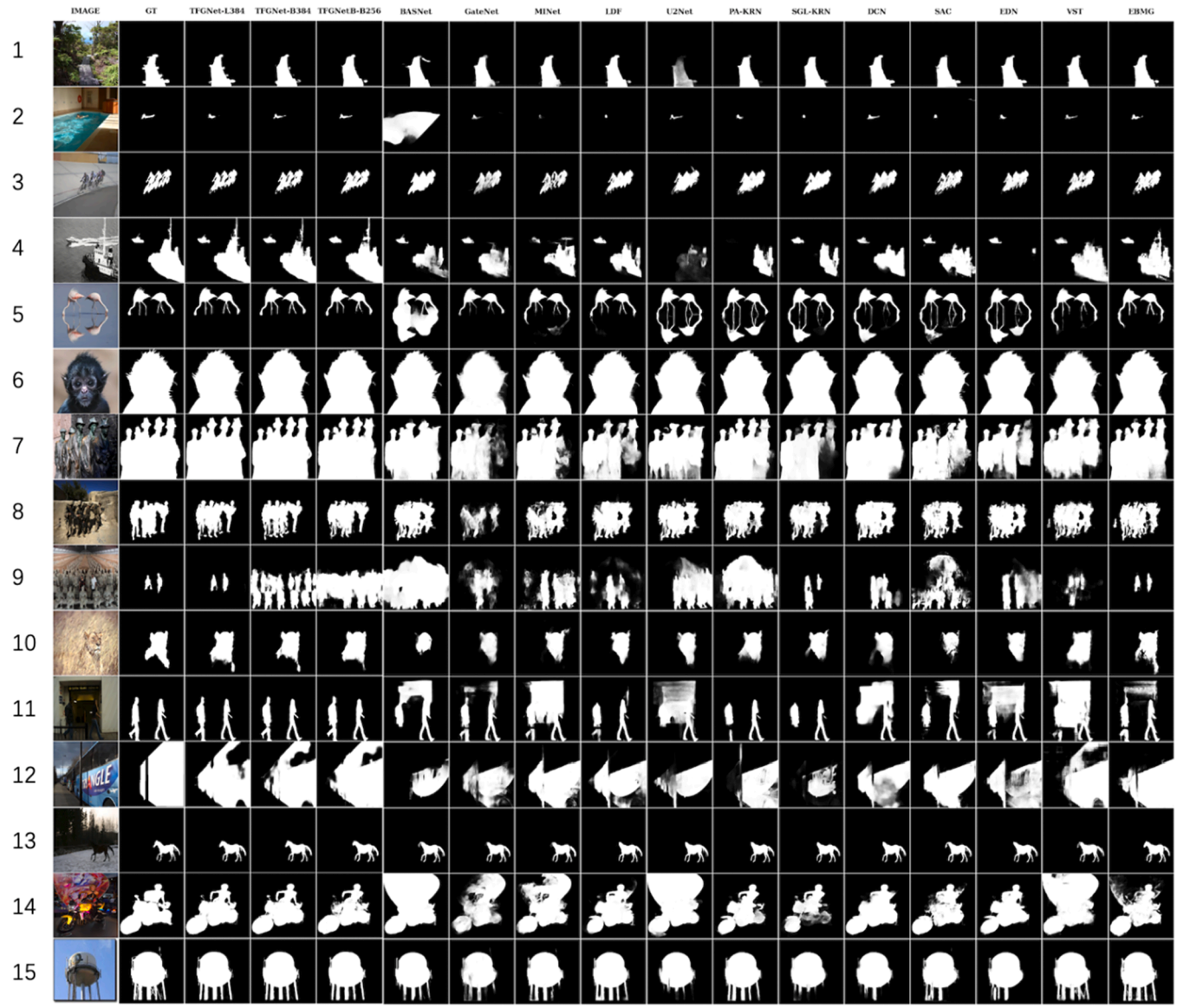

Fig. 9. Qualitative performance comparison for complex scene (Part 2).

Table 4
Quantitative performance comparison on USOD10K. The top three results are bold red, green, and blue [49].

No.	Method	Pub.	*MAE*	*MF*	*mF*	E_m	S_m
1	BASNet [14]	CVPR19	0.0352	0.8996	0.8809	0.9290	0.8937
2	MINet [23]	TIP20	0.0287	0.9219	0.9007	0.9426	0.9106
3	LDF [17]	CVPR20	0.0260	0.9294	0.9088	0.9472	0.9136
4	SGL-KRN [25]	AAAI21	0.0237	0.9338	0.9170	0.9554	0.9214
5	VST [32]	ICCV21	0.0347	0.8976	0.8557	0.9116	0.8916
6	SVAM-Net [49]	RSS22	0.0915	0.7150	0.6891	0.7636	0.7465
7	TC-USOD [48]	TIP23	0.0201	0.9236	-	-	**0.9215**
8	**TFGNet-B256**		**0.0200**	**0.9374**	**0.9248**	**0.9662**	**0.9290**
9	**TFGNet-B384**		**0.0178**	**0.9441**	**0.9319**	**0.9710**	**0.9347**
10	**TFGNet-L384**		**0.0173**	**0.9431**	**0.9316**	**0.9711**	**0.9347**

Fig. 10. Failure cases (Part 1).

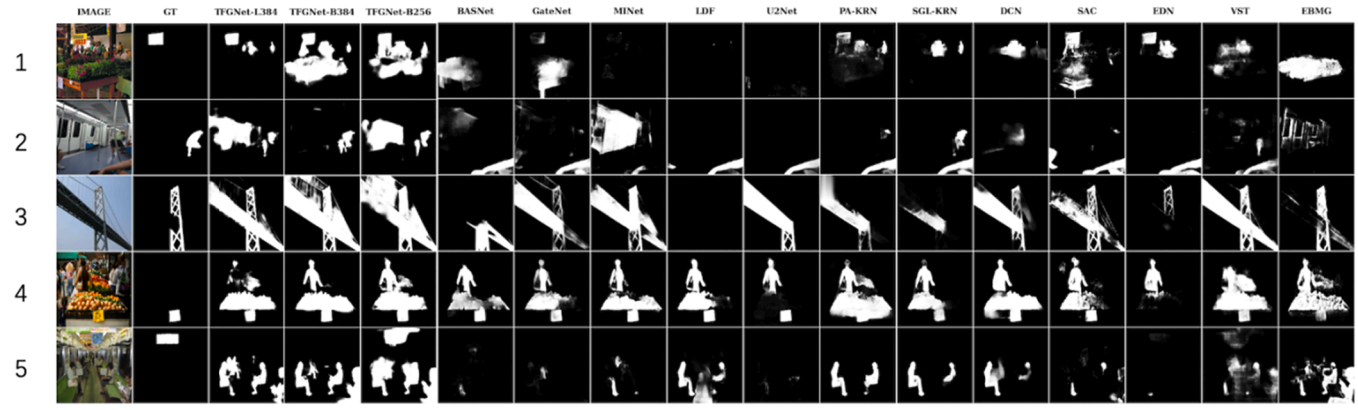

Fig. 11. Failure cases (Part 2).

comparison, we rank all models across backbones.

The comparison demonstrates that TFGNet-L384 outperforms other models across various metrics and datasets. TFGNet-L384 also has better F-measure and PR curves than other approaches, as depicted in Fig. 6 and Fig. 7. TFGNet-L384 is stable across all five datasets. Furthermore, both TFGNet-B256 and TFGNet-B384 outperform other methods overall on the five datasets, reinforcing the statistical evidence of the superiority of TFGNet architecture.

Table 1 indicates that CNN-based SOD models, such as ResNet 50, generally have fewer parameters than transformer-based SOD models, but they tend to exhibit lower accuracy overall. TFGNet-L384 has more parameters than other transformer models like SelfReformer-TF, ICON-P, and ICON-S. Yet, it outperforms them across five datasets, as illustrated in Fig. 5. In our experimental framework, TFGNet's processing speed is considered satisfactory. TFGNet-L384 processes a [384 × 384] image at a rate of 20 frames per second (FPS).

Given that the primary focus of this work is accuracy, TFGNet-L384 is a strong contender. Additionally, the other two models—TFGNet-B384 and TFGNet-B256—also outperform their transformer counterparts in precision while maintaining a comparable parameter count.

4.3.2. Qualitative performance comparison

We select 15 challenging scenes for comparison, as shown in Fig. 8 and Fig. 9. TFGNet has superior comprehensive performance. The predicted results are given as a probability. TFGNet achieves the most precise detection results overall, especially in terms of high integrity (e.g., rows 1, 8, 13, 14), correct localization (e.g., Rows 2–5, 7, 9), low-contrast backgrounds (e.g., Rows 10, 11, and 13), and precise boundaries (e.g., Rows 2, 10, 13, and 15). In contrast, other models, such as ICON-S, ICON-P, and SelfReformer, have less integrity and accurate boundary predictions.

4.4. Underwater SOD task

We examine how TFGNet performs in complex marine scenarios using an underwater SOD dataset: USOD10K [48]. The USOD10K has 10,255 images in 12 different aquatic scenes with 7:2:1 split of training, validation, and testing data. The results, presented in Table 4, indicate that TFGNet can be applied directly to underwater SOD without any degradation in performance.

4.5. Limitation

Figs. 10 and 11 illustrate several representative instances where TFGNet misidentified other foreground elements as salient objects. This misclassification primarily stems from the inherent ambiguity and subjective biases found within the annotations of complex scenes [2,30]. These challenging cases constitute only a minor portion of the dataset but pose significant detection difficulties for the model. It is important to note that such issues are not exclusive to TFGNet; other methods, including ICON, SelfReformer, VST, and EBMG, similarly encounter hurdles when processing these ambiguous situations.

For example, in Row 5, with the exception of TFG-B256, none of the other models identify the advertising sign hanging in the middle of the carriage as a salient object. In Row 1, TFGNet-L384 and TFGNet-B384 demonstrate sharper delineation of the advertising sign compared to ICON-S, ICON-P, SAC, and EDN. Similarly, in Row 3, TFGNet-L384 exhibits a more detailed and precise representation of the bridge's column

Table 5
Ablation study for loss. The top two results are bold red and green.

Dataset	No.	$\mathcal{L}^H$	$\mathcal{L}^L$	$\mathcal{L}^I$	MAE	MF	mF	E_m	S_m
DUT-OMRON	1			$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	0.0444	0.8592	0.8424	0.9079	0.8744
	2			$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	0.0431	0.8605	0.8443	0.9094	0.8757
	3	✓	✓	$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	0.0432	0.8610	0.8452	0.9107	0.8760
	4	✓	✓	$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	0.0424	**0.8627**	0.8464	0.9099	0.8765
	5	✓		$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	**0.0398**	**0.8638**	**0.8494**	**0.9129**	**0.8778**
	6	✓		$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	**0.0402**	0.8614	**0.8488**	**0.9118**	**0.8799**
DUTS-TE	1			$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	0.0227	0.9296	0.9124	0.9511	0.9260
	2			$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	0.0224	0.9299	0.9135	0.9517	0.9267
	3	✓	✓	$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	0.0221	0.9309	0.9145	0.9520	0.9270
	4	✓	✓	$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	0.0205	0.9341	0.9175	0.9542	0.9304
	5	✓		$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	**0.0198**	**0.9357**	**0.9197**	**0.9556**	**0.9311**
	6	✓		$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	**0.0193**	**0.9375**	**0.9238**	**0.9566**	**0.9340**
HKU-IS	1			$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	0.0184	0.9580	0.9426	0.9758	0.9427
	2			$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	0.0182	0.9582	0.9435	0.9761	0.9431
	3	✓	✓	$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	0.0183	0.9582	0.9441	0.9758	0.9429
	4	✓	✓	$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	**0.0181**	**0.9590**	**0.9449**	**0.9765**	**0.9436**
	5	✓		$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	0.0183	**0.9590**	**0.9449**	0.9760	0.9428
	6	✓		$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	**0.0176**	**0.9603**	**0.9472**	**0.9774**	**0.9449**
PASCAL-S	1			$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	0.0470	0.8997	0.8800	0.9272	0.8883
	2			$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	0.0463	0.9009	0.8812	0.9281	0.8888
	3	✓	✓	$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	0.0466	0.9017	0.8833	0.9275	0.8888
	4	✓	✓	$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	0.0458	0.9006	0.8821	0.9298	0.8897
	5	✓		$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	**0.0451**	**0.9025**	**0.8838**	**0.9309**	**0.8898**
	6	✓		$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	**0.0442**	**0.9038**	**0.8874**	**0.9332**	**08919**
ECSSD	1			$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	0.0213	0.9647	0.9519	0.9697	0.9465
	2			$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	0.0210	0.9649	0.9535	0.9700	0.9472
	3	✓	✓	$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	0.0213	0.9650	0.9532	0.9694	0.9467
	4	✓	✓	$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	0.0205	0.9664	0.9548	0.9717	0.9488
	5	✓		$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}$	**0.0202**	**0.9673**	**0.9558**	**0.9719**	**0.9491**
	6	✓		$\mathcal{L}^I_{BCE}+\mathcal{L}^I_{IoU}+\mathcal{L}^I_{Hist}$	**0.0186**	**0.9679**	**0.9568**	**0.9741**	**0.9510**

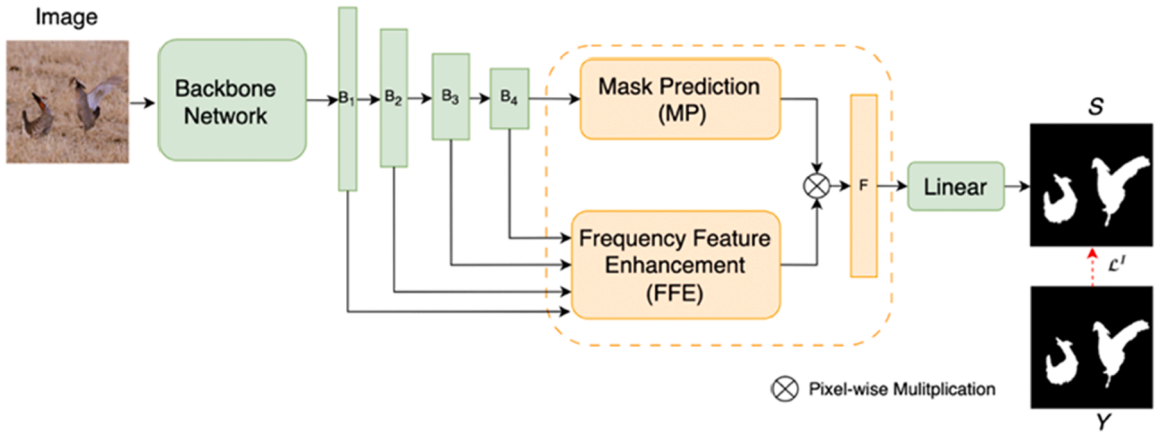

Fig. 12. One stream-TFGNet structure. Multiscale features from a backbone encoder are refined and optimized by a pixel-decoder (FFE) and a TF-decoder (MP).

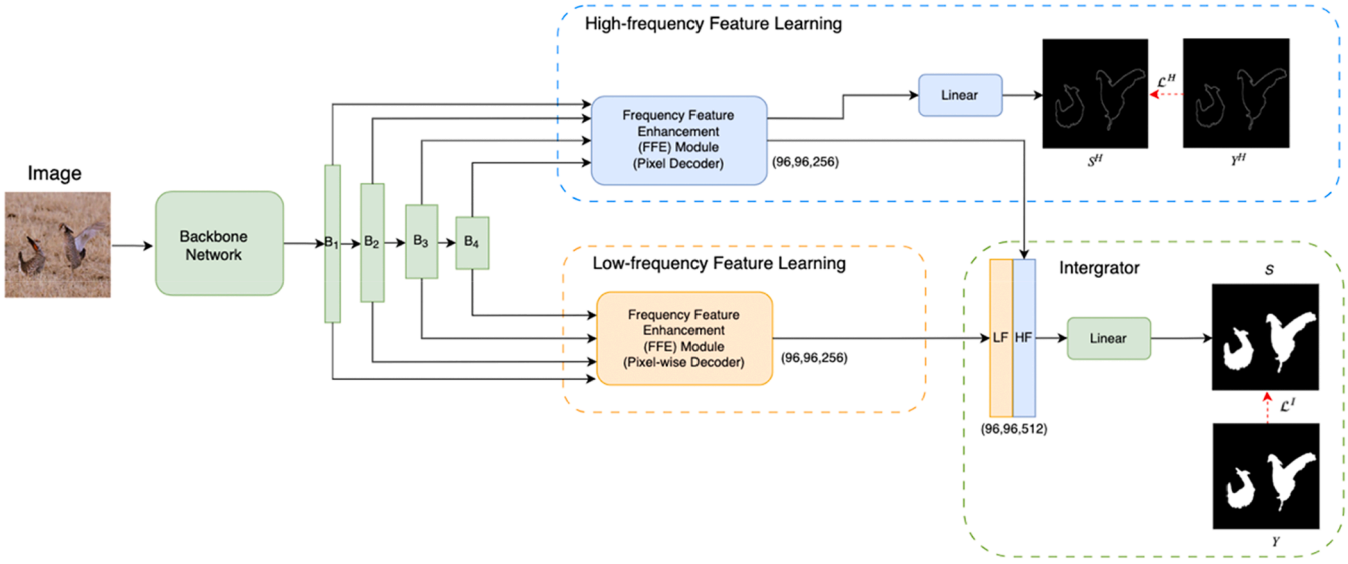

Fig. 13. Simple-TFGNet structure. Multiscale features from a backbone encoder are refined and optimized by an HSF Pixel-wise Decoder and an LSF Pixel-wise Decoder.

structures than its counterparts. These examples highlight that TFGNets possess greater robustness and maintains higher fault tolerance across various failure scenarios compared to other SOD models.

4.6. Ablation study

We ablate different model design choices on the performance of the proposed framework. Here, we use TFGNet-L384 as the baseline, and the training and testing settings are the same as in the above experiments.

4.6.1. Loss function

This experiment is designed to evaluate the influence of different loss configurations on the performance of TFGNet, as shown in Table 5. The test cases are categorized into three categories.

(i) Both branches are unsupervised (*Case* 1 and *Case* 2);
(ii) Both branches are supervised (*Case* 3 and *Case* 4);
(iii) Only the high-frequency branch is under supervision (*Case* 5 and *Case* 6).

Within each category, the study also assessed the impact of integrating the proposed histogram-based loss ($\mathscr{L}^{I}_{Hist}$) into the integrator module. The results indicate that the unsupervised cases for both branches (*Case* 1 and *Case* 2) generally performed worse than the other scenarios. When both branches are fully supervised (*Case* 3 and *Case* 4), there is a risk of over-supervision, which can sometimes lead to overfitting or reduced generalization.

Interestingly, when only the high-frequency branch was supervised (*Case* 5 and *Case* 6), the low-frequency branch benefited indirectly from the integrator's full-band supervision, highlighting the complementary relationship between high- and low-frequency information. Notably, the experiments demonstrated that applying supervision solely to the high-frequency branch (*Case* 7 and *Case* 8) yielded the best outcomes.

Moreover, introducing the histogram-based loss ($\mathscr{L}^{I}_{Hist}$) consistently improved performance across most scenarios compared to setups without this loss component. Based on these findings, the configuration in *Case* 6 is the optimal setup for TFGNet.

4.6.2. Network configuration

In this ablation study, we evaluate various configurations of TFGNet.

(i) **Baseline-TFGNet**: The original architecture of TFGNet, illustrated in Fig. 2, serves as baseline.

Table 6
Configuration comparisons. The top two results are colored red and green.

Dataset	Method	Branch Num.	MP/TF-Decoder	FFE/Pixel-Decoder	MAE	MF	mF	E_m	S_m
DUT-OMRON	One stream-TFGNet	1	✓	✓	0.0462	**0.8602**	0.8438	0.9067	0.8760
	Simple-TFGNet	2		✓	**0.0437**	0.8597	**0.8455**	**0.9083**	**0.8768**
	Baseline-TFGNet	2	✓	✓	**0.0402**	**0.8614**	**0.8488**	**0.9118**	**0.8799**
DUTS-TE	One stream-TFGNet	1	✓	✓	0.0205	0.9348	0.9191	0.9551	0.9321
	Simple-TFGNet	2		✓	**0.0200**	**0.9366**	**0.9225**	**0.9557**	**0.9331**
	Baseline-TFGNet	2	✓	✓	**0.0193**	**0.9375**	**0.9238**	**0.9566**	**0.9340**
HKU-IS	One stream-TFGNet	1	✓	✓	0.0179	0.9596	**0.9445**	0.9771	0.9444
	Simple-TFGNet	2		✓	**0.0177**	**0.9602**	**0.9472**	**0.9772**	**0.9447**
	Baseline-TFGNet	2	✓	✓	**0.0176**	**0.9603**	**0.9472**	**0.9774**	**0.9449**
PASCAL-S	One stream-TFGNet	1	✓	✓	0.0454	**0.9035**	0.8824	**0.9318**	**0.8922**
	Simple-TFGNet	2		✓	**0.0444**	0.9025	**0.8849**	**0.9332**	**0.8929**
	Baseline-TFGNet	2	✓	✓	**0.0442**	**0.9038**	**0.8874**	**0.9332**	0.8919
ECSSD	One stream-TFGNet	1	✓	✓	0.0196	0.9658	0.9527	0.9723	0.9491
	Simple-TFGNet	2		✓	**0.0188**	**0.9670**	**0.9562**	**0.9739**	**0.9499**
	Baseline-TFGNet	2	✓	✓	**0.0186**	**0.9679**	**0.9568**	**0.9741**	**0.9510**

(ii) **One Stream-TFGNet**: To assess the efficacy of the frequency decomposition strategy, we introduce a single-stream variant of TFGNet (Fig. 12). This configuration features a single feature learning through a Pixel-level Decoder (FFE module) and a TF-Decoder (MP module), employing a hybrid loss function for final predictions.

(iii) **Simple-TFGNet**: As depicted in Fig. 13, Simple-TFGNet consists of two branches, each equipped with the FFE module but lacking the MP module. The supervision remains consistent with those of the baseline network.

Table 6 summarizes the comparative results. Observations reveal that the TFGNet's two-stream configuration setting outperforms the one-stream setting on all five datasets. Specifically, Baseline-TFGNet achieves reductions of 12.99 %, 5.85 %, 1.68 %, 2.64 %, and 6.10 % of *MAE* compared to the One Stream-TFGNet for five datasets. This demonstrates the effectiveness of the frequency decomposition strategy in the network structure, enabling the model to capture and utilize both HSF and LSF salient features better, leading to more accurate and robust saliency predictions. In addition, we verify that the gains come from the mask-level global features provided by the TF-Decoder in the case of Simple-TFGNet. Specifically, compared to Simple-TFGNet, Baseline-TFGNet achieves reductions of 8.09 %, 3.62 %, 0.56 %, 0.45 %, and 1.06 % in MAE across five datasets due to incorporating the MP.

5. Conclusion

In this work, we explore the fundamental issues of feature learning in current SOD models from the perspective of frequency decomposition and synthesis, namely, how to balance the learning of high-spatial frequency (edge/local) features and low-spatial frequency (global) features. To address this challenge, we further elaborate on the theoretical basis for adopting a high-low frequency decomposition strategy as a solution and critically discuss the common issues present in existing edge enhancement/two-branch methods that utilize this approach. We provide deeper insights and a more detailed analysis of frequency decomposition in SOD compared to existing studies, such as SFENet.

We introduce TFGNet, a Transformer-based network architecture specifically designed for salient object detection (SOD). TFGNet features a parallel two-branch decoder that refines high-frequency boundaries and low-frequency inner regions under the guidance of decomposed frequency-specific supervisions. This architecture alleviates the difficulty of predicting the entire saliency map in complex scenes and highlights the importance of multiscale spatial frequency features in saliency detection tasks. We integrate a pixel decoder alongside a Transformer decoder in each frequency branch of TFGNet. The pixel decoder incorporates a Frequency Feature Enhancement (FFE) module, enriching the salient information extracted from multiscale feature maps producing more comprehensive and robust salient features. Meanwhile, the TF-decoder generates mask-level discriminative features and per-mask embeddings, enhancing the transformer network's local

representation capabilities.

We propose a new hybrid loss function integrating histogram dissimilarity measurement with traditional BCE and IoU losses. This improved loss function facilitates better optimization during training, ultimately contributing to more accurate predictions of salient object boundaries in complex scenes.

The encoder, feature optimization, and decoder components design of TFGNet leverage the strengths of both frequency decomposition and transformer architectures. The framework of TFGNet is clear and modular, with each branch and feature optimization component designed independently. Future research can further optimize and expand upon this solid foundation.

CRediT authorship contribution statement

Bin Liu: Writing – review & editing, Supervision, Project administration. **Rui Xu:** Visualization, Validation, Software. **Juncheng Liu:** Software, Methodology, Funding acquisition, Formal analysis. **Ruili Wang:** Supervision, Methodology, Conceptualization. **Yi Wang:** Writing – review & editing, Writing – original draft, Project administration, Methodology, Funding acquisition, Formal analysis, Conceptualization. **Feng Hou:** Writing – review & editing, Writing – original draft, Visualization. **Tianzhu Wang:** Visualization, Validation, Software, Methodology.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Yi Wang reports financial support was provided by National Natural Science Foundation of China. Yi Wang reports a relationship with National Natural Science Foundation of China that includes: funding grants. Yi Wang previously studied at Jilin University. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is partially supported the National Natural Science Foundation of China under Grant. 62476037, U22B2052, and 61976037. This work is also partially supported by 2020 Catalyst: Strategic NZ-Singapore Data Science Research Programme Fund, MBIE, New Zealand.

Data availability

Data will be made available on request.
Predicted maps (Predicted maps)

References

[1] A.K. Gupta, A. Seal, M. Prasad, P. Khanna, Salient object detection techniques in computer vision—a survey, Entropy 22 (10) (2020) 1174.
[2] W. Wang, Q. Lai, H. Fu, Shen, H. Ling, R. Yang, Salient object detection in the deep learning era: An in-depth survey, IEEE Trans. Pattern Anal. Mach. Intell. 44 (6) (2021) 3239–3257.
[3] M. Zong, R. Wang, Y. Ma, W. Ji, Spatial and temporal saliency-based four-stream network with multi-task learning for action recognition, Appl. Soft Comput. 132 (2023) 109884.
[4] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, Proc. Int. Conf. Med. Image Comput. Comput. -Assist. Interv. (2015) 234–241.
[5] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Proc. Adv. Neural Inf. Process. Syst. 25 (2012) 1097–1105.
[6] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) (2016) 770–778.
[7] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, Proc. Int. Conf. Learn. Represent. (ICLR) (2015).
[8] L.C. Chen, G. Papandreou, F. Schroff, H. Adam. Rethinking atrous convolution for semantic image segmentation," arXiv preprint arXiv:1706.05587 (2017).
[9] Y. Dai, F. Gieseke, S. Oehmcke, Y. Wu, K. Barnard, Attentional feature fusion, Proc. IEEE Winter Conf. Appl. Comput. Vis. (2021) 3560–3569.
[10] Y. Wang, R. Wang, X. Fan, T. Wang, X. He, Pixels, regions, and objects: Multiple enhancement for salient object detection, Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (2023) 10031–10040.
[11] X. Li, Y. Wang, T. Wang, R. Wang, Spatial frequency enhanced salient object detection, Inf. Sci. 647 (2023) 119460.
[12] L. Zhu, J. Chen, X. Hu, C.W. Fu, X. Xu, J. Qin, P.A. Heng, Aggregating attentional dilated features for salient object detection, IEEE Trans. Circuits Syst. Video Technol. 30 (10) (2020) 3358–3371.
[13] X. Hu, C.W. Fu, L. Zhu, T. Wang, P.A. Heng, SAC-Net: spatial attenuation context for salient object detection, IEEE Trans. Circuits Syst. Video Technol. 31 (3) (2021) 1079–1090.
[14] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, M. Jagersand, BASNet: Boundary-aware salient object detection, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2019) 7471–7481.
[15] Z. Wu, L. Su, Q. Huang, Decomposition and completion network for salient object detection, IEEE Trans. Image Process. 30 (2021) 6226–6239.
[16] J.X. Zhao, J.J. Liu, D.P. Fan, Y. Cao, J. Yang, M.M. Cheng, EGNet: edge guidance network for salient object detection, Proc. IEEE Int. Conf. Comput. Vis. (2019) 8779–8788.
[17] J. Wei, S. Wang, Z. Wu, C. Su, Q. Huang, Q. Tian, Label decoupling framework for salient object detection, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2020) 13022–13031.
[18] B. Cheng, A.G. Schwing, A. Kirillov, Per-pixel classification is not all you need for semantic segmentation, Adv. Neural Inf. Process. Syst. 34 (2021) 17864–17875.
[19] M. Anqi, M. Mohri, Y. Zhong, Cross-entropy loss functions: Theoretical analysis and applications. in Proceedings of the International Conference on Machine Learning, PMLR, 2023, pp. 23803–23828.
[20] V. Beers, Floris, E. Okafor, M.A. Wiereing, Deep neural networks with intersection over union loss for binary image segmentation. in Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods, SciTePress, 2019.
[21] Z. Wu, L. Su, Q. Huang, Cascaded partial decoder for fast and accurate salient object detection, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2019) 3902–3911.
[22] X. Zhao, Y. Pang, L. Zhang, H. Lu, L. Zhang, Suppress and balance: A simple gated network for salient object detection, Proc. Eur. Conf. Comput. Vis. (2020) 23–28.
[23] L. Zhang, J. Wu, T. Wang, A. Borji, G. Wei, H. Lu, A multistage refinement network for salient object detection, IEEE Trans. Image Process. 29 (2020) 3534–3545.
[24] X. Qin, Z. Zhang, C. Huang, M. Dehghan, O.R. Zaiane, M. Jagersand, "U^2-Net: Going deeper with nested U-structure for salient object detection,", Pattern Recognit. 106 (2020) 107404.
[25] X. Binwei, R. Liang, P. Chen, Locate globally, segment locally: A progressive architecture with knowledge review network for salient object detection, Proc. AAAI Conf. Artif. Intell. 35 (4) (2021) 3004–3012.
[26] L. Tang, B. Li, Y. Zhong, S. Ding, M. Song, Disentangled high quality salient object detection, Proc. IEEE Int. Conf. Comput. Vis. (2021) 3580–3590.
[27] Y. Fang, H. Zhang, J. Yan, W. Jiang, Y. Liu, UDNet: Uncertainty-aware deep network for salient object detection, Pattern Recognit. 134 (2023) 109099.
[28] M. Zhuge, D.P. Fan, N. Liu, D. Zhang, D. Xu, L. Shao, Salient object detection via integrity learning, IEEE Trans. Pattern Anal. Mach. Intell. 45 (3) (2023) 3738–3772.
[29] Y.H. Wu, Y. Liu, L. Zhang, M.M. Cheng, B. Ren, EDN: Salient object detection via extremely-downsampled network, IEEE Trans. Image Process. 31 (2022) 3125–3136.
[30] Z. Wu, S. Li, C. Chen, A. Hao, H. Qin, Deeper look at image salient object detection: Bi-stream network with a small training dataset, IEEE Trans. Multimed. 24 (2020) 73–86.
[31] H. Zhou, Y. Lin, L. Yang, J. Lai, X. Xie, Benchmarking deep models on salient object detection, Pattern Recognit. 145 (2024) 109951.
[32] N. Liu, N. Zhang, K. Wan, L. Shao, J. Han, Visual saliency transformer, Proc. IEEE/CVF Int. Conf. Comput. Vis. (2021) 4722–4732.
[33] J. Zhang, J. Xie, N. Barnes, P. Li, Learning generative vision transformer with energy-based latent space for saliency prediction, Adv. Neural Inf. Process. Syst. 34 (2021) 15448–15463.
[34] Y.K. Yun, W. Lin. "Selfreformer: Self-refined network with transformer for salient object detection," arXiv preprint arXiv:2205.11283 (2022).
[35] W. Wang, E. Xie, X. Li, D.P. Fan, K. Song, D. Liang, T. Lu, P. Luo, L. Shao, "PVT v2: Improved baselines with pyramid vision transformer, Comput. Vis. Media 8 (3) (2022) 415–424.
[36] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, Proc. IEEE/CVF Int. Conf. Comput. Vis. (2021) 10012–10022.
[37] Z. Liu, H. Hu, Y. Lin, Z. Yao, Z. Xie, Y. Wei, J. Ning, Y. Cao, Z. Zhang, S. Lin, B. Guo, Swin transformer: hierarchical vision transformer using shifted windows, IEEE Trans. Pattern Anal. Mach. Intell. 45 (1) (2023) 923–938.
[38] Z. Xie, J. Wang, J. Chen, T. Bai, J. Lu, Q. Wu, X. Shen, Augmenting transformers with dense reasoning for visual saliency detection, Proc. IEEE/CVF Int. Conf. Comput. Vis. (2021) 4580–4589.
[39] A. Dosovitskiy, L. Beyer, A. Kolesnikov, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth

16x16 words: Transformers for image recognition at scale, Proc. 9th Int. Conf. Learn. Represent. (2021).
[40] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (2016) 770–778.
[41] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, Proc. Int. Conf. Med. Image Comput. Comput. -Assist. Interv. (2015) 234–241.
[42] Q. Yan, L. Xu, J. Shi, J. Jia, Hierarchical saliency detection, Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Portland, OR, USA (2013) 1155–1162.
[43] Y. Li, X. Hou, C. Koch, J.M. Rehg, A.L. Yuille, The secrets of salient object segmentation, Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Columb., OH, USA (2014) 280–287.
[44] F. Perazzi, P. Krähenbühl, Y. Pritch, A. Hornung, Saliency filters: Contrast-based filtering for salient region detection, Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Provid., RI, USA (2012) 733–740.
[45] R. Margolin, L. Zelnik Manor, A. Tal, How to evaluate foreground maps, Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Columb., OH, USA (2014) 248–255.
[46] D. Fan, C. Gong, Y. Cao, B. Ren, M. Cheng, A. Borji, Enhanced-alignment measure for binary foreground map evaluation, Proc. Int. Jt. Conf. Artif. Intell. (IJCAI), Stockh., Swed. (2018) 698–704.
[47] M.M. Cheng, D.P. Fan, Structure-measure: A new way to evaluate foreground maps, Int. J. Comput. Vis. 129 (9) (2021) 2622–2638.
[48] L. Hong, X. Wang, G. Zhang, M. Zhao, USOD10K: A new benchmark dataset for underwater salient object detection, IEEE Trans. Image Process. (2023) (PP).
[49] M.J. Islam, R. Wang, J. Sattar, SVAM: saliency-guided visual attention modeling by autonomous underwater robots, Robot.: Sci. Syst. (RSS), NY, USA (2022).