IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 18 February 2023, accepted 26 March 2023, date of publication 29 March 2023, date of current version 16 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3263179

RESEARCH ARTICLE

Dual-Branch Network of Information Mutual Optimization for Salient Object Detection

ZIJUN CHEN, YINWEI ZHAN, (Member, IEEE), AND SHANGLEI GAO
School of Computer Science and Technology, Guangdong University of Technology, Guangzhou 510006, China
Corresponding author: Yinwei Zhan (ywzhan@ieee.org)

ABSTRACT Salient object detection (SOD) is to segment significant regions of images. Noticing that the saliency maps in existing SOD methods suffer from blurring boundaries owing to insufficient extraction of boundary features and inadequate fusion between boundary features and salient region features, a dual-branch network of information mutual optimization (DIMONet) is proposed. The DIMONet has a region detection branch and a boundary detection branch to extract the corresponding features simultaneously and is mainly composed of two components. One is the mutual optimization module (MOM) that refines salient region features and boundary features based on their internal relationship. The other is the fusion module of multi-receptive fields (FMMF) that integrates multi-layer features with the refined features to distinguish salient objects better and sharpen their boundaries. With the help of MOMs and FMMFs, noises from the background in the boundary features are gradually reduced and hence the boundaries of the salient regions get sharpened. Experiments on five benchmark datasets show that our method is superior to the 18 state-of-the-art methods.

INDEX TERMS Deep learning, salient object detection, mutual optimization, feature fusion.

I. INTRODUCTION

The purpose of salient object detection (SOD) is to detect the most fascinating subjects to people in a certain scene. Nowadays, SOD is widely used as an essential preprocessing technique in many downstream computer vision tasks, such as image translation [1], object tracking [2], [3] and semantic segmentation [4], [5].

In traditional SOD methods [6], [7], [8], [9], [10], [11], hand-crafted low-level features are widely used. However, the lack of high-level salient object information makes these features unsuitable for complex scenarios. Up to now, convolutional neural networks (CNNs) have accelerated the development of SOD thanks to their powerful ability to automatically learn high-level features. However, only extracting and fusing multi-layer features, most of the existing SOD methods [12], [13], [14], [15] are unable to make the boundaries of objects clean and clear due to the lack of the exploration of boundary information. In order to sharpen the boundaries of salient objects, some researchers [16], [17], [18], [19] design additional boundary prediction branches to extract accurate boundary features. But the structures of region detection branches are more complex than that of boundary detection branches, which makes their models pay more attention to the extraction of region features than the extraction of boundary features. As a result, the boundary features extracted by their boundary detection branches are full of noises from the background and therefore interfere with the detection of salient objects after fused with the region features, such as EGNet [17] and ITSD [16] in Fig. 1. Differently, some methods, for example SCRN [18], have the same structures of the region detection branch and the boundary detection branch and fuse the region features and the boundary features based on their internal relationship, and hence achieve better results than the EGNet and the ITSD. However, the SCRN does not consider the complementarity between multi-level features, which makes the salient objects cannot be accurately separated from the background.

In this paper, we design a dual-branch information mutual optimization network (DIMONet)to solve the blurring boundary problem in SOD task. The DIMONet has a salient region detection branch and a boundary detection branch of

The associate editor coordinating the review of this manuscript and approving it for publication was Essam A. Rashed.

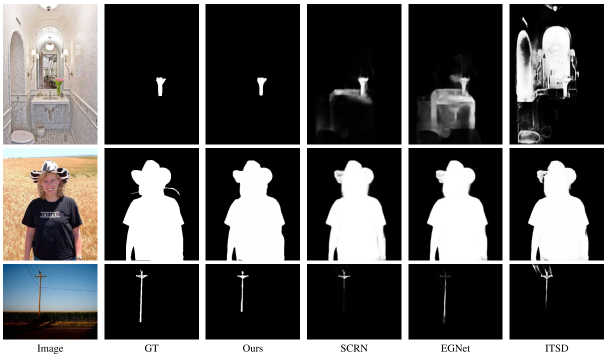


FIGURE 1. Prediction results by the proposed DIMONet, SCRN [18], EGNet [17] and ITSD [16].

the same structures to focus equally on the boundaries and the regions. In addition, in order to better refine the region features and the boundary features, a mutual optimization module (MOM) is proposed based on the internal relationship between the salient region and its boundary: the intersection set of the salient region and its boundary is the boundary, while their union set is the salient region. Besides, in order to make the salient region features and the boundary features characteristic and representative, a fusion module of multi-receptive fields (FMMF) is designed to fuse the refined features in the preceding stage and the original features in the succeeding stage. The fused features are then sent to a new MOM to be further refined. By utilizing the MOMs and the FMMFs to refine region features and boundary features several times, the noises in the boundary features can be reduced and the boundaries of the region features become clear.

In summary, our contributions are as follows:

1) We propose a DIMONet containing a region detection branch and a boundary detection branch of the same structures. Unlike previous networks, the DIMONet treats regions and boundaries equally, so that clean and accurate boundary features can be extracted.
2) We build a mutual optimization module to optimize the salient region features and the boundary features based on the internal relationship: the intersection set of the salient region and its boundary is the boundary, while their union set is the salient region. After being refined several times by the mutual optimization module, the features of the salient region and the boundary become clean and recognizable.
3) We design a fusion module of multi-receptive fields to make the salient region features and boundary features more representative.
4) Extensive experiments show that our method is superior to 18 state-of-the-art methods on five well-known datasets.

II. RELATED WORK

Hand-crafted features [6], [7], [8], [9], [10], [11] are widely used in most traditional salient object methods. However, these features can only represent some low-level semantic information, making these traditional methods unable to correctly segment salient objects in complex scenes. Recently, due to the fact that CNN has the strong ability that automatically learns high-level semantic information, many SOD methods based on CNN are proposed. Specifically, these methods can be divided into multi-level feature fusion methods and boundary-aware methods and are explained in detail below.

A. MULTI-LEVEL FEATURES FUSION METHODS

Some researchers believe that there is complementarity between multi-level features. Hence, various methods are proposed to integrate multi-level features in order to segment salient objects from natural scenes. Chen et al. [20] design a reverse attention network. By masking the predicted region in each side output, this network can gradually dig out the lacking parts of salient objects. Zhuge et al. [21] propose an integrity cognition network (ICON) to learn integrity features from micro and macro levels. Zhang et al. [15] fuse features of each stage in VGG-16 to generate features of different resolutions, which are then used for saliency detection. Xiao et al. [22] utilize short and long range connections to exploit the object context and preserve the object boundary for effectively integrating multi-scale features. Liu et al. [23] design a hierarchical recurrent convolutional neural network (HRCNN) in their deep hierarchical saliency network (DHSNet) that automatically learn various global structured saliency cues to refine the saliency map progressively. Wang et al. [24] propose a stagewise refinement model. They first generate a coarse prediction result and then integrate local context information by a pyramid pooling module and a multi-stage refinement mechanism to refine it. Hou et al. [25] build short connections in a top-down approach to densely combine multi-level features, and take the outputs of different layers into account to yield an ultimate saliency map. Wang et al. [26] design recurrent fully convolutional networks (RFCNs). By incorporating saliency prior knowledge and their recurrent architecture, the RFCNs can automatically recover image details and hence achieve more accurate results. Deng et al. [27] design a recurrent residual refinement network. By iterating high-level features and low-level features many times, this network can pick up residual information between mid-prediction and ground truth. Pang et al. [28] combine the features from neighboring levels to detect multi-scale objects in saliency detection. Wei et al. [29] design a cross feature module to mitigate the differences of multi-level features. Chen et al. [30] design a module for feature intertwining aggregation that fuses low-level features, high-level features and global features to generate a saliency map.

However, being a lack of exploration of the boundary information, the above methods do not make the object boundaries clear well.

B. BOUNDARY-AWARE METHODS

In order to make the boundaries of salient regions clear, some researchers introduce boundary labels in SOD. Su et al. [31] integrate multi-level features in the boundary localization module to strengthen the ability of the network to extract boundary features. Based on the logical interrelations between saliency maps and their boundary maps, Wu et al. [18] propose a cross refinement unit to simultaneously optimize multi-level features of saliency maps and boundary maps. Qin et al. [32] design a BASNet containing a coarse-refine architecture and a hybrid loss for salient object detection. Zhou et al. [16] design an adaptive contour loss to induce their network to focus more on hard samples. Wang et al. [19] propose a salient edge detection module that provides a powerful tip for their network to improve the object boundaries. Feng et al. [33] propose a boundary-enhanced loss and an attentive feedback module for their network to refine object boundaries. Wei et al. [34] utilize distance transformation to break the saliency map down into region map and detail map, which makes each pixel in the saliency map be treated unequally, and then the saliency map, region map and detail map are used for training. Zhao et al. [17] build an edge-guided network to extract salient object information and boundary information at the same time.

However, in most of these methods, the structures of region detection branch and boundary detection branch are different. These methods generally pay more attention to the design of the region detection branch than the boundary detection branch, which makes their network cannot extract clean and accurate boundary features well. As a result, noises in the boundary features interfere with the detection of salient regions.

III. PROPOSED METHOD

A. DIMONet PIPELINE

In the following, we denote Conv_n as a convolutional layer with kernel size $n \times n$ and y_i as an intermediate feature, $i \in \mathbf{Z}$.

Many works [24], [35] have shown that using the ResNet-50 as the backbone yields better results than using VGG-16. Therefore, we also use ResNet-50 as the backbone of the DIMONet. There are five features from low-level to high-level extracted by the ResNet-50. However, the low-level features are of the small receptive field and the largest resolution. Therefore, they contain a lot of noise and cost much computation. Therefore, only the features of the last four layers $L = \{L_1, L_2, L_3, L_4\}$ are used. For each L_i, extract the salient region features S_i and the boundary features B_i by one 1×1 convolutional layer with 64 output channels and one 3×3 convolutional layer in parallel, respectively:

$$\begin{cases} S_i = f_1(L_i), \\ B_i = f_2(L_i), \end{cases} \quad i = 4, 3, 2, 1, \tag{1}$$

where $f_1(\cdot)$ and $f_2(\cdot)$ represent operations $\mathrm{Conv}_3(\mathrm{Conv}_1(\cdot))$ with different initialization parameters.

An additional 3×3 convolutional layer is applied to S_4 and B_4 to obtain the global information S_5 and B_5, respectively. Therefore, two feature sets $S = \{S_1, S_2, S_3, S_4, S_5\}$ and $B = \{B_1, B_2, B_3, B_4, B_5\}$ are obtained and then are processed in parallel to form the region detection branch and boundary detection branch. Each branch contains four stages and each stage is a refinement of the output of the previous stage. In the first stage, the features S_5 and B_5 are first mutually optimized by the MOM $\mathbb{M}(\cdot)$ to obtain $\hat{S}_5$ and $\hat{B}_5$, then the feature sizes of $\hat{S}_5$ and $\hat{B}_5$ are made consistent with S_4 and B_4 by the upsampling operation $\mathrm{Up}(\cdot)$, and finally the upsampled $\hat{S}_5$ and S_4 and the upsampled $\hat{B}_5$ and B_4 are fused by two different FMMFs $\mathbb{F}(\cdot)$ to obtain $\tilde{S}_4$ and $\tilde{B}_4$, respectively. The remaining stages are similar to the first stage. The overall process of DIMONet can be expressed as:

$$\begin{cases} \tilde{S}_i = \mathbb{F}(S_i, \mathrm{Up}(\hat{S}_{i+1})), \\ \tilde{B}_i = \mathbb{F}(B_i, \mathrm{Up}(\hat{B}_{i+1})), \end{cases} \quad i = 4, 3, 2, 1, \tag{2}$$

$$\{\hat{S}_j, \hat{B}_j\} = \begin{cases} \mathbb{M}(S_j, B_j), & if \;\; j = 5, \\ \mathbb{M}(\tilde{S}_j, \tilde{B}_j), & if \;\; j = 4, 3, 2. \end{cases} \tag{3}$$

In order to guide the network to learn salient object information and boundary information more easily and accurately, a 1×1 convolution layer is applied to each feature in $\tilde{S}$ and $\tilde{B}$ to generate a set of saliency maps $\bar{S} = \{\bar{S}_1, \bar{S}_2, \bar{S}_3, \bar{S}_4\}$ and a set of boundary maps $\bar{B} = \{\bar{B}_1, \bar{B}_2, \bar{B}_3, \bar{B}_4\}$:

$$\begin{cases} \bar{S}_i = \mathrm{Conv}_1(\tilde{S}_i), \\ \bar{B}_i = \mathrm{Conv}_1(\tilde{B}_i), \end{cases} \quad i = 4, 3, 2, 1. \tag{4}$$

Like other U-shaped networks, the decoded feature resolutions in DIMONet gradually increase. As the resolution of the saliency map $\bar{S}_1$ is closest to the original input, a large amount of detailed information is retained. Therefore, the last saliency map $\bar{S}_1$ is taken as the final salient object mask in the inference stage.

B. MUTUAL OPTIMIZATION MODULE

The purpose of the region detection branch is to segment the complete region of targets from backgrounds, while the boundary detection branch is to detect the boundary of targets. To achieve effective refinement of features for each specific task, a MOM, as illustrated in Fig. 2, is used based on the internal interrelations: $\dot{B} \subset \dot{S}$, where $\dot{S}$ and $\dot{B}$ represent the ground truth saliency map and corresponding boundary map, respectively.

Specifically, the element-wise multiplication is used to get the intersection of $\dot{S}$ and $\dot{B}$, and the concatenation is to obtain the union of $\dot{S}$ and $\dot{B}$. For the salient object features, they are first concatenated with the boundary features along the channel axis and then use two 3×3 convolution layers to generate discriminative features of salient objects. Finally, a residual connection is used for better optimization. For the boundary features, its process is similar to the salient region features, but element-wise multiplication is used instead of

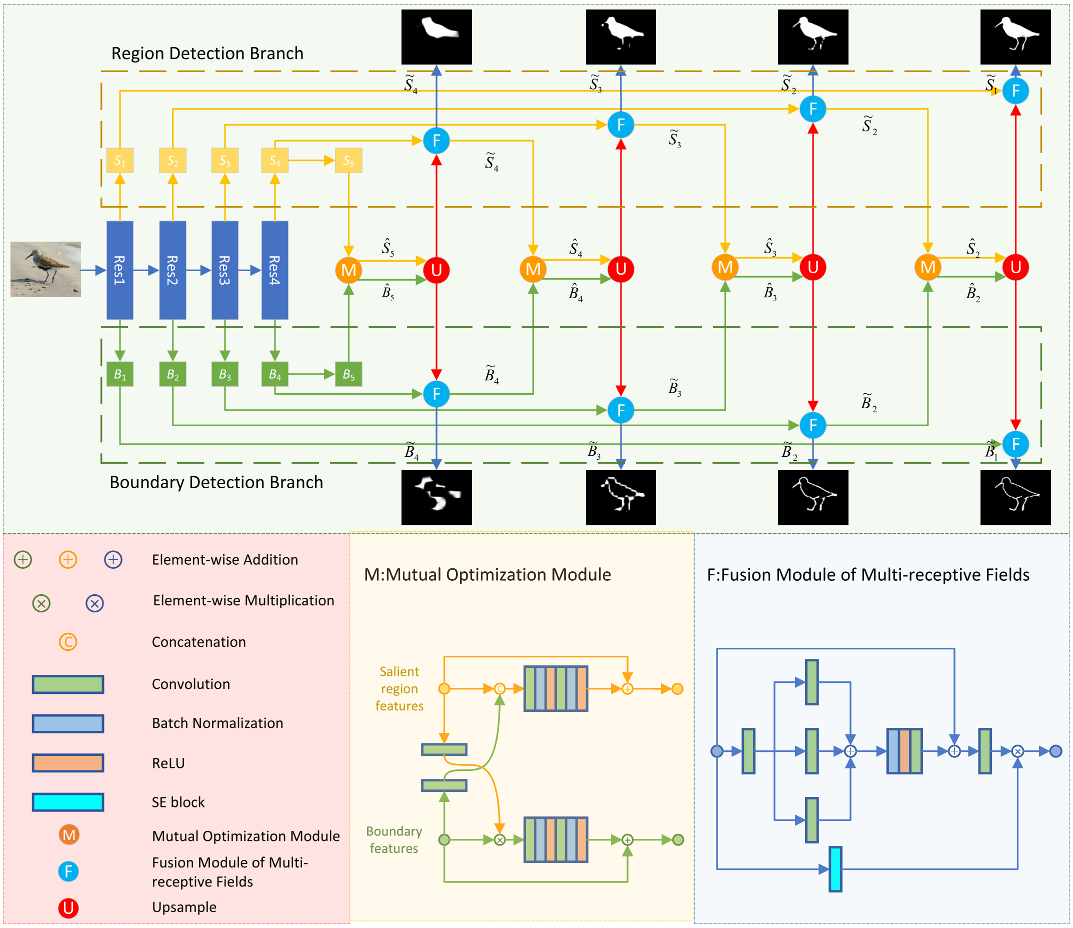


FIGURE 2. **The pipeline of DIMONet. The extracted salient region and boundary features from the backbone are denoted as S_i and B_i, where $i \in \{1, 2, 3, 4\}$ indexes the feature level. An additional 3×3 convolutional layer is applied to S_4 and B_4 to obtain the global information S_5 and B_5, respectively. These two kinds of features are processed in parallel to form the region detection branch and boundary detection branch. In the feature decoder, there are four stages and each stage is a refinement of the output of the previous stage. In the first stage, we first use the MOM to generate mutually optimized features $\hat{S}_5$ and $\hat{B}_5$. Then these features are upsampled and fused with the corresponding features of the encoder to obtain $\tilde{S}_4$ and $\tilde{B}_4$, which are sent to the next MOM for further mutual optimization. Finally, a 1×1 convolution layer is applied to each feature $\tilde{S}_i$ and $\tilde{B}_i$ to generate a corresponding saliency map $\bar{S}_i$ and a boundary map $\bar{B}_i$, where $i \in \{1, 2, 3, 4\}$.**

concatenation. The whole process of the MOM can be formalized as:

$$\begin{cases} S_{\text{out}} = \text{Conv}_3(\text{Conv}_3(\text{Cat}(B_{\text{in}}, S_{\text{in}}))) + S_{\text{in}}, \\ B_{\text{out}} = \text{Conv}_3(\text{Conv}_3(S_{\text{in}} \otimes B_{\text{in}})) + B_{\text{in}}, \end{cases} \tag{5}$$

where S_{in}, B_{in} and S_{out}, B_{out} are the inputs and outputs of the MOM, $\text{Cat}(\cdot)$ stands for concatenation and $\otimes$ for element-wise multiplication.

After applying the MOM, the features of the region detection branch and the boundary detection branch become neat and recognizable.

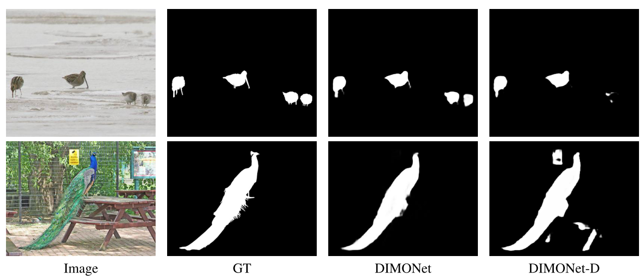


FIGURE 3. **Prediction results by the proposed DIMONet and DIMONet-D. DIMONet-D means using 3×3 convolution to fuse the features of the two tasks directly.**

C. FUSION MODULE OF MULTI-RECEPTIVE FIELDS

The utilization of MOM allows the features of the region detection branch and the boundary detection branch to become more discriminative. For better integration of these refined features generated by the MOM with multi-level features, a FMMF that contains a series of convolution layers with different kernel sizes and a squeeze and excitation (SE) block [36] is used and its structure is shown in Fig. 2. The FMMF firstly concatenates the refined features $\hat{x}$ produced by the former MOM with the corresponding stage features x in the primitive feature set along the channel axis. Then, a 1×1

convolution is applied for expanding the dimension of channels by four times, followed by three convolutions with kernel sizes of 3, 5, and 7 connected in parallel for multi-receptive fields feature extraction. This can be formulated as

$$\begin{cases} y_0 = \mathrm{Conv}_1(\mathrm{Cat}(\hat{x}, x)), \\ y_1 = \mathrm{Conv}_3(y_0), \\ y_2 = \mathrm{Conv}_5(y_0), \\ y_3 = \mathrm{Conv}_7(y_0). \end{cases} \tag{6}$$

In practice, the vanilla convolutions are replaced with the asymmetric convolutions [37] for efficiency. A 1×1 convolution is used to transform channels to the same number as the input and a residual connection is used for better optimization, i.e.,

$$y_4 = \mathrm{Conv}_1(\mathrm{ReLU}(\mathrm{BN}(y_1 + y_2 + y_3))) + x, \tag{7}$$

where BN and ReLU are the abbreviations of the batch normalization and the ReLU activation function.

However, the noises also exist in multi-level features. In order to focus the FMMF on useful features, the SE block, applied to the input x, is used to calculate an attention vector v. Then, the attention vector v acts on the intermediate feature y_4 by the element-wise multiplication to obtain the final result $\mathcal{Z}$. This process can be formalized as:

$$v = \mathrm{SE}(x), \quad \mathcal{Z} = v \otimes \mathrm{Conv}_1(y_4). \tag{8}$$

By gradually aggregating from high-level features to low-level features, the model can learn both the boundary information of low-level features and the salient region information of high-level features.

D. LOSS FUNCTION

The saliency labels and their corresponding boundary labels are used to train the proposed DIMONet. As in previous approaches, the binary cross entropy (BCE) loss

$$\mathcal{L}^{\mathrm{BCE}} = -\sum_{x,y} G_{x,y}\log(S_{x,y}) + (1 - G_{x,y})\log(1 - S_{x,y}), \tag{9}$$

is used to calculate the error pixel-wise between the ground truth and the prediction, where $G_{x,y} \in \{0, 1\}$ represents the label of pixel (x, y), and $S_{x,y} \in [0, 1]$ is the predicted value at pixel (x, y). Compared to the MSE and Focal loss functions, the BCE loss costs less calculation and is more suitable for binary classification tasks, such as salient object detection, than the MSE and Focal loss functions. In addition, the ablation study of loss functions also shows that using BCE loss to supervise our proposed network achieves better performance than using the MSE or Focal loss. Therefore, we utilize BCE loss to supervise the DIMONet.

However, BCE loss only focuses on the accuracy of each pixel and hence using the BCE loss cannot guide the DIMONet to learn the overall structure of the objects in the image well. Therefore, in order to make the structures between the ground truth and the prediction as similar as possible, we invoke an additional intersection over Union (IoU) loss

$$\mathcal{L}^{\mathrm{IoU}} = 1 - \frac{\sum_{x=1}^{H}\sum_{y=1}^{W} S_{x,y}G_{x,y}}{\sum_{x=1}^{H}\sum_{y=1}^{W}[S_{x,y}+G_{x,y}-S_{x,y}G_{x,y}]}, \tag{10}$$

where W and H are the width and height of saliency map G, respectively.

As mentioned above, there are four saliency maps $\bar{S} = \{\bar{S}_1, \bar{S}_2, \bar{S}_3, \bar{S}_4\}$ and four boundary maps $\bar{B} = \{\bar{B}_1, \bar{B}_2, \bar{B}_3, \bar{B}_4\}$ generated by the DIMONet. Therefore, we take the loss $\mathcal{L}^S$ of saliency maps

$$\mathcal{L}^S = \sum_{i=1}^{4}(\mathcal{L}_i^{\mathrm{BCE}} + \mathcal{L}_i^{\mathrm{IoU}}), \tag{11}$$

where $\mathcal{L}_i^{\mathrm{BCE}}$ and $\mathcal{L}_i^{\mathrm{IoU}}$ are the BCE and IoU assigned to the i-th saliency map, respectively.

The calculation of the loss $\mathcal{L}^B$ of boundary maps is similar to $\mathcal{L}^S$. As a result, the aggregate loss of the DIMONet $\mathcal{L}^{\mathrm{total}}$ is taken as:

$$\mathcal{L}^{\mathrm{total}} = \mathcal{L}^S + \mathcal{L}^B. \tag{12}$$

IV. EXPERIMENTS

A. DATASETS

To verify the performance, we first train the DIMONet on the DUTS-TR [38] and then evaluate on DUTS-TE [38], PASCAL-S [39], DUT-OMRON [11], HKU-IS [40] and ECSSD [9]. As the largest public dataset for saliency detection tasks, DUTS contains 10,553 images for training and 5,019 images for testing. PASCAL-S is a subset of the PASCAL VOC [41] dataset and consists of 850 images. DUT-OMRON is a challenging dataset containing 5,168 images. HKU-IS contains 4,447 images, most of which have salient objects more than one. ECSSD has 1,000 images selected from the Internet.

B. EVALUATION METRICS

Four evaluation metrics, mean absolute error (MAE), F-measure [42], E-measure [43] and S-measure [44], are adopted to quantitatively evaluate the performance.

Suppose S and G are a saliency map and its ground truth map. Then, the MAE is calculated as:

$$\frac{1}{W \times H}\sum_{x=1}^{W}\sum_{y=1}^{H}\left|S_{x,y} - G_{x,y}\right|. \tag{13}$$

F-measure is widely used to evaluate the performance of classification models and is calculated by the weighted harmonic mean of precision and recall. Define the precision P and the recall R as:

$$P = \frac{|S \wedge G|}{|S|}, \quad R = \frac{|S \wedge G|}{|G|}, \tag{14}$$

where $|\cdot|$ stands for the number of non-zero binary pixels. Then, the F-measure is calculated as:

$$\frac{(1 + \beta^2) \times P \times R}{\beta^2 \times P + R}, \tag{15}$$

where β^2 is set to 0.3 as suggested in [42]. To be fair, the average F-measure ($\bar{F_\beta}$) of each method on different datasets is used to measure the performance of different methods.

E-measure is an approach to measuring the similarity of two maps. Denote the mean values of S and G as μ_S and μ_G. Then, define the biases

$$M_S = S - \mu_S, \quad M_G = G - \mu_G, \tag{16}$$

the alignment matrix

$$M^{\mathrm{A}} = \frac{2M_S \circ M_G}{M_S \circ M_S + M_G \circ M_G}, \tag{17}$$

where $\circ$ is the Hadamard product, and the enhanced alignment matrix

$$M^{\mathrm{E}} = f(M^{\mathrm{A}}), \tag{18}$$

with $f(x) = (1+x)^2/4$. Therefore, the E-measure is calculated as:

$$\frac{1}{W \times H}\sum_{x=1}^{W}\sum_{y=1}^{H} M^{\mathrm{E}}_{x,y}. \tag{19}$$

The mean E-measure ($\bar{E}$) among all the thresholds that binarize the predicted map for each method is recorded.

S-measure is an approach to measuring the structural similarity of the predicted map and the ground-truth map. Suppose $\bar{S}$, $\bar{G}$, σ_S,σ_G and σ_{SG} are the mean, standard deviations of covariance of S and G and the covariance between them. Then, the structural similarity index measure (ssim) can be calculated as:

$$\mathrm{ssim} = \frac{2\bar{S} \times \bar{G}}{(\bar{S})^2 + \bar{G}^2} \cdot \frac{2\sigma_S\sigma_G}{\sigma_S^2 + \sigma_G^2} \cdot \frac{\sigma_{SG}}{\sigma_S\sigma_G}. \tag{20}$$

By recursively dividing each of the predicted maps and ground-truth maps into four blocks until the total number of blocks is T and assigning a different weight w_t to each block, the region-aware structural similarity can be calculated as:

$$S_r = \sum_{t=1}^{T} w_t \times \mathrm{ssim}(t). \tag{21}$$

After that, we denote $\bar{S}_{FG}$ and σ_{FG} are the mean and standard deviations of the probability values of the foreground region of S. Similarly, $\bar{S}_{BG}$ and σ_{BG} are the mean and standard deviations of the probability values of the background region of S. Then, object-aware structural similarity in foreground O_{FG} and background O_{BG} can be calculated as:

$$\begin{cases} O_{FG} = \dfrac{2\bar{S}_{FG}}{(\bar{S}_{FG})^2 + 1 + 2\lambda\sigma_{FG}}, \\ O_{BG} = \dfrac{2\bar{S}_{BG}}{(\bar{S}_{BG})^2 + 1 + 2\lambda\sigma_{BG}}, \end{cases} \tag{22}$$

where λ is a balance factor.

Therefore, the object-aware structural similarity S_o is calculated as:

$$S_o = \mu O_{FG} + (1-\mu) O_{BG}, \tag{23}$$

where μ is the ratio of foreground area in G to image area.

Finally, the S-measure (S_m) is formulate as:

$$S_m = 0.5 S_o + 0.5\, S_r. \tag{24}$$

C. IMPLEMENTATION DETAILS

Our DIMONet is implemented under the PyTorch framework [45]. All training and testing experiments are conducted with a single NVIDIA RTX 2080Ti GPU. A pre-trained ResNet-50 is utilized to initialize the parameters of the backbone of the DIMONet, and the parameters of the rest of the network are randomly initialized. The maximum learning rate is set to 5×10^{-5} for the ResNet-50 backbone and 5×10^{-4} for the rest of the network. Like the previous practice [34], warm-up and linear decay strategies are adopted to accelerate the convergence of the network. The size of the input image is set to 352×352 for training and testing. Horizontal flip, random crop and multi-scale input images are used for data augmentation. We use the Adam optimizer with a momentum of 0.9 and a weight decay of 5×10^{-4} for end-to-end training. The batchsize is set to 32 and the maximum epoch is set to 50.

D. PERFORMANCE COMPARISON

We compare with 18 state-of-the-art SOD methods, including PiCANet [46], AFNet [33], BANet [31], EGNet [17], SCRN [18], PoolNet [47], CPD [48], BASNet [32], GateNet [49], U2Net [50], DFI [51], GCPANet [30], ITSD [16], DNA [52], PurNet [53], CTDNet [54], EDN [55] and SHNet [56] to verify the effectiveness of our method. The saliency maps published by the authors of the above methods are used and evaluated using the same validation code for a fair comparison.

1) QUANTITATIVE COMPARISON

Multiple evaluation indicators are used to measure our method and the above state-of-the-art methods. From Fig. 4 and Fig. 5, we can see that the F_β curves and PR curves of our method are higher and smoother than others. Besides, Table 1 gives more detailed comparisons of the MAE, $\bar{F_\beta}$, $\bar{E}$ and S_m on five datasets. It is observed that our method achieves better performance on most metric scores. As for the other boundary-aware methods, such as EGNet, SCRN and ITSDNet, the DIMONet achieves 1.76% and 0.73% average percentage gains in terms of $\bar{F_\beta}$ and $\bar{E}$. In a word, the results of MAE show that the saliency map generated by the DIMONet is more similar to the ground truth than others. At the same time, the performances of $\bar{F_\beta}$, E_m and S_m indicate that the DIMONet can more accurately divide the salient objects from context.

2) QUALITATIVE COMPARISON

For the qualitative comparison of the DIMONet, some saliency maps generated by our method and other methods are visualized in Fig. 6. We observe that the DIMONet can accurately segment salient objects with clear boundaries from various complex scenes, including small

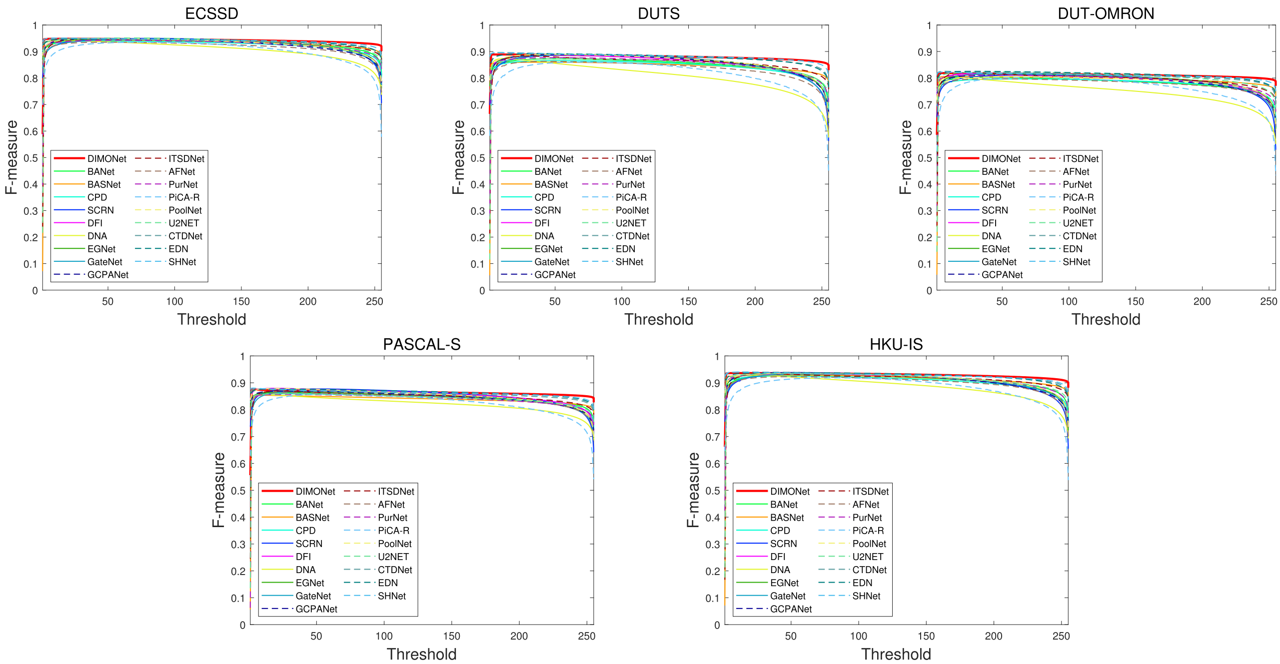


FIGURE 4. Comparison of the proposed method with 18 state-of-the-art methods in terms of F-measure curves over different thresholds on five datasets.

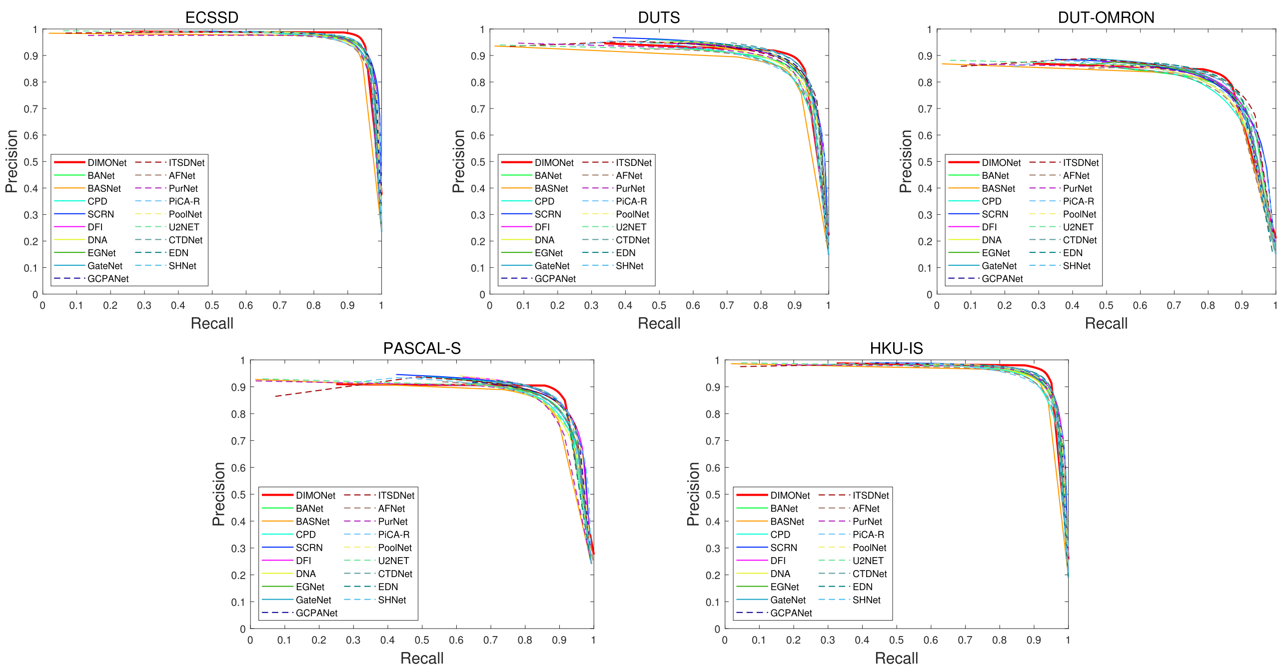


FIGURE 5. Comparison of the proposed method with 18 state-of-the-art methods in terms of PR curves over different thresholds on five datasets.

objects (1st and 2nd rows), object reflection (3rd row), objects with complex boundaries (4th and 5th rows) and objects with low contrast (6th row). Compared with other counterparts, our method can generate saliency maps with higher consistency and clearer boundaries, and is more suitable for various complex scenes.

3) BOUNDARY COMPARISON

In order to verify the superiority of our method in salient object boundary detection, we conduct the quantitative and qualitative comparisons with EGNet, ITSD, PoolNet and SCRN. From Table 2, we can see that our method achieves the best performance. Specially, compared with other four

TABLE 1. Performance comparison with 18 state-of-the-art methods over five datasets. "-" means the results cannot be obtained. The best two results are shown in red and green, respectively. "↑" denotes that higher is better, and "↓" denotes that lower is better.

Methods	GFLOPs	Param	ECSSD				PASCAL-S				HKU-IS				DUT-OMRON				DUTS-TE			
			$\bar{F}_\beta \uparrow$	MAE ↓	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE ↓	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE ↓	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE ↓	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE ↓	$\bar{E} \uparrow$	$S_m \uparrow$
PiCANet_{18}	59.79	47.22	.886	.046	.913	.913	.792	.074	.832	.848	.870	.043	.936	.905	.717	.065	.841	.826	.759	.051	.862	.860
AFNet_{19}	14.39	26.81	.908	.042	.918	.913	.738	.071	.853	.849	.888	.036	.942	.905	.738	.057	.853	.826	.793	.046	.879	.867
BANet_{19}	41.62	45.35	.923	.035	.928	.915	.823	.069	.852	.850	.900	.032	.950	.909	.746	.059	.860	.835	.815	.040	.892	.870
EGNet_{19}	294.91	111.66	.920	.037	.927	.925	.817	.073	.848	.852	.901	.031	.950	.917	.755	.053	.867	.841	.815	.039	.891	.886
SCRN_{19}	15.08	25.23	.918	.038	.926	.927	.826	.064	.857	.865	.896	.034	.949	.916	.746	.056	.863	.837	.809	.040	.888	.885
PoolNet_{19}	108.18	69.56	.915	.039	.924	.921	.815	.074	.848	.849	.899	.032	.949	.915	.747	.056	.863	.836	.809	.040	.889	.883
CPD_{19}	17.75	47.85	.917	.037	.925	.918	.820	.070	.849	.848	.891	.034	.944	.905	.747	.056	.866	.825	.805	.043	.887	.869
BASNet_{19}	240.71	87.06	.880	.037	.921	.916	.771	.075	.846	.838	.895	.032	.946	.909	.756	.056	.869	.836	.791	.048	.884	.865
GateNet_{20}	136.22	128.63	.916	.040	.924	.920	.819	.067	.851	.858	.899	.033	.949	.915	.746	.055	.862	.838	.807	.040	.889	.885
$\text{U}^2\text{Net}_{20}$	71.19	44.01	.892	.033	.924	.921	.770	.073	.842	.845	.896	.031	.948	.914	.761	.054	.871	.839	.792	.045	.886	.873
DFI_{20}	26.96	29.63	.920	.038	.924	.920	.830	.064	.855	.860	.901	.031	.951	.919	.752	.055	.865	.840	.814	.039	.892	.887
GCPANet_{20}	65.72	67.06	.919	.035	.920	.927	.827	.061	.847	.864	.898	.031	.949	.920	.748	.056	.860	.839	.817	.038	.891	.890
ITSDNet_{20}	23.82	26.08	.895	.035	.927	.925	.785	.071	.850	.859	.899	.031	.952	.917	.756	.061	.863	.840	.804	.041	.895	.884
DNA_{21}	-	-	.906	.042	.919	.914	.831	.081	.837	.835	.905	.035	.933	.905	.748	.063	.828	.818	.806	.046	.853	.860
PurNet_{21}	48.44	35.53	.895	.038	.927	.918	.770	.074	.849	.833	.896	.031	.949	.915	.747	.052	.872	.838	.792	.041	.894	.877
CTDNet_{22}	6.13	11.83	.925	.033	.925	.925	.840	.061	.861	.856	.910	.027	.953	.921	.765	.052	.870	.844	.850	.035	.907	.890
EDN_{22}	17.17	42.85	.924	.033	.929	.925	.845	.061	.862	.856	.905	.028	.951	.923	.766	.052	.870	.844	.851	.036	.908	.892
SHNet_{22}	54.79	38.96	.929	.033	.930	.927	.848	.059	.870	.859	.916	.027	.955	.924	.785	.052	.889	.844	.857	.035	.909	.892
Ours	12.54	24.88	.926	.034	.930	.928	.827	.060	.857	.867	.911	.027	.955	.925	.760	.052	.870	.846	.845	.035	.909	.895

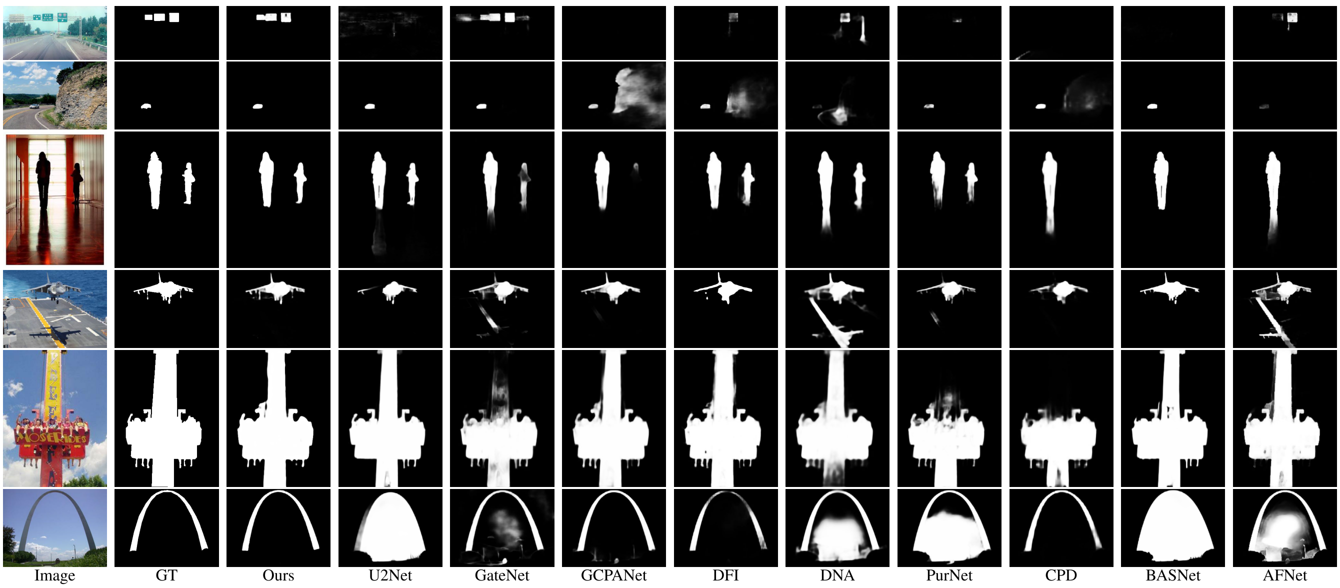


FIGURE 6. Qualitative comparison of the proposed method with existing state-of-the-art SOD methods under several challenging contexts.

methods, our method gains average improvements of 23.3%, 29.8% and 22.2% in terms of $\bar{F}_\beta$, MAE, and $\bar{E}$, respectively. In addition, in Fig. 7, our method is able to extract clear and accurate boundaries in the case of different kinds of salient objects (the first two rows). Besides, PoolNet, EGNet, ITSD and SCRN are disturbed by the background when the salient objects are small or the background is complex (the third and fourth rows), making the extracted boundaries inaccurate, incomplete or unclear. Differently, our method can extract the boundaries of salient objects accurately. Even when the salient objects have complex boundaries (the last two rows), our method can still extract them correctly and clearly.

E. ABLATION STUDY

In this section, a series of experiments are conducted to analyze the proposed modules, MOM and FMMF, and network architecture. The MAE, $\bar{F}_\beta$, $\bar{E}_m$ and S_m are used as the evaluation metrics.

TABLE 2. Quantitative comparisons of boundary maps generated by our method and other four state-of-the-art methods on DUTS-TE and ECSSD. The best result is marked in bold.

Methods	DUTS-TE				ECSSD			
	$\bar{F}_\beta \uparrow$	MAE ↓	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE ↓	$\bar{E} \uparrow$	$S_m \uparrow$
EGNet	.217	.087	.460	.569	.229	.097	.472	.565
ITSD	.332	.037	.726	.632	.349	.043	.708	.641
PoolNet	.147	.176	.381	.482	.167	.175	.396	.491
SCRN	.424	.031	.669	.635	.458	.035	.708	.622
Ours	**.519**	**.024**	**.867**	**.769**	**.569**	**.022**	**.886**	**.809**

1) THE EFFECTIVENESS OF THE MOM AND FMMF

To verify the effectiveness of the MOM and FMMF, these modules are gradually joined into a network with a basic encoder and a basic decoder. The basic encoder is a ResNet-50, and the basic decoder consists of multiple 3×3 convolution layers and 1×1 convolution layers. Like other U-shaped networks, the multi-level features extracted by the encoder are firstly processed by concatenation along the channel axis, and then 3×3 convolution layers for feature

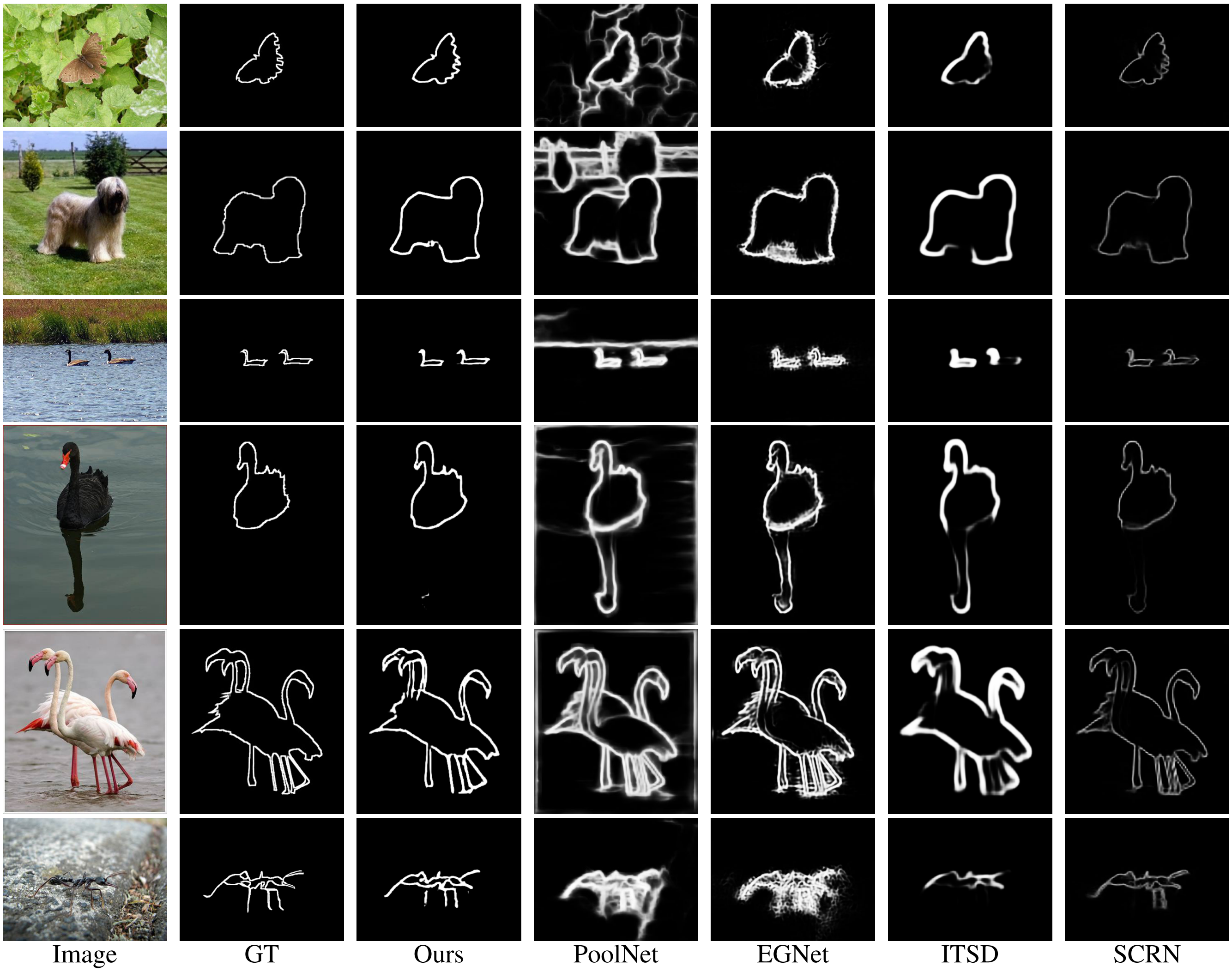


FIGURE 7. Visualization of boundary maps generated by the proposed method and other four state-of-the-art methods. Our method is able to generate clear and complete boundary maps of salient objects.

TABLE 3. Ablation study of the proposed modules on five datasets. The best result is marked in bold.

Methods	ECSSD				PASCAL-S				HKU-IS				DUT-OMRON				DUTS-TE			
	$F_\beta \uparrow$	MAE $\downarrow$	$E \uparrow$	$S_m \uparrow$	$F_\beta \uparrow$	MAE $\downarrow$	$E \uparrow$	$S_m \uparrow$	$F_\beta \uparrow$	MAE $\downarrow$	$E \uparrow$	$S_m \uparrow$	$F_\beta \uparrow$	MAE $\downarrow$	$E \uparrow$	$S_m \uparrow$	$F_\beta \uparrow$	MAE $\downarrow$	$E \uparrow$	$S_m \uparrow$
Base	.878	.038	.927	.905	.817	.066	.843	.830	.892	.031	.949	.900	.735	.061	.854	.810	.810	.040	.891	.857
Base + MOM	.906	.035	.924	.915	.830	.063	.856	.846	.915	.029	.952	.913	.759	.057	.860	.825	.830	.038	.906	.878
Base + FMMF	.904	.035	.924	.916	.824	.061	.851	.850	.903	.028	.952	.916	.757	.055	.867	.834	.819	.037	.903	.881
Base + MOM + FMMF	**.926**	**.034**	**.930**	**.928**	**.827**	**.060**	**.857**	**.867**	**.911**	**.027**	**.955**	**.925**	**.760**	**.052**	**.870**	**.846**	**.845**	**.035**	**.909**	**.895**

TABLE 4. Ablation study of the proposed network on five datasets. The best result is marked in bold.

Methods	ECSSD				PASCAL-S				HKU-IS				DUT-OMRON				DUTS-TE			
	$F_\beta \uparrow$	MAE $\downarrow$	$E \uparrow$	$S_m \uparrow$	$F_\beta \uparrow$	MAE $\downarrow$	$E \uparrow$	$S_m \uparrow$	$F_\beta \uparrow$	MAE $\downarrow$	$E \uparrow$	$S_m \uparrow$	$F_\beta \uparrow$	MAE $\downarrow$	$E \uparrow$	$S_m \uparrow$	$F_\beta \uparrow$	MAE $\downarrow$	$E \uparrow$	$S_m \uparrow$
SB	.887	.035	.924	.920	.801	.063	.846	.848	.899	.030	.948	.916	.741	.059	.857	.826	.813	.041	.898	.878
DB	.912	.036	.924	.917	.813	.064	.855	.846	.901	.028	.952	.919	.752	.057	.861	.831	.825	.038	.902	.883
w/o HS	.910	.033	.925	.924	.812	.062	.855	.858	.902	.028	.953	.921	.753	.055	.862	.837	.823	.038	.903	.888
DIMONet	**.926**	**.034**	**.930**	**.928**	**.827**	**.060**	**.857**	**.867**	**.911**	**.027**	**.955**	**.925**	**.760**	**.052**	**.870**	**.846**	**.845**	**.035**	**.909**	**.895**

integration and finally 1×1 convolution layers for generating saliency maps. We gradually displace the 3×3 convolution layers with MOM and FMMF. The results are shown in Table 3. It can be seen that involving each module can improve the performance of the model in comparison to the basic model. When all modules are involved, the best performances are obtained, which demonstrates the necessity and effectiveness of each module.

TABLE 5. Ablation study of the loss functions on five datasets. The best result is marked in bold.

Loss function(s)	ECSSD				PASCAL-S				HKU-IS				DUT-OMRON				DUTS-TE			
	$\bar{F}_\beta \uparrow$	MAE $\downarrow$	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE $\downarrow$	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE $\downarrow$	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE $\downarrow$	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE $\downarrow$	$\bar{E} \uparrow$	$S_m \uparrow$
Focal	.901	.047	.909	.903	.805	.073	.839	.840	.885	.041	.932	.916	.741	.69	.849	.821	.816	.046	.882	.883
MSE	.881	.065	.898	.916	.788	.081	.833	.826	.848	.049	.927	.907	.729	.072	.830	.810	.799	.050	.875	.856
BCE	.909	.039	.914	.915	.812	.062	.843	.861	.900	.030	.937	.914	.749	.059	.858	.838	.830	.039	.895	.877
Focal + IoU	.913	.038	.922	.916	.820	.069	.854	.849	.902	.037	.944	**.926**	.753	.063	.860	.831	.833	.039	.900	.893
MSE + IoU	.902	.043	.916	**.930**	.818	.070	.844	.856	.889	.036	.937	.919	.741	.062	.850	.834	.823	.044	.889	.892
BCE + IoU	**.926**	**.034**	**.930**	.928	**.827**	**.060**	**.857**	**.867**	**.911**	**.027**	**.955**	.925	**.760**	**.052**	**.870**	**.846**	**.845**	**.035**	**.909**	**.895**

TABLE 6. Ablation study of the effect of the MOM number on five datasets. The best result is marked in bold.

Number	ECSSD				PASCAL-S				HKU-IS				DUT-OMRON				DUTS-TE			
	$\bar{F}_\beta \uparrow$	MAE $\downarrow$	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE $\downarrow$	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE $\downarrow$	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE $\downarrow$	$\bar{E} \uparrow$	$S_m \uparrow$	$\bar{F}_\beta \uparrow$	MAE $\downarrow$	$\bar{E} \uparrow$	$S_m \uparrow$
$N = 1$	.906	.037	.925	.916	.820	.063	.851	.851	.904	.030	.951	.916	.755	.057	.867	.835	.820	.038	.902	.879
$N = 2$	.910	.036	.926	.920	.822	.062	.853	.855	.906	.028	.951	.918	.756	.056	.868	.839	.830	.036	.905	.884
$N = 3$	.919	.034	.928	.926	.825	.060	.857	.863	.910	.027	.953	.922	.759	.053	.870	.844	.839	.036	.908	.891
$N = 4$	**.926**	**.034**	**.930**	**.928**	**.827**	**.060**	**.857**	**.867**	**.911**	**.027**	**.955**	**.925**	**.760**	**.052**	**.870**	**.846**	**.845**	**.035**	**.909**	**.895**

In addition, experiments are conducted to verify the effect of the MOM number and the results are shown in Table 6. Compared with the last second row in Table 3, some evaluation indicators decrease when the features of the salient regions and the boundaries are refined once by the MOM. The reason is that the first refined features are high-level features. For the high-level features of the boundaries, they only highlight the around of salient objects and hence mislead the model after mutually being refined by the MOM with the region features. As the times of optimization increase, different level features of the boundaries are utilized to refine the region features, and hence the model can effectively detect the salient objects and all evaluation indicators improve. When the number of the MOM is four, our method achieves the best results.

2) THE EFFECTIVENESS OF THE PROPOSED NETWORK

A quantitative analysis of the proposed network and different architectures is made to verify the effectiveness of the proposed network architecture, and the results are shown in Table 4, where SB refers to the single-branch network only containing the region detection branch, DB refers to the dual-branch network without mutual optimization, and HS refers to hierarchical supervision. It is shown that the performance of region detection can be enhanced by adding a boundary detection branch. In addition, the network with hierarchical supervision performs obviously better.

3) THE EFFECTIVENESS OF THE LOSS FUNCTIONS

In order to verify the superiority of the loss functions we used, several experiments are designed. Under the same training strategy, different loss functions are used to supervise our proposed network. The results are shown in Table 5. It can be seen that using the BCE loss function to supervise the network achieves better performance than the MSE and Focal loss functions, which is consistent with the fact that the BCE loss functions are commonly used to train salient object detection networks. In addition, all four evaluation indicators significantly increase when the IoU loss function is integrated into different loss functions. This demonstrates the effectiveness of IoU loss. Further, when the BCE and IoU loss functions are used for supervision, the best performance is achieved.

V. CONCLUSION

In order to sharpen the blurring boundaries of salient objects in existing SOD methods, in this paper, we introduce the DIMONet. Unlike previous networks, the DIMONet has a salient region detection branch and a boundary detection branch of the same structures to focus equally on the boundaries and the regions. Besides, in order to refine the features of the two branches, the mutual optimization module is designed to mutually optimize the features based on the intrinsic relationship between them. Next, to make the features of the two branches more representative, the fusion module of multi-receptive fields is designed to fuse the features refined by the mutual optimization module and the multi-level features. With multiple optimizations of the mutual optimization modules and the fusion modules of multi-receptive fields from high-level features to low-level features, the features extracted from the boundary detection branch become clean and hence the salient maps of the salient region detection branch obtain clear boundaries. The results of experiments on five benchmark datasets show that our method is superior to the 18 state-of-the-art methods and is more suitable for various complex scenes.

REFERENCES

[1] L. Jiang, M. Xu, X. Wang, and L. Sigal, "Saliency-guided image translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16509–16518.

[2] P. Zhang, T. Zhuo, W. Huang, K. Chen, and M. Kankanhalli, "Online object tracking based on CNN with spatial–temporal saliency guided sampling," *Neurocomputing*, vol. 257, pp. 115–127, Sep. 2017.

[3] P. Zhang, W. Liu, D. Wang, Y. Lei, H. Wang, and H. Lu, "Non-rigid object tracking via deep multi-scale spatial–temporal discriminative saliency maps," *Pattern Recognit.*, vol. 100, Apr. 2020, Art. no. 107130.

[4] Y. Yao, T. Chen, G.-S. Xie, C. Zhang, F. Shen, Q. Wu, Z. Tang, and J. Zhang, "Non-salient region object mining for weakly supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 2623–2632.

[5] W. Shimoda and K. Yanai, "Weakly supervised semantic segmentation using distinct class specific saliency maps," *Comput. Vis. Image Understand.*, vol. 191, Feb. 2020, Art. no. 102712.
[6] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.
[7] J. Zhang, K. A. Ehinger, J. Ding, and J. Yang, "A prior-based graph for salient object detection," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 1175–1178.
[8] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, Mar. 2015.
[9] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1155–1162.
[10] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 733–740.
[11] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3166–3173.
[12] X. Zhang, T. Wang, J. Qi, H. Lu, and G. Wang, "Progressive attention guided recurrent network for salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 714–722.
[13] X. Hu, L. Zhu, J. Qin, C.-W. Fu, and P.-A. Heng, "Recurrently aggregating deep features for salient object detection," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2018, vol. 32, no. 1, pp. 1–8.
[14] L. Zhang, J. Dai, H. Lu, Y. He, and G. Wang, "A bi-directional message passing model for salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1741–1750.
[15] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 202–211.
[16] H. Zhou, X. Xie, J.-H. Lai, Z. Chen, and L. Yang, "Interactive two-stream decoder for accurate and fast saliency detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9141–9150.
[17] J. Zhao, J.-J. Liu, D.-P. Fan, Y. Cao, J. Yang, and M.-M. Cheng, "EGNet: Edge guidance network for salient object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8779–8788.
[18] Z. Wu, L. Su, and Q. Huang, "Stacked cross refinement network for edge-aware salient object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7264–7273.
[19] W. Wang, S. Zhao, J. Shen, S. C. H. Hoi, and A. Borji, "Salient object detection with pyramid attention and salient edges," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1448–1457.
[20] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 234–250.
[21] M. Zhuge, D.-P. Fan, N. Liu, D. Zhang, D. Xu, and L. Shao, "Salient object detection via integrity learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3738–3752, Mar. 2022.
[22] H. Xiao, J. Feng, Y. Wei, M. Zhang, and S. Yan, "Deep salient object detection with dense connections and distraction diagnosis," *IEEE Trans. Multimedia*, vol. 20, no. 12, pp. 3239–3251, Dec. 2018.
[23] N. Liu and J. Han, "DHSNet: Deep hierarchical saliency network for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 678–686.
[24] T. Wang, A. Borji, L. Zhang, P. Zhang, and H. Lu, "A stagewise refinement model for detecting salient objects in images," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4019–4028.
[25] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. Torr, "Deeply supervised salient object detection with short connections," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3203–3212.
[26] J. Wei and B. Zhong, "Saliency detection using fully convolutional network," in *Proc. Chin. Autom. Congr. (CAC)*, Nov. 2018, pp. 825–841.
[27] Z. Deng, X. Hu, L. Zhu, X. Xu, J. Qin, G. Han, and P.-A. Heng, "R^3Net: Recurrent residual refinement network for saliency detection," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Menlo Park, CA, USA, Jul. 2018, pp. 684–690.
[28] Y. Pang, X. Zhao, L. Zhang, and H. Lu, "Multi-scale interactive network for salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9413–9422.
[29] J. Wei, S. Wang, and Q. Huang, "F^3Net: Fusion, feedback and focus for salient object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 12321–12328.
[30] Z. Chen, Q. Xu, and R. Cong, "Global context-aware progressive aggregation network for salient object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 10599–10606.
[31] J. Su, J. Li, Y. Zhang, C. Xia, and Y. Tian, "Selectivity or invariance: Boundary-aware salient object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3799–3808.
[32] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "BASNet: Boundary-aware salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7479–7489.
[33] M. Feng, H. Lu, and E. Ding, "Attentive feedback network for boundary-aware salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1623–1632.
[34] J. Wei, S. Wang, Z. Wu, C. Su, Q. Huang, and Q. Tian, "Label decoupling framework for salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13025–13034.
[35] T. Wang, L. Zhang, S. Wang, H. Lu, G. Yang, X. Ruan, and A. Borji, "Detect globally, refine locally: A novel approach to saliency detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3127–3135.
[36] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
[37] X. Ding, Y. Guo, G. Ding, and J. Han, "ACNet: Strengthening the kernel skeletons for powerful CNN via asymmetric convolution blocks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1911–1920.
[38] 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 136–145.
[39] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 280–287.
[40] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5455–5463.
[41] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Sep. 2009.
[42] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1597–1604.
[43] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, and A. Borji, "Enhanced-alignment measure for binary foreground map evaluation," 2018, *arXiv:1805.10421*.
[44] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4548–4557.
[45] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, and L. Antiga, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–12.
[46] N. Liu, J. Han, and M.-H. Yang, "PiCANet: Learning pixel-wise contextual attention for saliency detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3089–3098.
[47] 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 *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3917–3926.
[48] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3907–3916.
[49] X. Zhao, Y. Pang, L. Zhang, H. Lu, and L. Zhang, "Suppress and balance: A simple gated network for salient object detection," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 35–51.
[50] X. Qin, Z. Zhang, C. Huang, M. Dehghan, O. R. Zaiane, and M. Jagersand, "U^2-Net: Going deeper with nested U-structure for salient object detection," *Pattern Recognit.*, vol. 106, Oct. 2020, Art. no. 107404.

[51] J.-J. Liu, Q. Hou, and M.-M. Cheng, "Dynamic feature integration for simultaneous detection of salient object, edge, and skeleton," *IEEE Trans. Image Process.*, vol. 29, pp. 8652–8667, 2020.

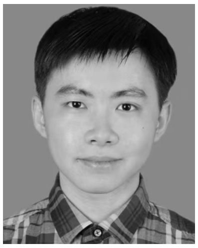
[52] Y. Liu, M.-M. Cheng, X.-Y. Zhang, G.-Y. Nie, and M. Wang, "DNA: Deeply supervised nonlinear aggregation for salient object detection," *IEEE Trans. Cybern.*, vol. 52, no. 7, pp. 6131–6142, Jul. 2022.

[53] J. Li, J. Su, C. Xia, M. Ma, and Y. Tian, "Salient object detection with purificatory mechanism and structural similarity loss," *IEEE Trans. Image Process.*, vol. 30, pp. 6855–6868, 2021.

[54] Z. Zhao, C. Xia, C. Xie, and J. Li, "Complementary trilateral decoder for fast and accurate salient object detection," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 4967–4975.

[55] Y.-H. Wu, Y. Liu, L. Zhang, M.-M. Cheng, and B. Ren, "EDN: Salient object detection via extremely-downsampled network," *IEEE Trans. Image Process.*, vol. 31, pp. 3125–3136, 2022.

[56] W. Zhang, L. Zheng, H. Wang, X. Wu, and X. Li, "Saliency hierarchy modeling via generative kernels for salient object detection," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 570–587.



ZIJUN CHEN was born in 1999. He received the bachelor's degree from the Guangdong University of Technology, in 2021, where he is currently pursuing the M.S. degree with the College of Computer Science supervised by Prof. Yinwei Zhan. His research interests include computer vision and machine learning.

YINWEI ZHAN (Member, IEEE) was born in 1966. He received the M.S. degree in applied mathematics from Jilin University, Jilin, China, in 1989, and the Ph.D. degree in applied mathematics from the Dalian University of Technology, Dalian, China, in 1992. He is currently a Professor and a Ph.D. Supervisor with the Guangdong University of Technology, Guangzhou, China. He has published more than 100 articles in his research areas. His research interests include image processing, wavelet analysis, pattern recognition, and computer vision.

SHANGLEI GAO was born in 1998. He received the bachelor's degree from Zaozhuang University, in 2021. He is currently pursuing the M.S. degree with the College of Computer Science, Guangdong University of Technology, supervised by Prof. Yinwei Zhan. His research interests include computer vision and machine learning.