ELWNet: An Extremely Lightweight Approach for Real-Time Salient Object Detection

ELWNet: An Extremely Lightweight Approach for Real-Time Salient Object Detection

Zhenyu Wang, Yunzhou Zhang*, Yan Liu, Delong Zhu, *Graduate Student Member, IEEE,* Sonya A. Coleman, *Member, IEEE,* and Dermot Kerr,

***Abstract*—Existing lightweight salient object detection (SOD) methods aim to solve the problem of high computational costs that is prevalent with heavyweight methods. However, compared with heavyweight methods, the detection accuracy of lightweight methods is greatly reduced while real-time performance is not significantly improved. Therefore, we aim to establish a trade off between computational cost and detection performance by improving the network efficiency. We propose a fast and extremely lightweight end-to-end wavelet neural network (ELWNet) for real-time salient object detection. ELWNet can achieve salient object detection and segmentation at approximately 70FPS (GPU), 19FPS (CPU) with 76K parameters and 0.38G FLOPs. We introduce wavelet transform theory into a neural network, proposing a wavelet transform module (WTM), a wavelet transform fusion module (WTFM), a novel feature residual mechanism, and construct an efficient architecture. The wavelet transform theory is integrated into the neural network to realize the interaction between the features in the frequency and the time domain. Meanwhile, ELWNet does not rely on a pre-trained model, which significantly reduces redundant features. We validate the performance of ELWNet using five well-known datasets, and demonstrate state-of-the-art performance compared with 24 other SOD models in terms of being lightweight, detection accuracy and real-time capabilities. Our method maintains high detection performance while reducing the number of model parameters by approximately 99% compared with heavyweight methods.**

***Index Terms*—Salient Object Detection, Wavelet Neural Network, Extremely Lightweight, Accuracy and Real-time.**

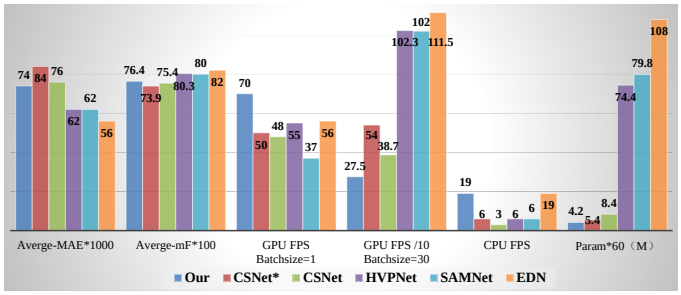


Fig. 1: Comparison of existing lightweight models. Our model achieves improvements in the extremely lightweight method. Compared with the lightweight models EDN, SAMNet and HVPNet, our model is compressed by 96%, 95% and 94%, and when serial processing, the FPS on GPU (CPU) is improved by 25% (same), 89% (217%) and 27% (217%), respectively, but the detection performance (The average of the Average-mF + Average-MAE) were only reduced by 19%, 12% and 12%, respectively. Unfortunately, the parallel processing speed of our model is not ideal.

I. Introduction

Salient object detection (SOD) has attracted significant attention due to its ability to quickly capture salient objects in complex scenes similar to the human visual system [1], [2]. Due to its importance in the field of computer vision, a large number of SOD methods have emerged. They have been widely used in video processing [3]–[6], co-salient object detection [7], semantic segmentation [8], simultaneous localization and mapping [9], RGB-D/T processing [10]–[13] and robot navigation [14], and SOD has been demonstrated to effectively improve the accuracy and robustness of these visual tasks. These methods continuously refresh the detection performance through complex network architectures, advanced feature fusion mechanisms, efficient loss functions and the introduction of edge features, but ignore the huge computational costs and are difficult to deploy on devices with limited computing resources. Therefore, the design of a lightweight SOD method with state-of-the-art performance to better serve computer vision tasks is still necessary.

Recently, the problem of SOD and associated computational costs has been addressed and a series of excellent solutions have been provided. Among them, CSNet [15], HVPNet [16], SAMNet [17] and EDN [18] are four representative lightweight SOD methods. Compared with the heavyweight model MINet [19] (with 162M model parameters), CSNet, HVPNet, SAMNet and EDN dominate the lightweight SOD methods with 0.14M, 1.24M, 1.33M and 1.80M model parameters, respectively. It is worth noting that CSNet can be defined as an extremely lightweight SOD model, and its variant CSNet* [15] has become the smallest SOD model so far with just 92K model parameters. As shown in Figure 1, under the

* Corresponding author.

Z. Wang is with Faculty of Robot Science and Engineering, Northeastern University, Shenyang, China, while Joint-Supervision at the Technical University of Munich, Munich, Germany.(e-mail: 1910652@stu.neu.edu.cn).

Y. Liu is with Faculty of Robot Science and Engineering, Northeastern University, Shenyang, China (e-mail: 1810630@stu.neu.edu.cn).

Y. Zhang is with College of Information Science and Engineering, Northeastern University, Shenyang, China (e-mail: zhangyunzhou@mail.neu.edu.cn

D. Zhu is with Department of Electronic Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong SAR, China (e-mail: zhudelong@link.cuhk.edu.hk).

S. A. Coleman and D. Kerr are with the Intelligent Systems Research Centre, Ulster University, Magee Campus, Londonderry BT48 7JL, U.K. (email: sa.coleman@ulster.ac.uk;d.kerr@ulster.ac.uk).

Manuscript received April 19, 2022; revised August 16, 2022.

premise of acceptable detection performance, these models are highly lightweight but their real-time performance is weak. We focus on the lightweight nature and real-time performance of lightweight models. Although our model has no advantage in the speed of parallel processing (Batchsize=30, mostly seen in offline processing), our model has faster speed for serial processing (Batchsize=1, mostly seen in online processing), and smaller batchsize requires less computing resources, so it is easier to deploy to mobile devices to achieve real-time processing. The main purpose of this paper is to improve the serial speed of lightweight models, so the real-time performance in this paper represents the serial speed if there is no special statement. Specifically, in the same hardware environment, the real-time performances of HVPNet, SAMNet and EDN are 55 (6) FPS, 37 (6) FPS and 56 (19) FPS respectively (CPU results are in parentheses and GPU have no parentheses). Even the extremely lightweight model CSNet and CSNet^* are only 48 (3) FPS and 50 (6) FPS respectively, not as good as the 73 FPS of the heavyweight model LDF [20], and unfortunately LDF cannot run efficiently on a CPU. It is notable that although lightweight SOD models have received attention, the overall performance is still poor. Hence, how to design a SOD model that takes into account lightweight parameters, accuracy and real-time capabilities is an important area of study.

In the current SOD field, the backbone network (ResNet [21] or Vgg [22], etc.) is often used as an encoder to extract features and a decoder is designed to decode the features, which is an "encoder-decoder" architecture. This is feasible for heavyweight SOD models, but the large number of backbone network parameters make the development of lightweight SOD models more challenging. The lightweight SOD model needs to continuously improve the detection performance through simple and efficient network architectures and strategies. Therefore, the main difficulties are: 1) how to realize a lightweight encoder to encode the image and generate high-quality multi-level semantic features; 2) how to realize a lightweight decoder to decode the multi-level semantic features of the encoder output and get the expected effect. Therefore, we need a novel architecture and mechanism to build a SOD network to achieve an appropriate trade-off among being lightweight, accuracy and real-time performance.

Motivated by this, we focus on the field of extremely lightweight and propose a novel wavelet neural network architecture to realize the extremely lightweight SOD method, named ELWNet. It is a plug-and-play SOD model, so almost all other vision tasks can benefit from it and are not limited by the shortage of computing resources. Aiming at the aforementioned two difficulties, this algorithm still adopts the "Encoder-Decoder" architecture, and we solve the existing problems by designing an encoder and a decoder. Benefiting from the mathematical theory of wavelet transforms [23], firstly, we ingeniously combine the wavelet transform theory with a convolutional neural network (CNN) to propose the wavelet transform module (WTM). With the help of the characteristics of wavelet transforms, we design an encoder, which replaces the traditional down-sampling (convolution stride=2, average/max-pooling) multi-level feature output with the help of the WTM. Additionally, it does not require pre-training on ImageNet and requires only a few computational resources to code output features with low redundancy. Subsequently, we propose the wavelet transform fusion module (WTFM), which combines wavelet transform theory and the residual mechanism to realize the up-sampling operation of features, which can restore more useful information, and build the decoder through the cascade of WTFM. The decoder makes full use of limited coding features in this efficient way, and outputs high-quality predicted saliency maps in the form of layer-by-layer refinement. The introduction of the wavelet transform enables the model to learn features between the time and the frequency domain, complementing each other, whilst achieving a comprehensive trade-off among lightweight, accuracy, and real-time performance. Finally, we use the regular hybrid loss function to optimize the network at the pixel and object level and show competitive performance.

The contributions of the paper are summarized as:

- We propose an extremely lightweight SOD method, named ELWNet, which has a good balance among lightweight, accuracy and real-time performance. It achieves high detection performance with 76K parameters, 0.38G of FLOPs, and 70 (19) FPS on GPU (CPU), making it the most lightweight model to our knowledge.
- We propose a novel wavelet transform module (WTM), which realizes feature down-sampling by fusing wavelet transform theory and a CNN. The encoder is built through the cascade of WTM, and high-quality multi-level encoding features are realized using limited parameters.
- We propose a novel wavelet transform fusion module (WTFM), which enables feature up-sampling and fusion via wavelet transform, a CNN, and a novel residual mechanism. The decoder is constructed through WTFM cascade to achieve high-quality decoding.

II. Related Works

Visual saliency detection can be traced back to 1998 to early work conducted by Itti et al. [24]. Salient object detection can be divided into two main approaches: those based on traditional methods and those based on deep learning. Traditional methods [25]–[27] mainly rely on hand-crafted features such as color, brightness, and texture to achieve salient object detection. With the rapid development of deep learning technologies, deep learning methods [28]–[47] based on high-level semantic features surpass traditional methods which are based on hand-crafted features. In this paper, we mainly focus on the SOD models based on deep learning.

A. Heavyweight Salient Object Detection

As SOD performance continues to benefit from the progression of deep learning technology, scholars have constantly improved SOD detection performance by designing various advanced modules and novel strategies. For example, Wang et al. [43] proposed a SOD method that integrates both top-down and bottom-up saliency inference in an iterative and cooperative manner. Meanwhile, they [44] proposed a PAGE-Net, which uses the essential pyramid attention structure for SOD and emphasizes the importance of salient edges. Ma et

al. [29] proposed a pyramidal shrinking decoder in which the adjacent fusion module preserves useful information in adjacent feature nodes and reduces noise and a SEM is designed to make the initial features more diverse. Wang et al. [45] proposed an attentive saliency network that learns to detect salient objects from fixations. The model narrows the gap between salient object detection and fixation prediction. Zhang et al. [30] proposed a multi-scale feature fusion framework based on a neural architecture search to realize the optimal automatic search strategy to guide the network fusion of multi-scale features. Wu et al. [31] proposed a decomposition and completion network for SOD, which improves network performance through joint learning of SOD with salient edge detection and salient skeleton detection. Zhao et al. [35] proposed a complementary trilateral decoder, comprising of semantic, spatial and boundary paths. The three branches in the network complement each other to achieve performance improvements. Mei et al. [42] constructed a dense context exploration module to capture dense multi-scale contexts and further leverage the learned contexts to enhance the features discriminability. Zhuge et al. [48] designed a DFA module, an ICE module and a PWV module so that the integrity cognition network is able to capture diverse features at each feature level and enhance feature channels. Finally, Lai et al. [47] proposed a weakly supervised method for visual saliency prediction to solve the limitation of dense annotation required by fully supervised methods, and achieved a balance between detection performance and computing resources.

In general, heavyweight SOD methods often sacrifice computational resources for detection performance. And it is difficult to apply to devices with insufficient computing resources.

B. Lightweight Salient Object Detection

Lightweight models have received attention in various fields with their lower computing resources. For example, inspired by the one teacher versus multiple students learning method, Shen et al. [49] proposed a distilled siamese tracking framework, which achieved the high compression rates and high frame-rates on the premise of competitive accuracy. Zhao et al. [50] proposed a real-time unsupervised video object segmentation network, and the proposed dynamic ASPP module and RNN-Conv module made the network better balance the relationship between computing resources and detection accuracy. Of course, the field of SOD is no exception.

Recent work has realized the limitations of heavyweight models, and focused on how to better balance detection performance and computing resource requirements, and three representative lightweight models (HVPNet, SAMNet, EDN) and an extremely lightweight model (CSNet) have been developed. Specifically, Liu et al. [16] proposed a HVP module to imitate the primate visual cortex for hierarchical visual perception learning. Subsequently, Liu et al. [17] proposed a SAM module (SAMNet), which enables small networks to encode both high-level features and low-level details. The proposed HVPNet and SAMNet have similar overall performance, and both achieve comparable performances with state-of-the-art SOD methods that use significantly more parameters, while saving several orders of magnitude of computational overhead. Recently, Wu et al. [18] proposed an extremely-downsampled network, which employs an extreme technique to effectively learn a global view of the whole image. They use scale-correlated pyramid convolution to build an elegant decoder for recovering object details from the above extreme downsampling. Compared with the two models mentioned above, although it requires more computing resources, the detection performance has been greatly improved. In the extremely lightweight field, Gao et al. [15] proposed the generalized OctConv to utilize both in-stage and cross-stages multi-scale features, while reducing the representation redundancy by a dynamic weight decay scheme. An extremely lightweight model CSNet^* is constructed using gOctConvs, which is the most lightweight SOD model to date. Compared with other lightweight models, CSNet series models remove the ImageNet pre-training process while still achieving high detection performance. Therefore, there is a lot of room for improvement in detection capability in lightweight field.

C. Wavelet Transform

The benefits of converting time domain features to the frequency domain are obvious, that is, there is no information loss in the mutual conversion between the RGB domain and the frequency domain. The difference is that the energy distribution in the frequency domain is more compact, and each channel clearly represents the information from different frequency bands. Therefore, it is necessary to introduce the frequency domain into the neural network [51], [52]. As we all know, wavelet transforms can not only obtain the frequency domain characteristics of the local time domain process, but also obtain the time domain characteristics of the local frequency domain process, which can deal well with stationary and non-stationary processes [23].

Nowadays, discrete wavelet transform (DWT) has been widely used in digital image processing. Although it is highly mathematical and rarely used in engineering, some excellent methods of introducing wavelet theory to neural networks have recently emerged. Through persistent homology analysis, Bae et al. [53] proposed a feature space deep residual learning algorithm which is better than the existing residual learning methods. Guo et al. [54] proposed a DWSR method to achieve high-quality recovery of missing details in sub-bands. Liu et al. [55] proposed a MWCNN, which embeds the wavelet transform into the CNN structure, reduces the resolution of the feature map and increases the receptive field, so as to realize a better trade-off between receptive field size and computational efficiency. Wang et al. [56] proposed a DeepGWC for semi-supervised node classification tasks. With the help of adding wavelet bases, the performance of DeepGWC is better than other existing graph deep models. Finally, Xin et al. [57] proposed an attention-based wavelet convolutional neural network for epilepsy EEG classification. The wavelet convolutional neural network obtains the signal components of different frequency bands. They are then fed into a convolutional neural network for further feature extraction and classification.

The design idea and purpose of our method are different from other methods based on wavelet transform. Specifically, different from [55], which only performs convolution

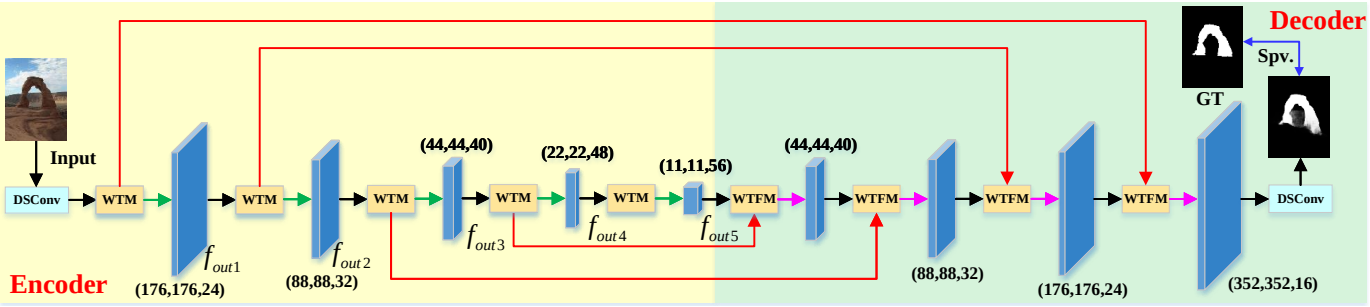


Fig. 2: The overall architecture of the ELWNet. WTM and WTFM decompose and fuse features at the frequency-domain level with the help of wavelet transform theory. WTM produces more recognizable multi-level features, and WTFM efficiently fuses multi-level features, injecting rigorous mathematical theory into the model. GT: Ground Truth; Spv.: Supervision.

operations after DWT and IDWT to achieve simple feature processing, our method fuses and compresses multi-frequency features at the full-frequency level, and adopts the way of first up-sampling and then down-sampling and the mechanism of hierarchical fusion of frequency features, which improves the performance of the model and enhances its anti-interference ability. Meantime, the novel residual mechanism we proposed further improves the performance. Like applications in other fields, the goal is to improve the performance of the model through the interaction of time-frequency features. Therefore, we apply the architecture of "wavelet transform + CNN" to the field of extremely lightweight SOD, which better balances the detection performance and computing resources.

III. Proposed Method

A. The Overall Architecture

We aim to design a salient object detection (SOD) method that can be arbitrarily embedded in various vision tasks, turning it into a plug-and-play SOD model that is not limited by computational resources. Therefore, we propose an extremely lightweight wavelet neural network for SOD, named ELWNet, which is the most lightweight SOD model to our knowledge. The overall architecture of the network is presented in Figure 2. We use the depth-wise separable convolution (DSConv) [58] operation instead of the conventional convolution (Conv) operation to effectively control the number of model parameters, similar to other lightweight methods [15]–[18]. The only input is an RGB image, which is fed to the encoder for feature encoding to output multi-level features, where the wavelet transform module (WTM) is the core of the encoder (see Section III.B), which effectively reduces the loss of useful features and feature redundancy during encoding. Subsequently, the decoder decodes the multi-level features output by the encoder. High-quality feature decoding is achieved with the help of the reversibility of the wavelet transform and novel strategies (see Section III.C), effectively reducing the loss of useful information caused by up-sampling and fusion operations. Finally, to demonstrate the superiority of our proposed model, we only use the regular hybrid loss function to fully supervise the network and output the expected saliency map (see Section III.D). Hence, our proposed ELWNet is an efficient model, which can learn enough information with limited model parameters.

B. Wavelet Transform Encoder

Since extremely lightweight models are very sensitive to the number of parameters, we cannot use an existing heavy-weight backbone (ResNet [21] or VGG [22]) or lightweight backbone (MobileNets [58] or GhostNet [59]) as encoders for ELWNet. These have a large number of parameters and are not conducive to compression to meet the needs of an extremely lightweight model. Therefore, we aim to design an encoder tailored for extremely lightweight SOD methods.

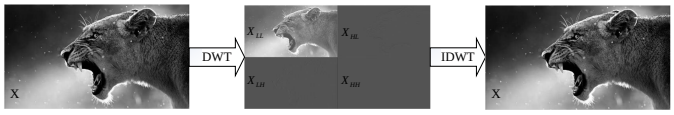


Fig. 3: Using a wavelet transform. After 2-D DWT, four sub-images are obtained. X_{LL}, X_{LH}, X_{HL} and X_{HH} are low frequency information, horizontal, vertical and diagonal high frequency information, respectively. The 4 sub-images can achieve perfect restoration through IDWT.

1) The Theory of Discrete Wavelet Transform: A wavelet transform (WT) inherits the advantages of a fourier transform (FT) while overcoming its disadvantages. A WT can provide a "time-frequency" window that changes with frequency, and is dominant in analyzing (non-)stationary signals [23]. Due to its reversibility, wavelet transforms are widely used in signal processing, image processing and other fields. In this paper, we use a Haar wavelet transform to process image features. Specifically, as shown in Figure 3, we decompose the input image X into four sub-band images X_{LL}, X_{LH}, X_{HL} and X_{HH} by convolution operations with low-pass filters ($f_{LL} = \begin{bmatrix} +1 & +1 \\ +1 & +1 \end{bmatrix}$) and high-pass filters ($f_{LH} = \begin{bmatrix} -1 & -1 \\ +1 & +1 \end{bmatrix}$, $f_{HL} = \begin{bmatrix} -1 & +1 \\ -1 & +1 \end{bmatrix}$, $f_{HH} = \begin{bmatrix} +1 & -1 \\ -1 & +1 \end{bmatrix}$) with convolutional stride 2, respectively. The resolution of the sub-band images is half of the input image, and the number of channels remains unchanged [60].

2) The Structure of the Wavelet Transform Module: It is well known that the encoder needs to encode the input features and produce multi-level features, and we usually achieve feature resolution down-sampling using stride = 2 convolution and average/maximum pooling. These operations result in the

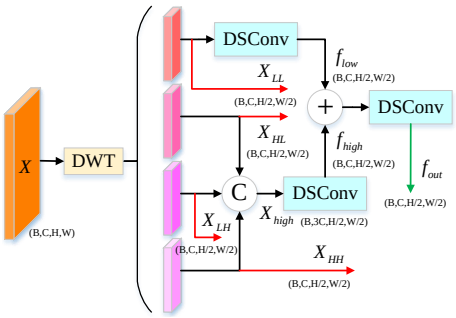


Fig. 4: The structure of the Wavelet Transform Module.

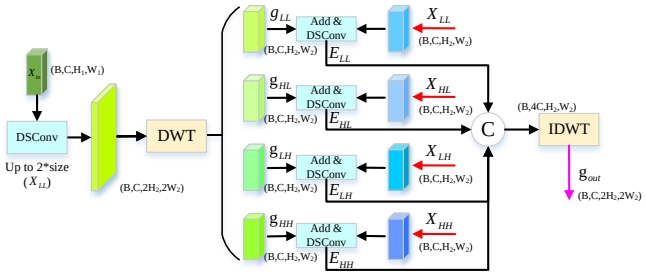


Fig. 5: The structure of Wavelet Transform Fusion Module.

loss of useful information to some extent, therefore how to effectively suppress the loss of useful information due to down-sampling is a problem worthy of attention. Due to the decomposition of a discrete wavelet transform (DWT), it can be regarded as a down-sampling operation. Motivated by this, as shown in Figure 4, we propose a wavelet transform module (WTM) that combines DWT with the convolution operation to achieve feature down-sampling. The working mechanism of WTM mainly includes three stages, as follows:

Stage 1: The input X is decomposed into 4 sub-band maps by a 2D DWT. The sub-band map X_{LL} acts as a low frequency component and represents the basic features of the input X. The sub-band maps X_{LH}, X_{HL} and X_{HH} are used as high-frequency components and represent the details of the input X (such as edge contours, etc.).

Stage 2: The low-frequency component X_{LL} and the fused high-frequency component X_{high} are further learned and extracted as useful information driven by the task through DSConv respectively. They are expressed as:

$$\begin{aligned} f_{low} &= DSConv(X_{LL}) \\ f_{high} &= DSConv(Concat(X_{LH},X_{HL},X_{HH})) \end{aligned} \tag{1}$$

Stage 3: f_{low} and f_{high} are closely integrated through DSConv, where the low-frequency and high-frequency features complement each other, resulting in a down-sampled feature map (with complete high-frequency and low-frequency information). The outputs of f_{out} are expressed as:

$$f_{out} = DSConv(f_{low} + f_{high}) \tag{2}$$

$Concat$ means that features are fused in a channel cascade. The batch normalization and rectified linear unit operations are performed once after each DSConv in the paper.

3) Module Comprehensive Design: As we all know, using stride=2 convolution (pooling) to achieve down-sampling is to learn convolution parameters (average or maximum pooling) in a data-driven manner. With different types of data, the convolution parameters trained by the network will be inconsistent, so it is often defined as a "black box" operation. When the WTM is used to implement the down-sampling operation, the DWT parameters are fixed and do not change with the data, and the down-sampling of the feature resolution is realized. Moreover, the down-sampling process of DWT has no feature loss as previously mentioned. Then, the obtained frequency domain features are learned and fused in a data-driven manner by DSConv (so that the network has a certain anti-interference ability), and the final output feature is a feature with complete frequency domain information.

4) The Architecture of Wavelet Transform Encoder: As shown in Figure 2, with the help of WTM, we propose an encoder similar to the ResNet and VGG, which realizes the output of five levels of different resolution features, which are expressed as f_{out1}, f_{out2}, f_{out3}, f_{out4} and f_{out5} respectively, from high to low resolution. The architecture of the encoder is very efficient, and multi-level features are generated by the 2x down-sampling function of the WTM. The output of WTM is divided into two parts: (1) f_{out} is used as the input of the next WTM to continue feature extraction; (2) X_{LL}, X_{HL}, X_{LH} and X_{HH} enhance the decoding capability of the decoder (see Section III.C) by means of a novel residual mechanism. This encoder, tailored for an extremely lightweight SOD method, does not require pre-training (for example, on ImageNet). Therefore, the extracted multi-level features have low redundancy and can more accurately capture the deep meaning of salient objects.

C. Wavelet Transform Decoder

1) The Theory of Inverse Discrete Wavelet Transform: Since the wavelet transform is reversible, as shown in Figure 3, IDWT is just the inverse operation of DWT. We also use the four filters (see Section III.B. *1)*) to analyze the low-frequency components (X_{LL}) and high-frequency components (X_{HL} , X_{LH} and X_{HH}), these four components are fused point-to-point by the value of the filter, which realizes the reconstruction of the original input image X [60].

2) The Structure of Wavelet Transform Fusion Module: Since the decoder needs to decode the multi-level features generated by the encoder, we focus on how to achieve high-quality up-sampling and fusion operations. For this purpose, we propose a novel wavelet transform fusion module (WTFM), as shown in Figure 5. Although the network structure of WTFM seems similar to CHFF in DMRA [67], they are completely different ideas. The essential difference is that our model emphasizes the processing of image features in the frequency domain to compensate for the incomplete feature learning in the time domain. Moreover, the CCIB in CHFF is the enhancement of a single input feature in the time domain, while we are using the fusion between two features in the frequency domain. The working mechanism of WTFM mainly includes three stages, as follows:

TABLE I: The datasets and evaluation criteria for salient object detection.

Datasets					
Name	Year	Stage	Size	Characteristic	Attribute
DUTS-TR [61]	2017	Train	10553	Complex	Multi-object, different sizes
DUTS-TE [61]	2017	Test	5019	Complex	Multi-object, different sizes
DUT-OMRON [25]	2013	Test	5168	Complex	Multi-object, different sizes
ECSSD [62]	2015	Test	1000	Simple	Mostly single-object, large size
PASCAL-S [63]	2014	Test	850	Complex	Multi-object, moderate size
HKU-IS [64]	2015	Test	4447	Complex	Multi-object, moderate size
Evaluation Criteria					
Name	Formula				Characterization
Detection Performer Criteria					
F-measure [65]	$mF = \frac{(1+\beta^2)Precision*Recall}{\beta^2 Precision+Recall}$				Weighted combination of precision and recall.
Mean Absolute Error [66]	$MAE = \frac{1}{W\times H}\sum_{i=1}^{W}\sum_{j=1}^{H}\lvert P(i,j)-G(i,j)\rvert$				Difference between the output and the GT.
Intersection-over-Union	$IOU = \frac{TP}{TP+FP+FN}$				Overlap between the output and the GT.
Efficiency Performer Criteria					
Model Parameters	#Param				Lightweight degree of the model.
Model Size	#Size				Size of storage occupied by the model.
Floating-Point Operations	FLOPs				Computational cost of the model.
Frames Per Second	FPS				Real-time performance of the model.

Stage 1 : The DWT is performed after the up-sampling operation on the input X_{in} (which has complete frequency characteristics), which realizes the decoupling of the low-frequency and high-frequency components of the input. This operation can not only achieve matching with the multi-frequency feature inputs using the residual structure, but also improve the anti-interference ability of the network by means of the up-sampling operation. The outputs of g_{LL}, g_{HL}, g_{HL} and g_{HH} of DWT are expressed as:

$$g_{LL}, g_{HL}, g_{HL}, g_{HH} = DWT[Up(DSConv(X_{in}))] \quad (3)$$

Stage 2 : The encoder stage, the decoder stage and the up-sampling operation in the first stage will introduce some uncertain factors (such as inaccuracy in the interpolation of values), so the previous features can enhance the current features at the frequency level by means of a residual mechanism. The enhanced E_{LL}, E_{HL}, E_{HL} and E_{HH} are expressed as:

$$E_i = DSConv(g_i + X_i), \quad i = \{LL, HL, LH, HH\} \quad (4)$$

Stage 3 : Benefiting from the reversibility of wavelet transform, the fusion of the low-frequency (E_{LL}) component and high-frequency (E_{HL}, E_{HL} and E_{HH}) components is achieved after IDWT processing, and the up-sampled feature g_{out} is obtained:

$$g_{out} = IDWT(E_{LL}, E_{HL}, E_{LH}, E_{HH}) \quad (5)$$

3) Module Comprehensive Design: As we all know, the method of up-sampling and fusing features in the traditional mode is to perform data-driven convolution parameter learning after direct up-sampling. This is a "black box" operation, and the difference in data will affect the performance. Different from it, WTFM first up-samples the features, performs data-driven convolution learning, and then implements down-sampling of features through DWT. This way of up-sampling and down-sampling will introduce a large amount of noise, so that the trained convolution parameters have a certain anti-interference ability. Then, the frequency domain information generated by both the DWT and residual mechanism are fused in a data-driven convolution method, which makes the energy distribution of useful features more compact in frequency domain. Finally, with the help of the theory that IDWT can restore features, the up-sampling of input features is realized.

4) The Architecture of Wavelet Transform Decoder: As shown in Figure 2, we propose a decoder adapted to the encoder. It is worth noting that we have made corresponding improvements compared to the basic idea of U-Net [79] to be more suitable for extremely lightweight SOD tasks. The subtlety of the decoder is that we employ a decoder structure that is asymmetrical to the encoder structure. We directly decode the high-level semantic information (defined as f_{out4} and f_{out5}) uniformly, that is, the residual feature of the WTM that generates the f_{out4} feature is directly provided to the f_{out5} feature for decoding. This asymmetric decoding structure can not only maintain the network performance, but also control the network parameters (one less WTFM is used to decode the features of f_{out4} separately). This is because high-level semantic information provides abstract features such as object location. For low-level semantic information (defined as f_{out1}, f_{out2} and f_{out3}), we need to decode the features layer by layer through the residual mechanism, because the detailed features such as contours provided by the low-level semantic information can more intuitively improve the segmentation effect of the saliency map. The ablation experiments in Section IV also demonstrate the superiority of our framework. Using the WTFM and residual mechanism, the proposed decoder can well decode the limited-informative encoded features produced by the encoder and output the expected saliency map.

D. Loss Function

To show the superiority of the proposed model, we only use the conventional loss function for supervised training of the model. Therefore, we use the hybrid loss function of binary cross entropy (BCE) and intersection over union (IOU) to optimize the network with respect to the pixel and object

TABLE II: Detection performance comparison with 24 state-of-the-art methods using five datasets including mF (larger is better) and MAE (smaller is better). For lightweight metrics, we crop the image to 320×320 resolution (except for VST (Auto-MSFNet) where the image size is 224×224 (256×256)). From left to right in the FPS column are the FPS under the GPU and CPU when the Batchsize is 1, and the FPS under the GPU when the Batchsize is 30 (except Batchsize=18 of DFI). R.C is Regular Conv and D.C is DSConv. The best results for an extremely lightweight method are marked with bold **red**.

Methods	Year	Type	#Param (M)	#Size (MB)	FLOPs (G)	FPS	DUTS-TE 5019 images		DUT-OMRON 5168 images		ECSSD 1000 images		PASCAL-S 850 images		HKU-IS 4447 images	
							mF↑	MAE↓	mF↑	MAE↓	mF↑	MAE↓	mF↑	MAE↓	mF↑	MAE↓
Heavyweight method (#Param>10M)																
BASNet [68]	CVPR 2019	R.C	87.06	348.5	199.31	41 (-) 56	.791	.048	.756	.056	.880	.037	.771	.076	.895	.032
CPD [69]	CVPR 2019	R.C	47.85	192.0	14.73	42 (-) 390	.805	.043	.747	.056	.917	.037	.820	.071	.891	.034
PoolNet [39]	CVPR 2019	R.C	69.56	278.5	89.65	64 (-) 119	.819	.037	.752	.054	.919	.035	.826	.065	.903	.031
SCRN [70]	ICCV 2019	R.C	25.23	101.4	12.53	38 (-) 361	.809	.040	.746	.056	.918	.037	.827	.063	.896	.034
EGNet [71]	ICCV 2019	R.C	111.66	447.1	244.13	34 (-) 63	.815	.039	.755	.053	.920	.037	.817	.074	.902	.031
DFI [72]	IEEE TIP 2020	R.C	29.61	118.8	22.44	42 (-) 31	.814	.039	.752	.055	.920	.035	.830	.065	.902	.031
U2-Net [73]	PR 2020	R.C	44.01	176.3	58.83	45 (-) 123	.792	.045	.761	.054	.892	.033	.770	.074	.896	.031
GCPANet [74]	AAAI 2020	R.C	67.06	268.6	54.36	62 (-) 201	.817	.038	.748	.056	.919	.035	.827	.062	.898	.031
F3Net [75]	AAAI 2020	R.C	25.54	102.5	13.63	62 (-) 417	.840	.035	.766	.053	.925	.033	.835	.061	.910	.028
GateNet [76]	ECCV 2020	R.C	128.63	514.9	112.64	52 (-) 120	.807	.040	.746	.055	.916	.040	.819	.067	.899	.033
ITSD [77]	CVPR 2020	R.C	26.07	106.2	19.71	54 (-) 266	.804	.041	.756	.061	.895	.034	.785	.066	.899	.031
MINet [19]	CVPR 2020	R.C	162.38	650.0	87.10	43 (-) 122	.828	.037	.755	.056	.924	.033	.829	.064	.909	.029
LDF [20]	CVPR 2020	R.C	25.15	100.9	12.87	73 (-) 426	.855	.034	.773	.052	.930	.034	.843	.060	.914	.028
PSGL-Net [36]	IEEE TIP 2021	R.C	25.55	102.6	16.12	60 (-) 342	.849	.036	.772	.053	.932	.031	.842	.061	.917	.028
Auto-MSFNet [30]	ACM MM 2021	R.C	33.35	130.4	24.55	58 (-) 410	.856	.034	.778	.050	.929	.033	.843	.061	.914	.027
VST [32]	ICCV 2021	R.C	44.09	178.4	23.24	40 (-) 156	.818	.037	.756	.058	.920	.033	.829	.061	.900	.029
PFSNet [29]	AAAI 2021	R.C	31.18	125.1	37.61	44 (-) 145	.846	.036	.774	.055	.932	.031	.837	.063	.919	.026
ICON [48]	IEEE TPAMI 2022	R.C	33.04	132.8	17.33	57 (-) 337	.838	.037	.772	.057	.928	.032	.833	.064	.910	.029
OLER [78]	ESWA 2022	R.C	26.58	-	-	-	.866	.033	.792	.050	.937	.030	.843	.063	.924	.026
Lightweight method (10M>= #Param>500K)																
HVPNet [16]	IEEE TCYB 2020	D.C	1.24	5.3	1.05	55 (6) 1023	.749	.058	.721	.065	.889	.052	.784	.089	.872	.044
SAMNet [17]	IEEE TIP 2021	D.C	1.33	5.8	0.50	37 (6) 1020	.745	.058	.717	.065	.891	.050	.778	.092	.871	.045
EDN [18]	IEEE TIP 2022	D.C	1.80	7.7	0.75	56 (19) 1115	.781	.050	.739	.058	.897	.049	.799	.084	.883	.040
Extremely Lightweight method (#Param)<=500K)																
CSNet [15]	IEEE TPAMI 2021	D.C	0.14	0.7	1.46	48 (3) 387	.687	**.074**	**.675**	**.081**	.844	.065	.723	.103	.840	.059
CSNet* [15]	IEEE TPAMI 2021	D.C	0.09	**0.5**	0.89	50 (6) **540**	.666	.082	.656	.087	.831	.074	.717	.111	.826	.065
ELWNet	year	D.C	**0.07**	**0.5**	**0.38**	**70 (19)** 275	**.696**	.075	.669	.083	**.858**	**.061**	**.746**	**.102**	**.850**	**.051**

level, and obtain superior performance. The overall hybrid loss function is defined as:

$$L_{total} = L_{bce} + L_{iou} \tag{6}$$

IV. Experiments

A. Datasets

Recently, the rapid development of SOD has benefited from a large number of public datasets and systematic evaluation criteria. Table I lists the datasets and evaluation criteria used. Readers can refer to the references for further details.

B. Implementation Details

The proposed ELWNet model is trained using the DUTS-TR dataset, and data augmentation techniques (such as horizontal flip, random crop and multi-scale input images) are employed to augment the training dataset. Our model is built on the PyTorch platform and runs on an NVIDIA RTX3090 GPU and Intel(R) Xeon(R) Platinum 8260C CPU @ 2.30GHz. The network is trained end-to-end by stochastic gradient descent (SGD), and the momentum and weight decay are set to 0.9 and 0.0005 respectively. The learning rate adopts a warm-up and a linear decay strategy and the maximum learning rate is 0.05. The batch size is set to 32, and the network converges after 200 epochs. During testing, each image is resized to 352×352 pixels, input to the network for prediction and finally restored to the original image size through bilinear interpolation.

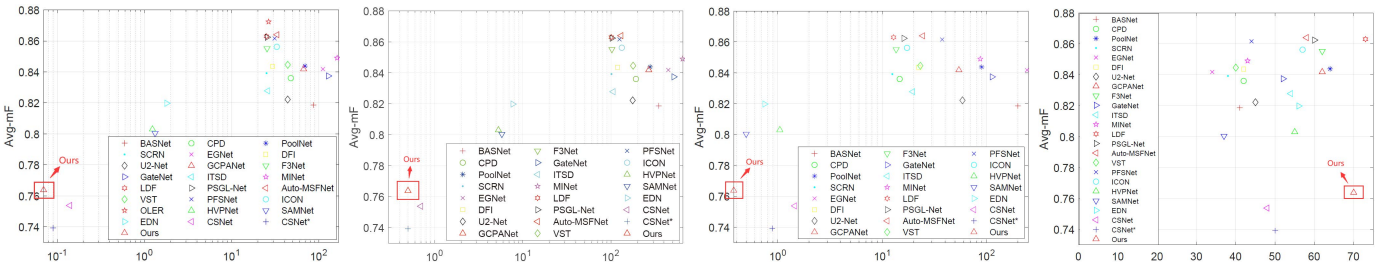
C. Performance comparison

As shown in Table II, we compare the proposed ELWNet model with 24 state-of-the-art SOD methods. For fair comparison, we use the saliency maps with the best performance as indicated by the authors and the lightweight metrics were evaluated using the same hardware. For detection metrics, we use the evaluation software provided on https://github.com/jiwei0921/Saliency-Evaluation-Toolbox. For lightweight metrics, we re-run the code provided by the authors to test in the same hardware environment.

Fig. 6: Quantitative comparison of the proposed method and extremely lightweight methods. Avg represents the average of the performance metrics across five datasets.

1) Quantitative Comparison: We divided SOD models into three categories according to model parameters: heavyweight models (#Param>10M), lightweight models (10M>= #Param>500K) and extremely lightweight models (#Param<=500K). We evaluate the models with respect to three criteria: model efficiency, detection accuracy and a combination of model efficiency and detection accuracy (comprehensive), so as to highlight the powerful performance and generalization ability of our proposed ELWNet.

Efficiency Performance Criteria. As shown in Table II,



(a) #Param (M) (b) #Size (MB) (c) FLOPs (G) (d) FPS

Fig. 7: Performance evaluation of the proposed method compared with 24 state-of-the-art methods.

we evaluate the efficiency performance of methods using four metrics of #Param, #Size, FLOPs and FPS. In these models, OLER does not have complete open source code, and the model parameters are extracted from the source paper [78]. ELWNet is the most lightweight SOD model to date with only 76K model parameters, 0.5MB model size, 0.38G FLOPs and 70 FPS. The FPS is lower than the heavyweight model LDF (although it has a high FPS, the huge number of model parameters and size make it difficult to deploy on devices with limited storage resources, or to embed in other visual tasks) due to the low degree of optimization of the wavelet transform and DSConv by the deep learning framework. The problem of optimization is not the focus of this paper. We can see that compared with HVPNet, SAMNet, EDN and CSNet(*), our method has advantages in lightweight and serial speed, but lags behind in parallel speed (See Section V). In practical applications, parallel and serial speed have their corresponding scene requirements, and this paper focuses on the serial speed of lightweight models. Specifically, compared with the extremely lightweight method CSNet(CSNet*), our method ELWNet reduces the number of model parameters, size and FLOPs by 50% (22%), 29% (same) and 74% (57%), respectively. The FPS are increased by 46% (533%) and 40% (217%) on the GPU (CPU), respectively. Compared with the lightweight method EDN, our method ELWNet reduces the model parameters, size and FLOPs by 96% , 94% and 49% , respectively. The FPS are increased by 25% (same) on the GPU (CPU), respectively. Compared with the representative heavyweight model MINet, the ELWNet reduces the #Param, #Size and FLOPs by 99.96%, 99.92% and 99.56% respectively, the FPS is increased by 63%. Additionally, heavyweight models are difficult to run smoothly on a CPU (even LDF cannot run on CPU). In general, ELWNet is an extremely lightweight SOD model, which has demonstrated improvement across all lightweight metrics. It can be easily embedded in mobile phones and some embedded chips and run smoothly.

Detection performance criteria. As shown in Table II and Figure 6, we evaluate the detection performance of ELWNet through three metrics, F-measure (mF: Mean-F-measure and MaxF: Max-F-measure), MAE and Mean-IOU. It is worth emphasizing that except for the CSNet series models and the proposed ELWNet, other models need to be pre-trained on ImageNet. Although we can make a fair comparison with the CSNet series models, we believe it is beneficial to pre-train the proposed ELWNet on ImageNet. From Table II, we can be seen that the detection performance of the model decreases with the improvement of the model's lightweight. Due to different application scenarios, heavyweight, lightweight and extremely lightweight models all have a role to play. We compare the models on five commonly used datasets. We can see that for the proposed ELWNet, compared with CSNet (CSNet*), the average performance on mF, MAX-F, MAE and Mean-IOU metrics is improved by 1.3% (3.3%), -1.2% (1.2%), 3.9% (11.9%) and 4.5% (7.8%) respectively. On the whole, in terms of extremely lightweight SOD, ELWNet has reached a high level in various detection performance metrics. Although its detection performance is not as good as that of heavyweight and lightweight models, it is approaching these performance levels. Simultaneously, it can also be seen from Section IV.D that the proposed model exhibits strong visual effects, which are sufficient to meet the detection performance requirements of most visual tasks.

Comprehensive criteria. For a lightweight model, it is unreasonable to consider the lightweight nature of the model in isolation of the detection performance. Our goal is to create an extremely lightweight model within an acceptable range of detection performance. To further demonstrate performance, we combine detection and efficiency performance for a comprehensive evaluation of ELWNet, as shown in Figure 7 where Avg-mF is the average of all mF metrics across the five datasets. In the subplots of avg-mF vs. #Param, avg-mF vs. #Size, and avg-mF vs. FLOPs, our method exhibits competitive performance in terms of lightweight, detection accuracy, and real-time performance. Specifically, compared with the extremely lightweight model CSNet (CSNet*), the proposed ELWNet improves Avg-mF by 1.3% (3.3%) and FPS by 46% (533%) and 40% (217%) on the GPU (CPU) under the premise that the #Param, #Size, and FLOPs are decreased by 50% (22%), 29% (same) and 74% (57%), respectively. Compared with heavyweight models, the detection performance has a certain loss, but the detection performance of our proposed ELWNet model is sufficient to meet the requirements of most visual tasks (See Table II and Section IV.D), and simultaneously, it has a degree of lightness that is difficult for heavyweight models to match (for example, ELWNet is 96% and 99% compressed compared with EDN and MINet respectively). In general, ELWNet has a good trade-off among being lightweight, accuracy and real-time performance.

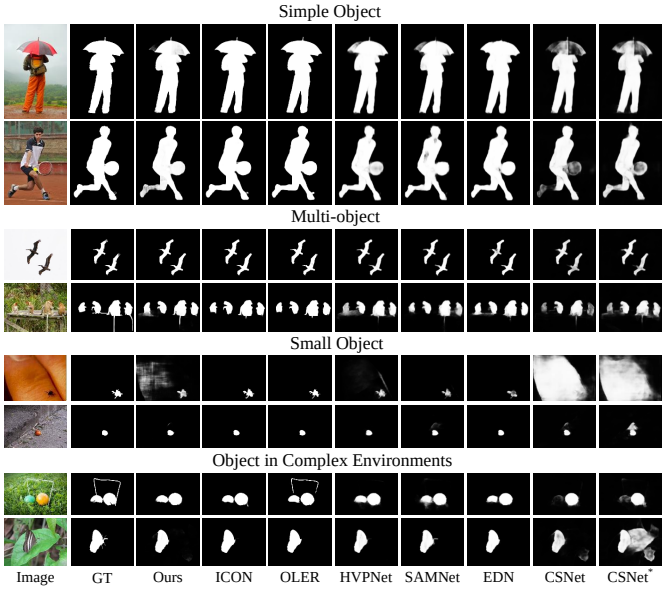


Fig. 8: Visual comparison of the proposed model with 7 state-of-the-art methods. Simple objects (Row 1 and 2), multi-objects (Row 3 and 4), small objects (Row 5 and 6), and objects in complex environments (Row 7 and 8).

D. Visual Comparison

For the actual application scenarios for SOD, good visual qualitative effects are more important than quantitative performance metrics. As shown in Figure 8, we evaluate the visual effects of the models from four typical scenarios: a simple object, multi-objects, a small object, and objects in complex environments. The proposed ELWNet is visually compared with partial heavyweight models, lightweight models and extremely lightweight models. For simple object and multi-object, our model can accurately localize and segment salient objects without serious errors such as missed detection. Although the detailed information cannot be recovered well, it has demonstrated competitive performance compared with the lightweight models, and the visual effect has also shown a greater advantage in the extremely lightweight models. For small objects, our model successfully detects and segments salient small objects. Although the background noise is not completely filtered out, compared with the extremely lightweight models, our visual effect is significantly improved. Even other extremely lightweight models cannot be used because they cannot detect small bugs on the finger (e.g. row 5). For salient object detection in complex environments, like some heavyweight and lightweight models, it is difficult for them to completely detect salient objects, and there will be problems such as missed detection or inability to restore edge details. However, compared with the extremely lightweight models, the visual effect of our model is greatly improved, which greatly reduces the chance of missed detection of salient objects. To sum up, although our model still has a gap in visual performance when compared with the heavyweight and lightweight models, we have achieved good results in the extremely lightweight SOD field, and are constantly approaching the visual performance of heavyweight and lightweight SOD models. With extremely low computing power requirements, our model can be applied to various complex vision application scenarios, and can be easily embedded into other vision algorithms, which has strong practicability.

E. Ablation Studies

To further evaluate our proposed method, ablation experiments are used to confirm that our designed components and strategies are effective. Therefore, we consider six aspects: 1) different types of wavelet transforms; 2) the effectiveness of the wavelet transform; 3) the effectiveness of the components; 4) different combinations of supervision; 5) different network structures; 6) analysis of ablation experiments at the visual level. All the ablation experiments follow the same implementation setup to ensure comparative evaluation.

1) Different types of wavelet transforms: Wavelet transform is different from Fourier transform. According to different wavelet basis functions, although they can decompose features into low-frequency and high-frequency features, the results obtained by wavelet transform are not the same. Therefore, as shown in Table III, we designed three sets of ablation experiments to demonstrate the effect of different wavelet basis functions on the overall performance of the model. We use Daubechies wavelet (db2), Coiflets wavelet (coif2), and Biorthogonal wavelet (bior2.4) to replace the Haar wavelet used in the proposed ELWNet respectively. We can find that when different wavelets are used, the lightweight of the models is consistent, but the gap between their detection performance is obvious. The main reason for this situation is due to the different characteristics between different wavelets. We believe that the properties of regularity and symmetry are relatively important in this model architecture. Good regularity performance can achieve better smoothing effect in image reconstruction, reducing the visual impact of quantization or rounding errors. And the good symmetry prevents phase distortion when analyzing and reconstructing the image. Although the Haar wavelet is the simplest wavelet function, it satisfies the characteristics necessary for image decomposition and reconstruction, so compared with db2, coif2 and bior2.4 show better performance. Specifically, compared with the best-performing coif2, the average detection performance (average of mF + MAE) of our proposed ELWNet is improved by 5%. This also shows that it is very important to choose the appropriate wavelet basis function for the SOD task. Compared with other three wavelet basis functions, Haar wavelet is more suitable for this architecture because of its tight support, regularity and symmetry, and makes the proposed method ELWNet show the state-of-the-art performance.

2) The effectiveness of the wavelet transform: We aim to verify that the introduction of the wavelet transform helps to improve the overall performance, and hence the overall network framework remains unchanged. Therefore, as shown in Table III, we designed seven sets of ablation experiments:

Plan A/B/C: DWT is replaced by stride=2 DSConv (plan A), average pooling (plan B) and max pooling (plan C), respectively, whilst reserving IDWT.

TABLE III: Results of ablation studies. The highest evaluation metric is marked in bold **red**.

NO.	Setting	#Param (M)	FLOPs (G)	FPS	DUTS-TE		DUT-OMRON		ECSSD		PASCAL-S		HKU-IS	
					mF↑	MAE↓	mF↑	MAE↓	mF↑	MAE↓	mF↑	MAE↓	mF↑	MAE↓
1	db2	0.07	0.38	70	.668	.084	.649	.091	.840	.068	.725	.110	.828	.058
2	coif2	0.07	0.38	70	.678	.082	.656	.089	.847	.067	.738	.107	.840	.054
3	bior2.4	0.07	0.38	70	.674	.082	.653	.090	.841	.068	.731	.109	.834	.055
4	Plan A	0.13	0.56	55	.677	.080	.645	.092	.847	.064	.740	.104	.825	.058
5	Plan B	0.07	0.38	78	.626	.100	.616	.107	.812	.079	.694	.125	.802	.068
6	Plan C	0.07	0.38	75	.682	.080	.664	.087	.838	.070	.728	.110	.837	.055
7	Plan D	0.14	0.80	70	.707	.072	.688	**.078**	.860	.062	.745	.103	.853	.050
8	Plan E	0.09	0.62	116	.632	.099	.616	.107	.816	.078	.693	.125	.805	.066
9	Plan F	0.09	0.62	113	.641	.093	.621	.101	.821	.079	.717	.116	.797	.070
10	Plan G	0.09	0.62	95	.691	.077	.669	.081	.850	.066	.736	.108	.843	.055
11	Plan H	**0.05**	**0.30**	**202**	.664	.087	.635	.099	.839	.071	.727	.112	.815	.064
12	Plan I	**0.05**	**0.30**	**202**	.650	.095	.632	.101	.829	.074	.703	.122	.815	.065
13	Plan J	**0.05**	**0.30**	191	.663	.088	.646	.095	.834	.072	.717	.116	.825	.060
14	Plan J-2	0.07	0.38	156	.664	.087	.635	.101	.822	.073	.728	.108	.812	.064
15	Plan K	0.07	0.38	70	.638	.094	.624	.099	.818	.079	.702	.123	.815	.062
16	Plan L	0.07	0.38	68	.609	.110	.600	.115	.798	.088	.686	.134	.789	.073
17	Plan M	0.07	0.38	67	.622	.104	.611	.110	.816	.080	.690	.127	.799	.068
18	Plan N	0.08	0.34	134	.690	.080	.663	.090	.854	.061	.742	.103	.841	.054
19	w/o. IOU	0.07	0.38	70	.642	.092	.622	.102	.830	.078	.715	.117	.819	.065
20	Plan O	0.09	0.38	62	.705	.075	.675	.083	.851	.065	.738	.102	.853	.051
21	Plan P	0.38	1.69	81	**.723**	**.069**	**.689**	.080	**.865**	**.058**	**.760**	**.097**	**.862**	**.045**
22	Ours	0.07	0.38	70	.696	.075	.669	.083	.858	.061	.746	.102	.850	.051

Plan D/E/F: DWT is replaced by stride=2 DSConv (plan D), average pooling (plan E) and max pooling (plan F), respectively, and IDWT is replaced by an up-sampling operation (interpolate('bilinear') + DSConv).

Plan G: Replace IDWT with up-sampling (interpolate('bilinear') + DSConv), and retain DWT.

Through these experiments, the effectiveness of DWT and IDWT in the overall framework has been comprehensively analyzed. By comparing the plans A/B/C with the proposed ELWNet, we can see that the overall performance of the model is seriously degraded after DWT is replaced by the conventional down-sampling methods (stride=2 DSConv, average pooling and max pooling) while retaining the IDWT. It can be see that the average detection performance (average of mF + MAE) of ELWNet on the five datasets is 5% higher than plan C. This is because the features information generated by conventional down-sampling methods cannot be fully utilized by IDWT, as the two information formats are inconsistent. Subsequently, we design the plans D/E/F to further verify the necessity of the existence of DWT and IDWT. We determined that without both DWT and IDWT, the performance of plan D is similar (the average detection performance (average of mF + MAE) is only improved by 1%) to that of the proposed ELWNet, however it requires twice the computational resources (Param: 0.14 vs. **0.07**, FLOPs: 0.80 vs. **0.38**). Specifically, compared with plan D, on the five datasets, the average mF metric of our proposed ELWNet has decreased by 1%, and the average MAE metric has decreased by 1%. At the same time, the model parameters have improved by 50%, and the FLOPs has improved by 53%. We believe that the average lightweight performance has improved by (50%+53%)/2≈52% while the detection performance metric has reduced by (1%+1%)/2≈1%, which is enough to prove the superiority of the our model. By comparing plans A/B/C and plans D/E/F, we again demonstrate that the feature information generated by the conventional down-sampling methods is difficult to fully utilize with IDWT. Finally, by comparing plans D/E/F with plan G, we can see that without IDWT, the conventional up-sampling operation (interpolate ('bilinear') + DSConv) also cannot fully utilize the feature information generated by DWT. Overall, it is not difficult to see that only when DWT and IDWT co-exist, the overall network shows the strongest comprehensive performance and achieves a balance between computing resources and detection performance.

3) The effectiveness of components: Next, we will demonstrate that our proposed WTM and WTFM effectively improve the overall performance of the network. Therefore, as shown in Table III, we designed eight sets of ablation experiments:

Plan H/I/J: Replace WTM with stride=2 DSConv (Plan H), average pooling (Plan I) and max pooling (Plan J) operations respectively, and then achieve feature learning through the DSConv. Additionally WTFM is replaced by up-sampling operation (interpolate('bilinear') + DSConv).

Plan J-2: Plan J-2 is an enhanced version of Plan J. By adding DSConv, making the model lightweight performance similar to ELWNet.

Plan K/L/M: Replace WTM with stride=2 DSConv (Plan K), average pooling (Plan L) and max pooling (Plan M) operations respectively, and then achieve feature learning through a DSConv operation whilst reserving WTFM.

Plan N: Replace WTFM with an up-sampling operation (interpolate('bilinear') + DSConv) whilst reserving WTM.

Through these experiments, the effectiveness of the two modules, WTM and WTFM, in the overall framework has been comprehensively analyzed. Firstly, by comparing plans H/I/J, we found that using the max pooling method (plan J) makes the detection performance of the model optimal, and the detection performance cannot be compared with ELWNet due to the different degree of light weight. Therefore, we added

plan J-2 to compare with the proposed ELWNet, however the result is contrary to our expectations, and even performed worse than plan J. This is mainly because the max pooling causes a large amount of information to be lost, and then the DSConv operation after the max pooling results in the network not learning more useful information, and a large amount of redundant information (such as noise or repeated features) is introduced. It is not difficult to see that the setting of plan J has saturated the performance and it is difficult to improve it. Secondly, by comparing plans H/I/J/J-2 and plans K/L/M, it can be seen that the introduction of WTFM will greatly reduce the overall performance of the network without the introduction of WTM. The reason for this is that the wavelet transform theory is only used in the decoder, and the multi-level features generated by the encoder do not have wavelet characteristics, that is, the encoding and decoding feature formats are inconsistent. Then, by comparing plans H/I/J with plan N, it can be seen that in the absence of WTFM, the use of WTM effectively improves the overall performance of the network, proving the effectiveness of WTM. Similarly, the effectiveness of WTM is also proven by comparing plans K/L/M with the proposed ELWNet. Finally, by comparing plan N with ELWNet, using the five datasets, our model improves the detection performance (average of mF + MAE) by 3% while improving the average lightweight performance by 0.4%. We can see that the ELWNet has improved both in terms of lightweight metrics and detection performance metrics, especially in MAE metric. From these experiments we can conclude that the best performance of the model can only be achieved by introducing both WTM and WTFM as they complement and promote each other, resulting in a strong comprehensive performance. This also demonstrates that WTM and WTFM are more competitive than traditional down-sampling and up-sampling operations.

4) Different combinations of supervision: A powerful network model still cannot achieve strong performance without a reasonable loss function for supervised training. As the focus of this paper is in proposing a novel extremely lightweight SOD architecture, instead of proposing a new loss function, we use the BCE+IOU hybrid loss function which enables fair comparisons with other models. Since the IOU loss function is not suitable for SOD pixel-level segmentation, the SOD network cannot be directly trained. Therefore, the ablation experiment scheme uses only the BCE loss function to supervise the training of the model, thus highlighting the necessity of the IOU loss function. As shown in Table III, comparing the NO.19 and NO.22, we can see that the introduction of the IOU loss function greatly improves the network performance, because the BCE+IOU hybrid loss function enables the network to learn feature at the object and pixel level, creating a model that greatly improves the detection performance without changing the lightweight nature of the model.

5) Different network structures: We verify the effectiveness of our designed network architecture. As shown in Table III, we design two ablation experiments to demonstrate the advantages of the proposed ELWNet in the extremely lightweight field. The specific settings are as follows:

Plan O: Different from the unified encoding of f_{out4} and f_{out5} output by the encoder in Figure 2, we decode f_{out4} and f_{out5} output by the encoder separately, that is to say, an additional WTFM is added.

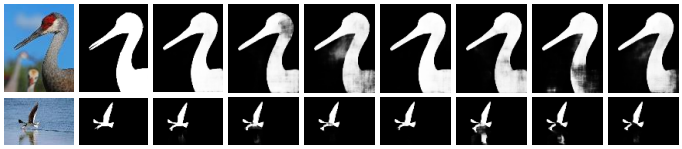


Fig. 9: Visual comparison of ablation experiments. ELWNet illustrates an accurate visual effect.

Plan P: We change all DSConv in ELWNet to regular Conv without changing anything anywhere else.

Through the above experimental, we can see the effectiveness of the proposed ELWNet. Specifically, compared with plan O, our model directly decodes high-level semantic information in a unified way, which not only reduces the complexity of the model, but also improves the detection performance of the network. Using the five test datasets, our model has improved the detection performance (average of mF + MAE) by 1% on the basis of a 11% improvement in average lightweight performance, which also shows that unified decoding is better than separate decoding under this architecture. Meanwhile, we test whether the performance of ELWNet under regular Conv is competitive. Specifically, by comparing plan P, we found that the experimental results are consistent with our expectations. Using the five test datasets, although our model detection performance (average of mF + MAE) is reduced by 4%, the average lightweight performance is improved by 80%. It also reflects the effectiveness of the proposed structure. According to the definition of model parameters, it is still an extremely lightweight SOD model. Therefore, different models can be selected for visual tasks in different scenes to achieve the desired effect.

6) Analysis of ablation experiments at the visual level: We focus on whether the introduction of wavelet transform theory can help improve the overall performance of the network at the visual level. As shown in Figure 9, we analyze this from the following three perspectives:

(1) By comparing the plan C, plan G and ELWNet, it is not difficult to find that the use of IDWT (plan C) or DWT (plan G) alone cannot make the network reach the optimal state without changing the overall framework, which seriously affects the visual effect of the model, indicating that DWT and IDWT can only play a powerful role when used together.

(2) Compared with plan D and ELWNet, it replaces DWT with traditional down-sampling (stride=2 DSConv) and IDWT with traditional up-sampling (interpolation ('bilinear') + DSConv) on the premise that the overall framework remains unchanged. Although the metric performance is slightly higher than that of ELWNet (see NO.7 of Table III), the visual effect map is similar to that of ELWNet and the model complexity of ELWNet is reduced by half. This shows that the introduction of wavelet transform can effectively control the complexity of the model under the premise of the detection performance.

(3) By comparing the plan J, plan K, plan N and ELWNet,

we found that neither WTM (plan K) nor WTFM (plan N) can achieve the optimal state. When WTM and WTFM do not exist (plan J), as well as plan K, can not suppress noise and segment accurate salient object. Similarly, it illustrates the importance of WTM and WTFM and the optimal state of the model can only be achieved when they are used together.

In conclusion, we found that the quality of visual effects is almost consistent with the quality of the metric value. It can be seen that the proposed ELWNet greatly improves the visual effect and retains more contextual details. Therefore, the ablation experiments of the above six aspects have more comprehensively confirmed the superiority of ELWNet.

V. Limitations

The proposed ELWNet has two main limitations:

1) Limitations of model detection performance. Although our proposed model achieves optimal detection performance in the extremely lightweight domain, it still falls short of the heavyweight and lightweight models in terms of overall detection performance. This is also to be expected. As shown in Figure 8, the visual effect of our model can still show good performance in the case of simple object and multiple objects, but when there are small objects, our model often cannot suppress the background noise very well. In complex environments, our model also misses valid objects, and cannot recover details such as the edges of the objects very well.

2) Real-time Performance Limitations. For serial speed, compared with the 70 FPS of ELWNet, the FPS of plan E is 116. This demonstrates that the introduction of wavelet transforms reduces the inference speed. At the same time, the FPS of plan P is 81, which shows that DSConv also reduces the inference speed. To sum up, the real-time performance of the model was seriously affected due to the low degree of optimization of wavelet transform and DSConv under the Pytorch platform. For parallel speed, compared with other lightweight models, our model has a large gap, but basically meets the real-time requirements. Therefore, the application scenario is limited, and may be delayed when parallel data needs to be processed at high speed.

VI. Conclusion

In view of the current problems with heavyweight models consuming a lot of computing resources, and lightweight models having low performance and weak real-time performance, we propose a novel extremely lightweight SOD method ELWNet. We integrate wavelet transform theory with a convolutional neural network, and propose two novel modules: wavelet transform module (WTM) and wavelet transform fusion module (WTFM). Additionally, with the help of a novel residual mechanism, we build an extremely lightweight SOD model of an "Encoder-Decoder" architecture. Our method removes the need for pre-training, and still maintains high detection and real-time performance on the basis of being extremely lightweight, achieving a good trade-off among lightweight, accuracy and real-time performance. To date, ELWNet is the most lightweight model with the best comprehensive performance in the field of extremely lightweight SOD methods, and achieves efficient execution using a CPU.

Further improvements in the detection performance of lightweight models so they approach the performance of a heavyweight model remain a topic for future development. In future work, we will focus on introducing more mathematical knowledge and theories into our model, and propose some efficient modules and mechanisms that will can not only improve the performance of the network, but also make the network more interpretable.

Acknowledgment

This work was supported by National Natural Science Foundation of China (No. 61973066), Major Science and Technology Projects of Liaoning Province (No. 2021JH1/10400049), Fundation of Key Laboratory of Aerospace System Simulation (No. 6142002200301) and Fundamental Research Funds for the Central Universities (N2004022).

References

[1] C. Koch and S. Ullman, "Shifts in selective visual attention: Towards the underlying neural circuitry," *Human Neurobiology*, vol. 4, no. 4, pp. 219–227, 1985.

[2] W. Wang, Q. Lai, H. Fu, J. Shen, H. Ling, and R. Yang, "Salient object detection in the deep learning era: An in-depth survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021.

[3] C. Chen, J. Song, C. Peng, G. Wang, and Y. Fang, "A novel video salient object detection method via semisupervised motion quality perception," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 2732–2745, 2022.

[4] W. Wang, J. Shen, and L. Shao, "Video salient object detection via fully convolutional networks," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 38–49, 2018.

[5] W. Wang, J. Shen, J. Xie, M.-M. Cheng, H. Ling, and A. Borji, "Revisiting video saliency prediction in the deep learning era," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 220–237, 2021.

[6] W. Wang, J. Shen, X. Lu, S. C. H. Hoi, and H. Ling, "Paying attention to video object pattern understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 7, pp. 2413–2428, 2021.

[7] D.-P. Fan, T. Li, Z. Lin, G.-P. Ji, D. Zhang, M.-M. Cheng, H. Fu, and J. Shen, "Re-thinking co-salient object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 8, pp. 4339–4354, 2022.

[8] Z. Zhang, Z. Cui, C. Xu, Y. Yan, N. Sebe, and j. Yang, "Pattern-affinitive propagation across depth, surface normal and semantic segmentation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4101–4110.

[9] K. Wang, S. Ma, F. Ren, and J. Lu, "Sbas: Salient bundle adjustment for visual slam," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2021.

[10] Z. Liu, Y. Tan, Q. He, and Y. Xiao, "Swinnet: Swin transformer drives edge-aware rgb-d and rgb-t salient object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 7, pp. 4486–4497, 2022.

[11] J. Wang, K. Song, Y. Bao, L. Huang, and Y. Yan, "Cgfnet: Cross-guided fusion network for rgb-t salient object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 2949–2961, 2022.

[12] T. Zhou, D.-P. Fan, M.-M. Cheng, J. Shen, and L. Shao, "Rgb-d salient object detection: A survey," *Computational Visual Media*, vol. 7, pp. 37–69, 2021.

[13] K. Fu, D.-P. Fan, G.-P. Ji, Q. Zhao, J. Shen, and C. Zhu, "Siamese network for rgb-d salient object detection and beyond," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5541–5559, 2022.

[14] C. Craye, D. Filliat, and J.-F. Goudou, "Environment exploration for object-based visual saliency learning," in *IEEE International Conference on Robotics and Automation*, 2016, pp. 2303–2309.

[15] M.-M. Cheng, S. Gao, A. Borji, Y.-Q. Tan, Z. Lin, and M. Wang, "A highly efficient model to study the semantics of salient object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021.

[16] Y. Liu, Y.-C. Gu, X.-Y. Zhang, W. Wang, and M.-M. Cheng, "Lightweight salient object detection via hierarchical visual perception learning," *IEEE Transactions on Cybernetics*, pp. 1–11, 2020.

[17] Y. Liu, X.-Y. Zhang, J.-W. Bian, L. Zhang, and M.-M. Cheng, "Samnet: Stereoscopically attentive multi-scale network for lightweight salient object detection," *IEEE Transactions on Image Processing*, vol. 30, pp. 3804–3814, 2021.

[18] Y.-H. Wu, Y. Liu, L. Zhang, M.-M. Cheng, and B. Ren, "Edn: Salient object detection via extremely-downsampled network," *IEEE Transactions on Image Processing*, vol. 31, pp. 3125–3136, 2022.

[19] Y. Pang, X. Zhao, L. Zhang, and H. Lu, "Multi-scale interactive network for salient object detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9410–9419.

[20] J. Wei, S. Wang, Z. Wu, C. Su, Q. Huang, and Q. Tian, "Label decoupling framework for salient object detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 022–13 031.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, May 2015.

[23] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *IEEE Transactions on Information Theory*, vol. 36, no. 5, pp. 961–1005, 1990.

[24] L. Itti and C. Koch, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[25] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3166–3173.

[26] X. Huang, Y. Zheng, J. Huang, and Y.-J. Zhang, "50 fps object-level saliency detection via maximally stable region," *IEEE Transactions on Image Processing*, vol. 29, pp. 1384–1396, 2020.

[27] Y.-Y. Zhang, S. Zhang, P. Zhang, H.-Z. Song, and X.-G. Zhang, "Local regression ranking for saliency detection," *IEEE Transactions on Image Processing*, vol. 29, pp. 1536–1547, 2020.

[28] Z. Wang, Y. Zhang, Y. Liu, S. Liu, S. Coleman, and D. Kerr, "Mfc-net : Multi-feature fusion cross neural network for salient object detection," *Image and Vision Computing*, vol. 113, p. 104243, 2021.

[29] M. Ma, C. Xia, and J. Li, "Pyramidal feature shrinking for salient object detection," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 3, pp. 2311–2318, 2021.

[30] M. Zhang, T. Liu, Y. Piao, S. Yao, and H. Lu, "Auto-msfnet: Search multi-scale fusion network for salient object detection," in *ACM International Conference on Multimedia*, 2021.

[31] Z. Wu, L. Su, and Q. Huang, "Decomposition and completion network for salient object detection," *IEEE Transactions on Image Processing*, vol. 30, pp. 6226–6239, 2021.

[32] N. Liu, N. Zhang, K. Wan, J. Han, and L. Shao, "Visual saliency transformer," in *IEEE/CVF International Conference on Computer Vision*, 04 2021.

[33] L. Sun, Z. Chen, Q. M. J. Wu, H. Zhao, W. He, and X. Yan, "Ampnet: Average- and max-pool networks for salient object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 11, pp. 4321–4333, 2021.

[34] X. Hu, C.-W. Fu, L. Zhu, T. Wang, and P.-A. Heng, "Sac-net: Spatial attenuation context for salient object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 3, pp. 1079–1090, 2021.

[35] Z. Zhao, C. Xia, C. Xie, and J. Li, "Complementary trilateral decoder for fast and accurate salient object detection," in *ACM International Conference on Multimedia*, 2021.

[36] S. Yang, W. Lin, G. Lin, Q. Jiang, and Z. Liu, "Progressive self-guided loss for salient object detection," *IEEE Transactions on Image Processing*, vol. 30, pp. 8426–8438, 2021.

[37] J. Zhang, J. Xie, N. Barnes, and P. Li, "Learning generative vision transformer with energy-based latent space for saliency prediction," *Advances in Neural Information Processing Systems 34 pre-proceedings*, vol. 34, 2021.

[38] Y. Liu, Y. Zhang, S. Liu, S. Coleman, Z. Wang, and F. Qiu, "Salient object detection by aggregating contextual information," *Pattern Recognition Letters*, vol. 153, pp. 190–199, 2022.

[39] J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng, and J. Jiang, "A simple pooling-based design for real-time salient object detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3912–3921.

[40] Q. Zhang, M. Duanmu, Y. Luo, Y. Liu, and J. Han, "Engaging part-whole hierarchies and contrast cues for salient object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 6, pp. 3644–3658, 2022.

[41] Z. Wang, Y. Zhang, Y. Liu, Z. Wang, S. Coleman, and D. Kerr, "Tf-sod: a novel transformer framework for salient object detection," *Neural Computing and Applications*, 2022.

[42] H. Mei, Y. Liu, Z. Wei, D. Zhou, X. Wei, Q. Zhang, and X. Yang, "Exploring dense context for salient object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1378–1389, 2022.

[43] W. Wang, J. Shen, M.-M. Cheng, and L. Shao, "An iterative and cooperative top-down and bottom-up inference network for salient object detection," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5961–5970.

[44] W. Wang, S. Zhao, J. Shen, S. C. H. Hoi, and A. Borji, "Salient object detection with pyramid attention and salient edges," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1448–1457.

[45] W. Wang, J. Shen, X. Dong, A. Borji, and R. Yang, "Inferring salient objects from human fixations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 1913–1927, 2020.

[46] W. Wang and J. Shen, "Deep visual attention prediction," *IEEE Transactions on Image Processing*, vol. 27, no. 5, pp. 2368–2378, 2018.

[47] Q. Lai, T. Zhou, S. Khan, H. Sun, J. Shen, and L. Shao, "Weakly supervised visual saliency prediction," *IEEE Transactions on Image Processing*, vol. 31, pp. 3111–3124, 2022.

[48] M. Zhuge, D.-P. Fan, N. Liu, D. Zhang, D. Xu, and L. Shao, "Salient object detection via integrity learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2022.

[49] J. Shen, Y. Liu, X. Dong, X. Lu, F. S. Khan, and S. Hoi, "Distilled siamese networks for visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 12, pp. 8896–8909, 2022.

[50] Z. Zhao, S. Zhao, and J. Shen, "Real-time and light-weighted unsupervised video object segmentation network," *Pattern Recognition*, vol. 120, p. 108120, 2021.

[51] K. Xu, M. Qin, F. Sun, Y. Wang, Y.-K. Chen, and F. Ren, "Learning in the frequency domain," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1740–1749.

[52] Z. Qin, P. Zhang, F. Wu, and X. Li, "Fcanet: Frequency channel attention networks," in *IEEE/CVF international conference on computer vision*, 2021, pp. 783–792.

[53] W. Bae, J. Yoo, and J. C. Ye, "Beyond deep residual learning for image restoration: Persistent homology-guided manifold simplification," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 1141–1149.

[54] T. Guo, H. S. Mousavi, T. H. Vu, and V. Monga, "Deep wavelet prediction for image super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 1100–1109.

[55] P. Liu, H. Zhang, K. Zhang, L. Lin, and W. Zuo, "Multi-level wavelet-cnn for image restoration," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 886–88 609.

[56] J. Wang and Z. Deng, "A deep graph wavelet convolutional neural network for semi-supervised node classification," in *2021 International Joint Conference on Neural Networks*, 2021, pp. 1–8.

[57] Q. Xin, S. Hu, S. Liu, L. Zhao, and Y.-D. Zhang, "An attention-based wavelet convolution neural network for epilepsy eeg classification," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 30, pp. 957–966, 2022.

[58] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[59] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "Ghostnet: More features from cheap operations," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1577–1586.

[60] R. S. Stanković and B. J. Falkowski, "The haar wavelet transform: its status and achievements," *Computers and Electrical Engineering*, vol. 29, no. 1, pp. 25–44, 2003.

[61] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan, "Learning to detect salient objects with image-level supervision," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3796–3805.

[62] J. Shi, Q. Yan, L. Xu, and J. Jia, "Hierarchical image saliency detection on extended cssd," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 4, pp. 717–729, 2016.
[63] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2014, pp. 280–287.
[64] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5455–5463.
[65] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1597–1604.
[66] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2012, pp. 733–740.
[67] W. Ji, G. Yan, J. Li, Y. Piao, S. Yao, M. Zhang, L. Cheng, and H. Lu, "Dmra: Depth-induced multi-scale recurrent attention network for rgb-d saliency detection," *IEEE Transactions on Image Processing*, vol. 31, pp. 2321–2336, 2022.
[68] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "Basnet: Boundary-aware salient object detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7471–7481.
[69] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3902–3911.
[70] Z. Wu, L. Su, and Q.-m. Huang, "Stacked cross refinement network for edge-aware salient object detection," in *IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7263–7272.
[71] J.-X. Zhao, J. Liu, D.-P. Fan, Y. Cao, J. Yang, and M.-M. Cheng, "Egnet: Edge guidance network for salient object detection," in *IEEE/CVF International Conference on Computer Vision*, 2019, pp. 8778–8787.
[72] J.-J. Liu, Q. Hou, and M.-M. Cheng, "Dynamic feature integration for simultaneous detection of salient object, edge and skeleton," *IEEE Transactions on Image Processing*, vol. 29, pp. 8652–8667, 2020.
[73] X. Qin, Z. Zhang, C. Huang, M. Dehghan, and M. Jagersand, "U2-net: Going deeper with nested u-structure for salient object detection," *Pattern Recognition*, vol. 106, p. 107404, 2020.
[74] Z. Chen, Q. Xu, R. Cong, and Q. Huang, "Global context-aware progressive aggregation network for salient object detection," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 7, pp. 10 599–10 606, 2020.
[75] J. Wei, S. Wang, and Q. Huang, "F3net: Fusion, feedback and focus for salient object detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, pp. 12 321–12 328.
[76] X. Zhao, Y. Pang, L. Zhang, H. Lu, and L. Zhang, "Suppress and balance: A simple gated network for salient object detection," in *European Conference on Computer Vision*, pp. 35–51.
[77] H. Zhou, X. Xie, J.-H. Lai, Z. Chen, and L. Yang, "Interactive two-stream decoder for accurate and fast saliency detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9138–9147.
[78] Z. Yao and L. Wang, "Object localization and edge refinement network for salient object detection," *Expert Systems with Applications*, vol. 213, p. 118973, 2023.
[79] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, 2015, pp. 234–241.

Zhenyu Wang received the M.S. degree in electronics and communication Engineering from the Dalian Maritime University, Dalian, China, in 2019.

He is currently a Ph.D. student joint education by the Faculty of Robotics and Engineering of Northeastern University in Shenyang, China, and the Department of Electronic and Computer Engineering of Technical University in Munich, Germany. He has participated in several research projects and published several journal articles. His research interests include intelligent robots, computer vision.

Yunzhou Zhang received the B.S. and M.S. degrees in mechanical and electronic engineering from the National University of Defense Technology, Changsha, China, in 1997 and 2000, respectively, and the Ph.D. degree in pattern recognition and intelligent system from Northeastern University, Shenyang, China, in 2009.

He is currently a Professor with the College of Information Science and Engineering, Northeastern University. He leads the Cloud Robotics and Visual Perception Research Group. His research has been supported by funding from various sources. He has published many journal articles and conference papers. His research interests include intelligent robots, computer vision, wireless sensor networks.

Yan Liu received the B.S. degree in Mathematics and Applied Mathematics from Tonghua Normal University, Tonghua, China, in 2016, and the M.S. degree in System Theory from Northeastern University, Shenyang, China, in 2018.

She is currently a Ph.D. student at the Faculty of Robot Science and Engineering, Northeastern University, Shenyang, China. Her research interests are system control and intelligent robot.

Delong Zhu (Graduate Student Member, IEEE) received the B.S. degree in computer science and technology from Northeastern University, Shenyang, China, in 2015, and the Ph.D. degree from the Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong, in 2020.

He spent nine months at the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA, as a Visiting Scholar. His research interests include motion planning in dynamic environments and deep reinforcement learning.

Sonya A. Coleman (Member, IEEE) received the B.Sc. degree (Hons.) in mathematics, statistics, and computing and the Ph.D. degree in mathematics from Ulster University, Londonderry, U.K., in 1999 and 2003, respectively.

She is currently a Professor with the School of Computing and Intelligent Systems, Ulster University, and a Cognitive Robotics Team Leader with the Intelligent Systems Research Centre. Her research has been supported by funding from various sources. She has authored or coauthored over 150 publications in robotics, image processing, and computational neuroscience.

Dermot Kerr received the B.Sc. degree (Hons.) in computing science and the Ph.D. degree in computing and engineering from Ulster University, Londonderry, U.K., in 2005 and 2008, respectively.

He is currently a Senior Lecturer with the School of Computing, Engineering and Intelligent Systems, Ulster University. His current research interests include computational intelligence, biologically inspired image processing, mathematical image processing, omnidirectional vision, and robotics.