Salient Object Detection via Integrity Learning

Mingchen Zhuge, Deng-Ping Fan†, Nian Liu, Dingwen Zhang, Dong Xu *Fellow, IEEE*, and Ling Shao *Fellow, IEEE*

Abstract—Although current salient object detection (SOD) works have achieved significant progress, they are limited when it comes to the integrity of the predicted salient regions. We define the concept of integrity at both a micro and macro level. Specifically, at the micro level, the model should highlight all parts that belong to a certain salient object. Meanwhile, at the macro level, the model needs to discover all salient objects in a given image. To facilitate integrity learning for SOD, we design a novel **I**ntegrity **Co**gnition **N**etwork (**ICON**), which explores three important components for learning strong integrity features. 1) Unlike existing models, which focus more on feature discriminability, we introduce a diverse feature aggregation (DFA) component to aggregate features with various receptive fields (*i.e.*, kernel shape and context) and increase feature diversity. Such diversity is the foundation for mining the integral salient objects. 2) Based on the DFA features, we introduce an integrity channel enhancement (ICE) component with the goal of enhancing feature channels that highlight the integral salient objects, while suppressing the other distracting ones. 3) After extracting the enhanced features, the part-whole verification (PWV) method is employed to determine whether the part and whole object features have strong agreement. Such part-whole agreements can further improve the micro-level integrity for each salient object. To demonstrate the effectiveness of our ICON, comprehensive experiments are conducted on seven challenging benchmarks. Our ICON outperforms the baseline methods in terms of a wide range of metrics. Notably, our ICON achieves ∼10% relative improvement over the previous best model in terms of average false negative ratio (FNR), on six datasets. Codes and results are available at: https://github.com/mczhuge/ICON.

Index Terms—Saliency Detection, Salient Object Detection, Capsule Network, Integrity Learning.

1 Introduction

Salient object detection (SOD) aims to imitate the human visual perception system to capture the most significant regions in a given image [1]–[3]. As SOD is widely used in the field of computer vision, it plays a vital role in many downstream tasks, such as object detection [4], image retrieval [5], co-salient object detection [6], multi-modal matching [7], VR/AR applications [8] and semantic segmentation [9]–[11].

Traditional SOD methods [1], [12] predict saliency maps in a bottom-up manner, and are mainly based on hand-crafted features, such as color contrast [13], [14], boundary backgrounds [15], [16], or center priors [17]. To improve the representation capacity of the features used in SOD, current models employ convolutional neural network (CNN) or fully convolutional network architectures, which enable powerful feature learning processes to replace manually designed features. These methods have achieved remarkable progress and pushed the performance of SOD to a new level. More details of recent deep learning based SOD methods can be found in the surveys/benchmarks [1], [3], [18]–[20].

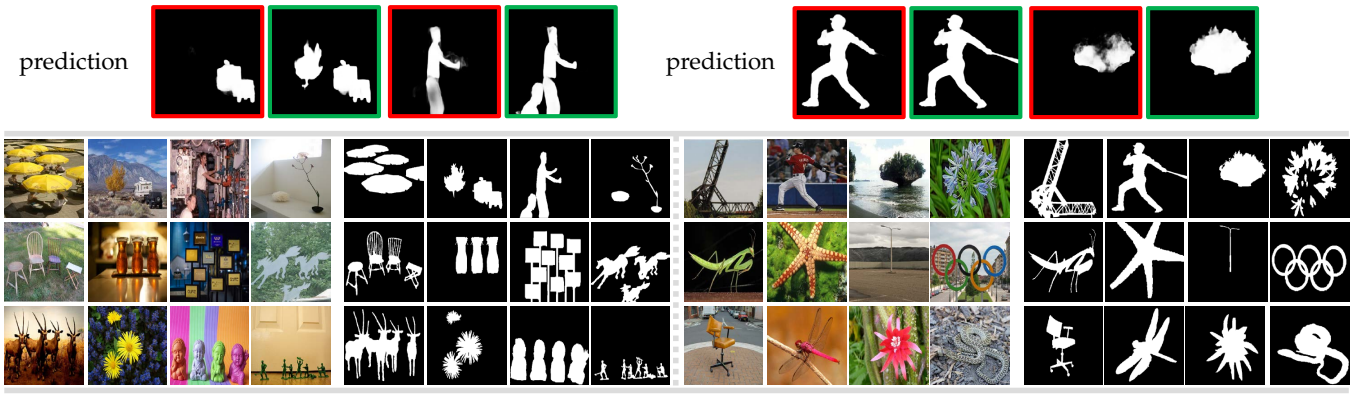
(a) macro integrity (b) micro integrity

Fig. 1. Integrity issues (*i.e.*, (a) macro integrity and (b) micro integrity) of SOD. The red and yellow contours represent ground truths. Grey regions denote predictions. The drawing style was inspired by [21].

The current success in building deep learning based salient object detectors is mainly due to the use of *multi-scale/level feature aggregation*, *contextual modeling*, *top-down modeling*, and *edge-guided learning* mechanisms. Specifically, models with a multi-scale/level feature aggregation mechanism enhance the features from different levels and scales of the network, and then fuse them to generate the final SOD results. These approaches can help discover salient objects of various sizes and highlight the salient regions under the guidance of both coarse semantics and fine details. For example, the network proposed by Zhang *et al.* [22] first adaptively fuses multi-level features at five different scales, and then use them to generate predictions. Similarly, Luo *et al.* [23] proposed to extract the global and local features at the low and high feature scales, respectively, and then fuse them to generate the final results.

Contextual modeling is another key mechanism in SOD. It helps infer the saliency of each local region by considering the surrounding contextual information. Current studies in the field of SOD usually design various attention modules

- *The first two authors share equal contributions.*
- *Deng-Ping Fan is with the College of Computer Science, Nankai University, Tianjin, China. (Email: dengpfan@gmail.com)*
- *Mingchen Zhuge and Nian Liu are with the IIAI, Abu Dhabi, UAE. (Email: mczhuge@gmail.com, liunian228@gmail.com)*
- *Dingwen Zhang is with the Brain and Artificial Intelligence Laboratory, School of Automation, Northwestern Polytechnical University, Xi'an 710072, China. (Email: zhangdingwen2006yyy@gmail.com)*
- *Dong Xu is with the School of Electrical and Information Engineering, The University of Sydney, Sydney, New South Wales Australia. (Email: dong.xu@sydney.edu.au)*
- *Ling Shao is with Terminus Group, Beijing, China (Email: ling.shao@ieee.org).*
- *† Work was done while Mingchen Zhuge was an intern at IIAI mentored by Deng-Ping Fan. Corresponding author: Nian Liu and Dingwen Zhang.*



(a) macro integrity (b) micro integrity

Fig. 2. Integrity is a good indicator of saliency prediction. Here are some samples from SOD datasets, with images (left) and ground truth (right) listed. The group of images on (a) needs the model to have the judgment of macro-level integrity, while the group on (b) needs the model to extract micro-level integrity efficiently. On the top, several predictions with different integrity qualities are presented. Note that: ■ Bad predictions, ■ Comparably high-quality predictions.

to explore such information. Specifically, Zhao *et al.* [24] proposed a pyramid feature attention network, where channel attention and spatial attention modules are introduced to process high- and low-level features, respectively, and consider the contextual information in different feature channels and spatial locations. Liu *et al.* [25] proposed to learn a pixel-wise contextual attention for SOD. Deep models learned with such an attention module can infer the relevant importance between each pixel and its global/local context location, and thus achieve the selective aggregation of contextual information.

For top-down modeling, some SOD methods adopt carefully designed decoders to gradually infer salient regions under the guidance of high-level semantic cues. For example, Wang *et al.* [26] built an iterative and cooperative inference network for SOD, where multiple top-down network streams work together with the bottom-up network streams in an iterative inference manner. Zhao *et al.* [27] proposed a gated dual-branch decoding structure to achieve cooperation among different levels of features in the top-down flow, improving the discriminability of the whole network. In [28], Liu *et al.* adopted a pyramid pooling module to build global guiding features, which they introduced to improve the top-down flow modeling.

In order to accurately predict salient object boundaries, another group of methods introduce additional network streams or learned objective functions to force the network to pay more attention to the contours that separate the salient objects from the surrounding background. For example, Wei *et al.* [29] built a label decoupling framework for SOD, which explicitly decomposes the original saliency map into a body map and a details map. Specifically, the body map concentrates on the central areas of the salient objects, while the details map focuses on the regions around the object boundaries. To improve the prediction precision of the salient contours and reduce the local noise in the salient edge predictions, Wu *et al.* [30] proposed the mutual learning strategies to separately guide the foreground contour and edge detection tasks.

Although the aforementioned mechanisms can improve the SOD performance in several aspects, the detection results produced are still not optimal. In our opinion, this is likely due to the under-exploration of another helpful and important mechanism, *i.e.*, the *integrity learning* mechanism (see Fig. 1 (a) and Fig. 1 (b)). In this work, we define the integrity learning mechanism at two levels. At the micro level, the model should focus on part-whole relevance within a single salient object. At the macro level, the model needs to identify all salient objects within the given image scene. In Fig. 2, we present some examples of the integrity qualities at both the macro and micro levels. It is clear that there exists a strong correlation between integrity and prediction performance.

In order to pursue two-level integrity, we introduce three key components in our deep neural network design. The first is diverse feature aggregation (DFA). Unlike existing models, which focus more on feature discriminability, DFA aggregates the features from various receptive fields (in terms of both the kernel shape and context) to increase their diversity. Such feature diversity provides the foundation for mining integral salient objects, since it considers richer contextual patterns to determine the activation of each neuron. The second component is called integrity channel enhancement (ICE), which aims at enhancing the feature channels that highlight the integral salient objects (at both the micro and macro levels), while suppressing the other distracting ones. As it is rare for the feature channels enhanced by ICE to perfectly match the real salient object regions, we further adopt a part-whole verification (PWV) component to judge whether the part features and whole features have a strong agreement to form the integral objects. This can help further improve integral learning at the micro level.

It is worth mentioning that some existing works have also tried to solve the macro-level integrity issue by introducing the auxiliary task for learning deep salient object detectors [31], [32]. However, these methods require additional supervision information on the number of salient objects within each image. In contrast, our newly proposed approach can tackle both macro- and micro-level integrity issues within a unified and entirely different learning framework, without additional supervision.

Our overall framework for integrity learning is called

the Integrity Cognition Network (ICON), details of which are shown in Fig. 3. Specifically, ICON first leverages five convolutional blocks for basic feature extraction. Then, it passes the deep features of each level to a diverse feature aggregation module to extract the different feature bases. Next, the diverse feature bases extracted from three adjacent feature levels are sent to an integrity channel enhancement module. Here, an integrity guiding map is generated and then used to guide the attention weighting of each feature channel. Finally, the integrity channel enhancing features produced from the three feature levels are combined and passed through the part-whole verification module, which is implemented using capsule routing layers [33]. After further verifying the agreement between the object parts and whole regions, the missing parts will be reinforced. To sum up, this paper provides three main contributions:

- We investigate the integrity issue in SOD, which is essential yet under-studied in this field.
- We introduce three key components for achieving integral SOD, namely diverse feature aggregation, integrity channel enhancement, and part-whole verification.
- We design a novel network, *i.e.*, ICON, that incorporates the three components and demonstrate its effectiveness on seven challenging datasets. In addition to its prominent performance, our approach also achieves real-time speed (∼60fps).

The remainder of the paper is organized as follows. In § 2, we discuss the related works. Then, we describe the proposed ICON in detail (see § 3). Experimental results, including performance evaluations and comparisons, are given in § 4. Finally, conclusions are drawn in § 5.

2 Related Work

Over the past several decades, a number of SOD methods have been proposed and have achieved encouraging performance on various benchmark datasets. These existing SOD methods can be roughly categorized into scale learning based, boundary learning based, and integrity learning based approaches.

2.1 Scale Learning Approaches for SOD

Scale variation is one of the major challenges for SOD. Many works have tried to handle this issue from different perspectives. Inspired by the HED model [34] for edge detection, DSS [35] introduced deep-to-shallow side-outputs with rich semantic features. This design enables shallow layers to distinguishes real salient objects from the background, while retaining high resolution. In addition, Zhang *et al.* [22] designed a multi-level feature aggregation framework and employed the hierarchical features as the saliency cues for final saliency prediction. Meanwhile, RADF [36] integrates multi-level features and refines them within each layer with a recurrent pattern. This effectively suppresses the non-salient noise in lower layers and increases the salient details of features in higher layers. Further, Zhao *et al.* [37] proposed to use the F-measure loss, which can generate precise contrastive maps to help segment multi-scale objects. To efficiently extract multi-scale features, Pang *et al.* [38] embedded self-interaction modules into their decoder units to learn the integrated information. Introduced more recently, GateNet [39] adopts Fold-ASPP to gather multi-scale saliency cues. Finally, Liu *et al.* [40] utilized a centralized information interaction strategy to simultaneously process multi-scale features.

2.2 Boundary Learning Approaches for SOD

Boundary learning plays another important role for improving SOD results. Early works used boundary learning via biologically inspired methods [14], [41], [42]. However, the results of these models exhibit undesirable blurring and usually lose entire salient areas. The more recent CNN-based approaches, which operate at the patch level (instead of pixel level), also suffer from blurred edges, due to the stride and pooling operations. To address this, several works (*e.g.*, [43]) use pre-processing technology (*e.g.*, superpixel [44]) to preserve the object boundaries, while other works, such as DSS [35], DCL [45], and PiCANet [46], employ post-processing (*e.g.*, conditional random fields [47]) to enhance edge details. The main drawback of these approaches is their slow inference speed. To learn the intrinsic edge information, PoolNet [28] employs an auxiliary module for edge detection. Besides, many other works have improved edge quality by introducing boundary-aware loss functions. For instance, the recent works [24], [48]–[51] used explicit boundary losses to guide the learning of boundary details. Considering that the cross entropy loss prefers to predict hard pixel samples (*e.g.*, 0 or 1) as non-integer values, BASNet [52] introduced a new prediction-refinement network and hybrid loss. Dealing with the inherent defect of blurry boundaries, HRSOD [53] serves as the first high-resolution SOD dataset, which explores how high-resolution data can improve the performance of the salient object edges. F3Net [54] proves that assigning larger weights to boundary pixels in loss functions is a simple way to handle boundary problems. In addition, recent works such as SCRN [55], LDF [29], VST [56] build two-stream architectures to model salient objects and boundaries simultaneously.

2.3 Integrity Learning Approaches for SOD

Integrity learning is an under-explored research topic in SOD. Among the limited existing models, DCL [45] processes contrast information at both the pixel and patch levels in order to simultaneously integrate global and local structural information. CPD [58] utilizes an effective decoder to summarize the discriminative features, and segments the integral salient objects with the aid of holistic attention modules. TSPOANet [59] models part-object relationships in SOD, and produces better wholeness and uniformity scores for segmented salient objects with the help of a capsule network. GCPANet [60] makes full use of global context to capture the relationships between multiple salient objects or regions, and alleviates the dilution effect of features. Wu *et al.* [61] used a bi-stream network combining two feature backbones and gate control units to fuse complementary information. Recently, transformers have become a hot area of research in the field of computer vision. Mao *et al.* [62] proposed a transformer-based architecture for the context learning problem, which can also be considered as an integrity learning based approach.

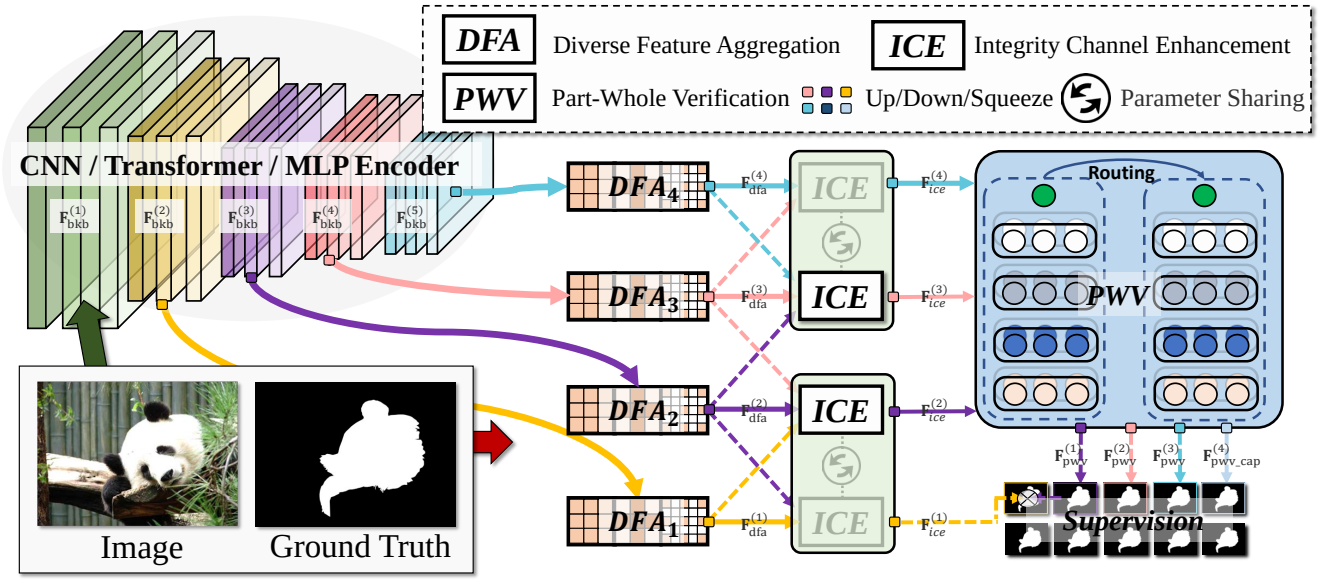


Fig. 3. Overall architecture of the proposed ICON. Feature extraction: $\mathbf{F}_{bkb}^{(1)}$ - $\mathbf{F}_{bkb}^{(5)}$ denote different layers from ResNet-50 [57]. Component 1: DFA aggregates the features with various receptive fields. Component 2: ICE aims at enhancing the feature channels that highlight the potential integral salient object. Component 3: PWV judges whether the part and whole features have strong agreement.

3 Framework

3.1 Overview of ICON

As shown in Fig. 3, our method is based on an encoder-decoder architecture. The encoder uses ResNet-50 as the backbone to extract multi-level features. Meanwhile, the decoder integrates these multi-level features and generates the saliency map with multi-layer supervision. For simplicity, from here on we denote the features generated by the backbone as a set $\mathcal{F}_{bkb} = \{\mathbf{F}_{bkb}^{(0)}, \mathbf{F}_{bkb}^{(1)}, \mathbf{F}_{bkb}^{(2)}, \mathbf{F}_{bkb}^{(3)}, \mathbf{F}_{bkb}^{(4)}\}$. To improve the computational efficiency, we do not use $\mathbf{F}_{bkb}^{(0)}$ in the decoder due to its large spatial size.

Next, we enhance the backbone features by passing them through the various feature aggregation (DFA) module, which consists of various convolutional blocks. Thereafter, we further use the integrity channel enhancement (ICE) module to strengthen the responses of the integrity-related channels and coarsely highlight the integral salient parts. Finally, we utilize the part-whole verification (PWV) module to verify the agreement between object parts and the whole salient region, to further refine the saliency map.

3.2 Diverse Feature Aggregation

Recent works [63]–[65] have demonstrated that enriching the receptive fields of the convolution kernel can help the network learn features that capture different object sizes. In this work, we go one step further and incorporate convolution kernels with different shapes to deal with the shape diversity of different objects. Specifically, we adopt the novel DFA module to enhance the diversity of the extracted multi-level features, using three kinds of convolutional blocks with different kernel sizes and shapes, as shown in Fig. 4-(A). Technically, we utilize a practical combination of the asymmetric convolution [66], atrous convolution [67], and original convolution to capture diverse spatial features. The overall procedure is summarized as follows:

$$\mathbf{F}_{dfa}^{(i)} = \text{Concat}\left[\mathcal{X}_{asy}(\mathbf{F}_{bkb}^{(i)}), \mathcal{X}_{atr}^{2}(\mathbf{F}_{bkb}^{(i)}), \mathcal{X}_{ori}(\mathbf{F}_{bkb}^{(i)})\right], \tag{1}$$

where $\mathbf{F}_{dfa}^{(i)}$ denotes the features produced by the above process, $\mathcal{X}_{*}$ denotes different types of blocks (*i.e.*, asymmetric, atrous, original), and $\text{Concat}[\cdot]$ is the concatenation operation.

Note that we use $\mathcal{X}_{atr}^{r}$ to denote atrous convolution operations with different dilation rates r, *e.g.*, $\mathcal{X}_{atr}^{2}$ is the atrous convolution with dilation rate 2, and use the asymmetric convolution ($\mathcal{X}_{asy}$) with a crux-shape [66]. Specially, $\mathcal{X}_{asy}$ contains three layers, one with a normal 3×3 square kernel $\mathbf{K}_{3\times3}$, one with a horizontal 1×3 kernel $\mathbf{K}_{1\times3}$, and one with a vertical 3×1 kernel $\mathbf{K}_{3\times1}$, shared in the same sliding window. It can be described as:

$$\mathcal{X}_{asy}(\mathbf{I}) = (\mathbf{I} \star \mathbf{K}_{3\times3}) \oplus (\mathbf{I} \star \mathbf{K}_{1\times3}) \oplus (\mathbf{I} \star \mathbf{K}_{3\times1}), \tag{2}$$

where $\star$ is the 2D convolutional operator, $\oplus$ is the element-wise addition, and $\mathbf{I}$ denotes the input feature.

In such a way, our DFA module can enrich the feature space by fusing the learned knowledge from the crux kernel, dilated kernel, and normal kernel in the first stage. As a result, DFA can cover different salient regions in various contexts, enhancing integrity. We mark the features processed by DFA as $\mathcal{F}_{dfa} = \{\mathbf{F}_{dfa}^{(1)}, \mathbf{F}_{dfa}^{(2)}, \mathbf{F}_{dfa}^{(3)}, \mathbf{F}_{dfa}^{(4)}\}$.

3.3 Integrity Channel Enhancement

Several recent studies [68]–[71] have achieved promising visual categorization results by using the spatial or channel attention mechanism. Though these methods are driven by various motivations, they all essentially aim to build the correspondence between different features to highlight

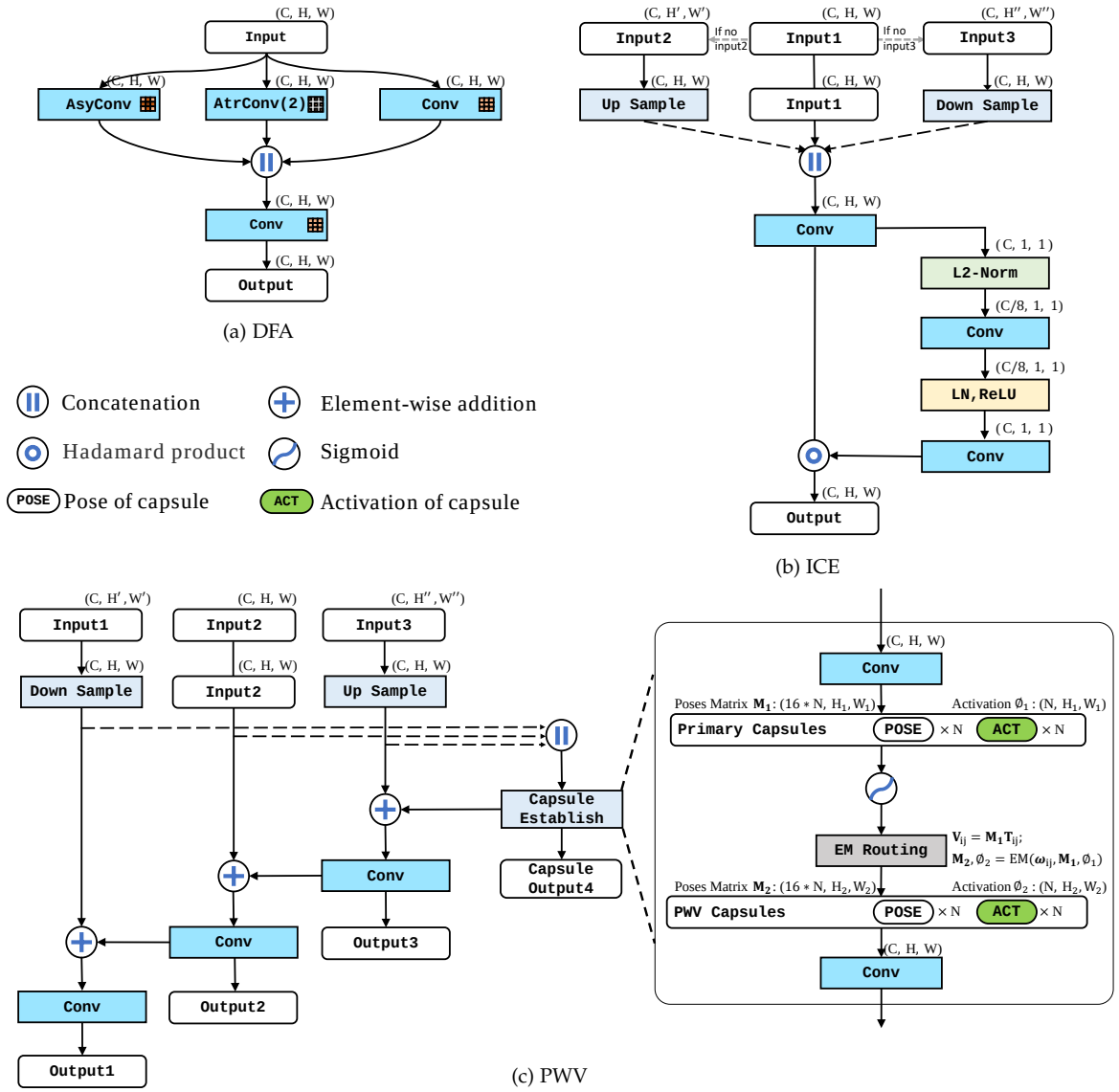


Fig. 4. Details of the proposed modules. (a) The diverse feature aggregation (DFA) component combines different convolutional kernels to enhance the representational ability of the framework. (b) The integrity channel enhancement (ICE) component, mines integrity information in channels. (c) The part-whole verification (PWV) component is designed for modeling the relation between parts and whole object. AsyConv = *asymmetric block* [66], AtrConv = *atrous block* [67], Conv = conv block. All these blocks include convolution, BatchNorm and ReLU components. EM Routing = the expectation-maximum routing mechanism [33]. C, H, W denote the channel number, height, and width of the feature tensor, respectively.

the most significant object parts. However, how to mine the integrity information hidden in different channel of features reminds under studied. To address this, we propose a simple ICE module to further mine the relations within different channels, and enhance the channels that highlight the potential integral targets.

We consider multi-scale information from every three adjacent features. First, we re-scale the next and previous feature levels and use upsampling and downsampling operations to adjust them to a spatial resolution of $H \times W$. Then, we generate the fusion maps $\mathbf{F}_{fuse}^{(i)}$ by concatenating the three input features:

$$\mathbf{F}_{fuse}^{(i)} = \text{Concat}\left[\mathbf{F}_{dfa}^{(i-1)}, \mathbf{F}_{dfa}^{(i)}, \mathbf{F}_{dfa}^{(i+1)}\right]. \tag{3}$$

After that, we extract the integrity embedding $\mathbf{I}_{emb}^{(i)}$ by applying the l_2 norm on $\mathbf{F}_{fuse}^{(i)}$. Next, to further integrate the integrity information, we use a parameter-efficient bottleneck design to learn $\mathbf{I}_{emb}$. As the channel transform would slightly increase the difficulty of optimization, we add layer normalization inside two convolution layers (before ReLU) to ease optimization, as [70] does in their design:

$$\mathbf{F}_{ice}^{(i)} = \mathbf{F}_{fuse}^{(i)} \otimes \mathcal{X}_{ori}(\text{ReLU}(\text{LN}(\mathcal{X}_{ori}(\mathbf{I}_{emb}^{(i)})))), \tag{4}$$

where $\otimes$ is the element-wise multiplication operation and LN means layer normalization.

By using the proposed ICE module, the channels with better integrity can be effectively enhanced. As can be seen in Fig. 5, after feeding the features into our ICE, the foreground region is noticeably distinguished from the background, and the features produced by ICE tend to

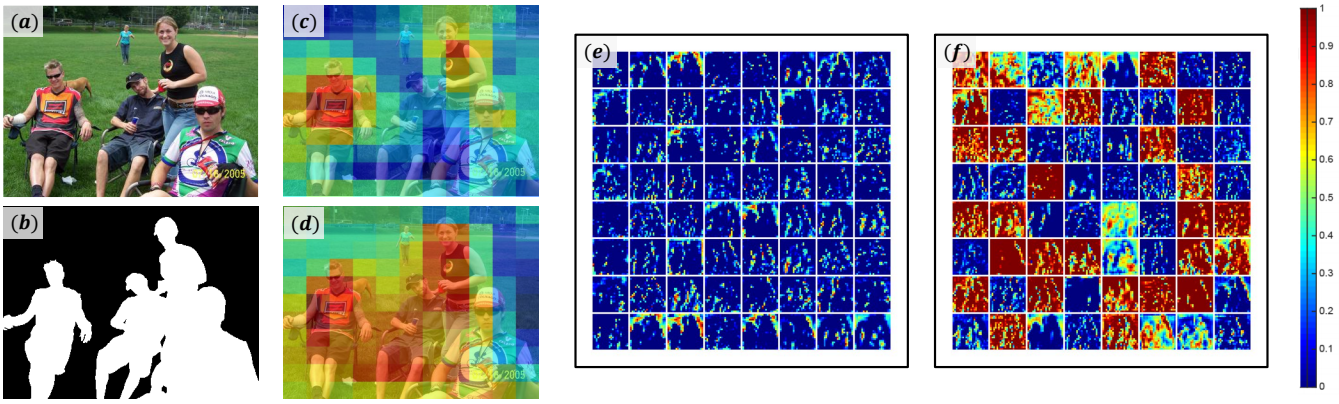


Fig. 5. Visual comparison of heatmaps through ICE. (a) Input image. (b) Ground truth. Feature heatmaps (c) before and (d) after our ICE module. Channel visualizations (e) before and (f) after ICE module. Our ICE module helps the network focus more on the integral salient regions and distinguish the foreground from the background. The feature heatmap presented in (d) clearly demonstrates the ability to capture the integrity representation at both macro- and micro levels. The channel visualizations are generated by squeezing all channels and we use Matplotlib's pseudo-color 'jet' for colorization. Zoom in for better viewing.

highlight the integral objects at both the micro and macro levels. In our implementation, if there are not enough multi-level features input in the first and last levels, we fill the input with the features from the current level. Besides, we use two ICE modules with shared parameters to help our ICON model integrate cues at multiple levels.

3.4 Part-Whole Verification

The PWV module aims to enhance the learned integrity features by measuring the agreement between object parts and the whole salient region. To achieve this goal, we adopt a capsule network [33], [72], which has been proved effective in modeling part-whole relationships. Motivated by the success of the prior work SegCaps [73], we embed the capsule network into ICON. In PWV, one key issue is how to assign votes from the low-level to the high-level capsules. As the high-level capsules need to form the whole object representation by aggregating the object parts from the relevant low-level capsules, we use EM routing [33] to model the association between the low-level and high-level capsules in a clustering-like manner. The inputs of PWV are three different ICE features ($\mathcal{F}_{ice}$). Specifically, we first reduce the ICE features at each level to a united resolution, *i.e.*, 22×22, in order to reduce the computational costs.

Next, we build our primary capsules. To be specific, we use eight pose vectors to build a pose matrix $\mathbf{M}$, and an activation $\phi \in [0, 1]$ to represent each capsule. The pose matrix contains the instantiated parameters to reflect the properties of object parts or the whole object, while the activation represents the existence probability of the object. Capsules from the primary capsule layer pass information to those in the next PWV capsule layer through a routing-by-agreement mechanism. Specifically, when the capsules from a lower layer produce votes for the capsules in a higher level, the votes ω_{ij} are obtained by a matrix multiplication operation between the learned transformation matrices $\mathbf{T}_{ij}$ and the lower-level pose matrix $\mathbf{M}_i$, where i and j are the indices of the lower- and higher level capsules, respectively. Once these votes are obtained, they are used in the EM routing algorithm [33] to get the higher-level capsule $\mathbf{C}_j$ with the pose matrices $\mathbf{M}_j$ and activation ϕ_j. After that, we obtain the part-whole verified features. Subsequently, element-wise addition and upsampling operations are introduced to fuse these part-whole verified features at adjacent levels in a bottom-up manner, which encourages cooperation among multi-scale features. After PWV module, the model generate $\mathcal{F}_{pwv} = \{\mathbf{F}^{(1)}_{pwv}, \mathbf{F}^{(2)}_{pwv}, \mathbf{F}^{(3)}_{pwv}, \mathbf{F}^{(4)}_{pwv\_cap}\}$.

3.5 Supervision Strategy

In this work, in addition to the BCE loss, we also use the IoU loss [52], [74]. Specifically, the overall loss of the proposed ICON is formulated as $\mathcal{L}_{CPR}(P, G)$, where P is the generated saliency prediction map, and G is the ground truth saliency map. $\mathcal{L}_{CPR}$ incorporates the cooperative BCE loss and IoU loss, *i.e.*, $\mathcal{L}_{CPR} = \mathcal{L}_{BCE} + \mathcal{L}_{IoU}$. Specifically, $\mathcal{L}_{BCE}$ is formulated as follows:

$$\begin{aligned}\mathcal{L}_{BCE} = -\sum_{x=1}^{H}\sum_{y=1}^{W}[&G(x,y)\log(P(x,y)) \\ &+ (1-G(x,y))\log(1-P(x,y))],\end{aligned} \tag{5}$$

where W and H are the width and height of the images, respectively. Meanwhile, L_{IoU} is defined as:

$$\mathcal{L}_{IoU} = 1 - \frac{\sum_{x=1}^{H}\sum_{y=1}^{W} P(x,y)G(x,y)}{\sum_{x=1}^{H}\sum_{y=1}^{W}[P(x,y) + G(x,y) - P(x,y)G(x,y)]}, \tag{6}$$

where $G(x, y)$ and $P(x, y)$ are the ground truth label and predicted saliency label of the location (x, y), respectively. During training, we use multi-level supervision strategy that widely used in this field [28], [38], [54], [60]. Apart from using four features from $\mathcal{F}_{pwv}$, we fuse $\mathbf{F}^{(1)}_{pwv}$ and $\mathbf{F}^{(1)}_{ice}$ by dot-product as an extra feature for supervision, and this feature is also used to generate final predictions during the inference period. To match the ground-truth maps in both training and inference periods, features' channel will be reduced to 1-dimension, and the spatial size will be recovered as the same as the input image.

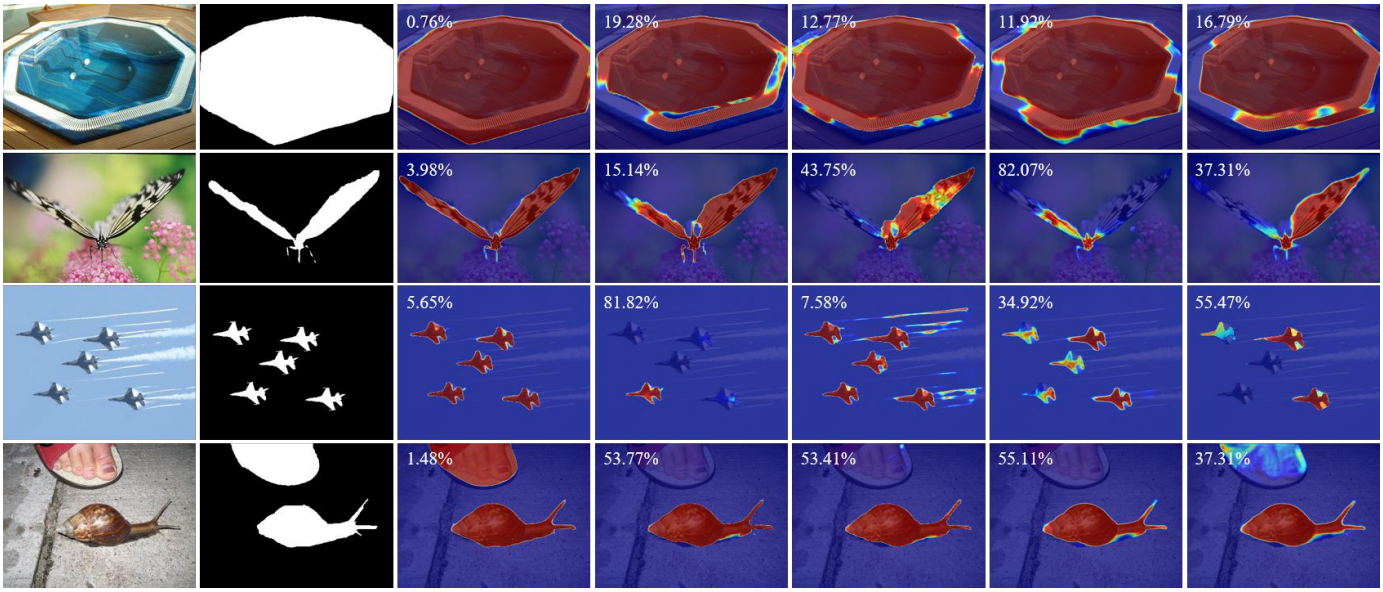


Image GT Ours-R LDF [29] MINet-R [38] GateNet-R [27] EGNet-R [51]

Fig. 6. Visual comparisons at different false negative ratios (FNR). Note that the FNR heavily reflects the *integrity*.

4 Experiments

4.1 Datasets

We train our ICON on the **DUTS-TR** [75] dataset, which is commonly used for the SOD task and contains 10,553 images. Then, we evaluate all the models on seven popular benchmarks: **ECSSD** [76], **HKU-IS** [77], **OMRON** [15], **PASCAL-S** [78], **DUTS-TE** [75], **SOD** [79] and attribute-based **SOC** [18], which are all annotated with pixel-level labels. Specifically, ECSSD is made up of 1,000 images with meaningful semantics. HKU-IS includes 4,447 images, containing multiple foreground objects. OMRON consists of 5,168 images with at least one object. These objects are usually structurally complex. PASCAL-S was built from a dataset originally used for semantic segmentation, and it consists of 850 challenging images. DUTS is a relatively large dataset with two subsets. The 10,553 images in DUTS-TR are used for training, and the 5,019 images in DUTS-TE are employed for testing. SOD includes 300 very challenging images. SOC contains complicated scenes, which are more challenging than those in the other six SOD datasets.

4.2 Implementation Details

We run all experiments on the publicly available Pytorch 1.5.0 platform. An eight-core PC with an Intel Core i7-9700K CPU (with 4.9GHz Turbo boost), 16GB 3000 MHz RAM and an RTX 2080Ti GPU card (with 11GB memory) is used for both training and testing. During network training, each image is first resized to 352×352 (for the VGG [80]/ResNet [57]/PVT [81], [82] backbones) or 384×384 (for Swin [83]/CycleMLP [84]), and data augmentation methods such as normalizing, cropping and flipping, are used. Some encoder parameters are initialized from VGG-16, ResNet-50, PVTv2, Swin-B and CycleMLP-B4. We initialize some layers of PWV by zeros or ones, while other convolutional layers are initialized following [85]. We use the SGD optimizer [86] to train our network, setting its hyperparameters as: initial learning rate lr = 0.05, momen = 0.9, eps = 1e-8, weight_decay = 5e-4. Warm-up and linear decay strategies are used to adjust the learning rate. The batch size is set to 32 (ResNet), 10 (PVTv2/CycleMLP) or 8 (VGG/Swin), and the maximum number of epochs is set to 60 (ResNet-based training takes ∼2.5 hours). In addition, we use apex[1] and fp16 to accelerate the training process. Gradient clipping is also used to prevent gradient explosion. The inference process of the ResNet-based architecture for a 352×352 image only takes 0.0164s, including the IO time.

4.3 Evaluation Metrics

We use five metrics to evaluate our model and existing state-of-the-art algorithms:

(1) MAE (M) evaluates the average pixel-wise difference between the predicted saliency map (P) and the ground truth map (G). We normalize P and G to $[0, 1]$, so the MAE score can be computed as $M = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |P(x, y) - G(x, y)|$.

(2) Weighted F-measure (F_β^ω) [87] offers an intuitive generalization of F_β, and is defined as $F_\beta^\omega = \frac{(1+\beta^2)\text{Precision}^\omega \cdot \text{Recall}^\omega}{\beta^2 \cdot \text{Precision}^\omega + \text{Recall}^\omega}$. As a widely adopted metric [18], [22], [45], [49], [53], [59], [88]–[91], F_β^ω can handle the interpolation, dependency and equal-importance issues which might cause inaccurate evaluation by MAE and F-measure [92]. We set β^2 to 0.3 to emphasize the precision over recall, as suggested in [1]. By assigning different weights (ω) to different errors following the specific location and neighborhood information, F_β^ω extends the F-measure to non-binary evaluation.

(3) S-measure (S_m) [93] focuses on evaluating the structural similarity, which is much closer to human visual perception. It is computed as $S_m = m\text{s}_o + (1 - m)\text{s}_r$, where s_o and s_r denote the object-aware and region-aware structural similarity and m is set to 0.5, following [93].

(4) E-measure (E_ξ) [94] combines the local pixel values with the image-level mean value in one term and can

1. https://github.com/NVIDIA/apex

TABLE 1
Quantitative results on six datasets. The best performances are shown in **bold**. The symbols "↑"/"↓" mean that a higher/lower score is better. '-V/VGG-Based': VGG16 [80], '-R/ResNet-Based': ResNet50 [57], '-P': PVTv2-1K [82], '-S': Swin-B-22k [83], '-M': CycleMLP-B4 [84].

Summary			**ECSSD** [76]				**PASCAL-S** [78]				**DUTS** [75]				**HKU-IS** [77]				**OMRON** [15]				**SOD** [79]			
Method	MACs	Params	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
VGG16-Based Methods																										
RAS	-	-	.893	.914	.857	.056	.799	.835	.731	.101	.839	.871	.740	.059	.887	.920	.843	.045	.814	.843	.695	.062	.767	.791	.718	.123
CPD	59.46	29.23	.910	.938	.895	.040	.845	.882	.796	.072	.867	.902	.800	.043	.904	.940	.879	.033	.818	.845	.715	.057	.771	.787	.718	113
EGNet	149.89	108.07	**.919**	.936	.892	.041	.848	.877	.788	.077	.878	.898	.797	.044	.910	.938	.875	.035	**.836**	.853	.728	**.057**	.788	.803	.736	.110
ITSD	14.61	17.08	.914	.937	.897	.040	.856	.891	.811	.068	.877	.905	.814	.042	.906	.938	.881	.035	.829	.853	.734	.063	.797	.826	.764	.098
MINet	94.11	47.56	**.919**	.943	**.905**	**.036**	.854	.893	.808	**.064**	.875	.907	.813	**.039**	.912	.944	.889	**.031**	.822	.846	.718	**.057**	-	-	-	-
GateNet	108.34	100.02	.917	.932	.886	.041	.857	.886	.797	.068	.870	.893	.786	.045	.910	.934	.872	.036	.821	.840	.703	.061	-	-	-	-
Ours-V	64.90	19.17	**.919**	**.946**	**.905**	**.036**	**.861**	**.902**	**.820**	**.064**	**.878**	**.915**	**.822**	.043	**.915**	**.950**	**.895**	.032	.833	**.865**	**.743**	.065	**.814**	**.848**	**.784**	**.089**
ResNet50-Based Methods																										
CondInst	-	-	.721	.717	.603	.115	.813	.852	.757	.084	.760	.764	.631	.070	.748	.754	.648	.093	.646	.629	.433	.114	.670	.657	.535	.148
PointRend	-	-	.753	.766	.667	.111	.810	.850	.763	.099	.774	.794	.667	.081	.784	.805	.715	.091	.651	.647	.453	.125	.693	.793	.589	.141
PiCANet	54.05	47.22	.917	.925	.867	.046	.854	.870	.772	.076	.869	.878	.754	.051	.904	.916	.840	.043	.832	.836	.695	.065	.793	.799	.722	.103
AFNet	21.66	35.95	.913	.935	.886	.042	.849	.883	.797	.070	.867	.893	.785	.046	.905	.935	.869	.036	.826	.846	.717	.057	-	-	-	-
BASNet	127.36	87.06	.916	.943	.904	.037	.838	.879	.793	.076	.866	.895	.803	.040	.909	.943	.889	.032	.836	.865	.751	.056	.772	.801	.728	.112
CPD	17.77	47.85	.918	.942	.898	.037	.848	.882	.794	.071	.869	.898	.795	.043	.905	.938	.875	.034	.825	.847	.719	.056	.771	.782	.713	.110
EGNet	157.21	111.69	.925	.943	.903	.037	.852	.881	.795	.074	.887	.907	.815	.039	.918	.944	.887	.031	.841	.857	.738	**.053**	.807	.822	.767	.097
SCRN	15.09	25.23	.927	.939	.900	.037	**.869**	.892	.807	.063	.885	.900	.803	.040	.916	.935	.876	.034	.837	.848	.720	.056	-	-	-	-
F3Net	16.43	25.54	.924	.948	.912	.033	.861	.898	.816	**.061**	**.888**	.920	.835	**.035**	.917	.952	.900	**.028**	.838	.864	.747	**.053**	.806	.834	.775	.091
ITSD	15.96	26.47	.925	.947	.910	.034	.859	.894	.812	.066	.885	.913	.823	.041	.917	.947	.894	.031	.840	.865	.750	.061	.809	.836	.777	.093
MINET	87.11	126.38	.925	.950	.911	.033	.856	.896	.809	.064	.884	.917	.825	.037	.919	.952	.897	.029	.833	.860	.738	.056	-	-	-	-
GateNet	162.13	128.63	.920	.936	.894	.040	.858	.886	.797	.067	.885	.906	.809	.040	.915	.937	.880	.033	.838	.855	.729	.055	-	-	-	-
Ours-R	20.91	33.09	**.929**	**.954**	**.918**	**.032**	.861	**.899**	**.818**	.064	**.888**	**.924**	**.836**	.037	**.920**	**.953**	**.902**	.029	**.844**	**.876**	**.761**	.057	**.824**	**.854**	**.794**	**.084**
Transformer-Based Methods																										
VST	23.16	44.63	.932	.951	.910	.033	.872	.902	.816	.061	.896	.919	.828	.037	.928	.952	.897	.029	.850	.871	.755	.058	.820	.846	.778	.086
Ours-P	34.70	65.68	.940	.964	.933	.024	.882	.921	.847	.051	**.917**	.950	.882	**.022**	**.935**	.967	**.925**	**.022**	.865	.896	.793	.047	**.832**	**.864**	**.813**	**.078**
Ours-S	52.59	94.30	**.941**	**.966**	**.936**	**.023**	**.885**	**.924**	**.854**	**.048**	**.917**	**.954**	**.886**	.025	**.935**	**.968**	**.925**	**.022**	**.869**	**.900**	**.804**	**.043**	.825	.856	.802	.083
MLP-Based Methods																										
Ours-M	26.13	54.92	.940	.964	.934	.025	.873	.912	.838	.056	.909	.942	.874	.029	.935	.966	.926	.022	.855	.886	.783	.051	.821	.853	.803	.081

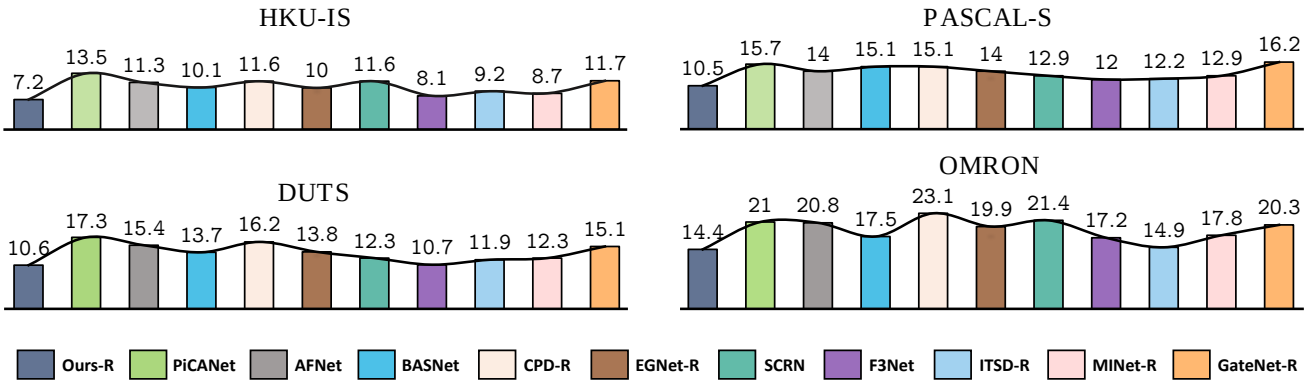


Fig. 7. FNR statistics across 11 methods on different datasets.

be computed as: $E_\xi = \frac{1}{W\times H}\sum_{x=1}^{W}\sum_{y=1}^{H}\theta(\xi)$, where ξ is the alignment matrix and $\theta(\xi)$ indicates the enhanced alignment matrix. We adopt mean E-measure (E_ξ^m) as our final evaluation.

(5) FNR is the false negative ratio. We further evaluate the integrity using this metric, which can detect whether the predictions are integral with salient pixels. The FNR is computed by:

$$FN(x,y) = \begin{cases} 1, G(x,y) = 1 \ \& \ P(x,y) = 0, \\ 0, \ others. \end{cases} \tag{7}$$

$$FNR = \frac{\sum_{x=1}^{W}\sum_{y=1}^{H} FN(x,y)}{\sum_{x=1}^{W}\sum_{y=1}^{H} G(x,y)} \times 100\%, \tag{8}$$

where FN is the pixel-level indicator that determines whether a pixel is a false negative. We show several examples of FNR in Fig. 6. It clearly and accurately reflects the *integrity* of predictions and is sensitive at the macro and micro level.

4.4 Comparison with the SOTAs

We compare the proposed approach with 14 very recent state-of-the-art methods, including Condinst [95], PointRend [96], PiCANet [46], RAS [97], AFNet [98], BASNet [52], CPD [58], EGNet [51], SCRN [55], F3Net [54], MINet [38], ITSD [99], GateNet [39] and VST [56].

4.4.1 *Quantitative Evaluation*

Table 1 reports the quantitative results on six traditional benchmark datasets, comparing with the 14 state-of-the-art algorithms in terms of S-measure, E-measure, weighted F-measure, and MAE. Our model is clearly superior to the other alternatives. Besides, we also show the FNR results of ours and the baseline methods in Fig. 7. As can be seen, our approach achieves the lowest FNR scores across all datasets. Visual comparisons (see Fig. 6) also demonstrate its efficiency in capturing integral objects. In fact, ICON performs favorably against the existing methods across all datasets

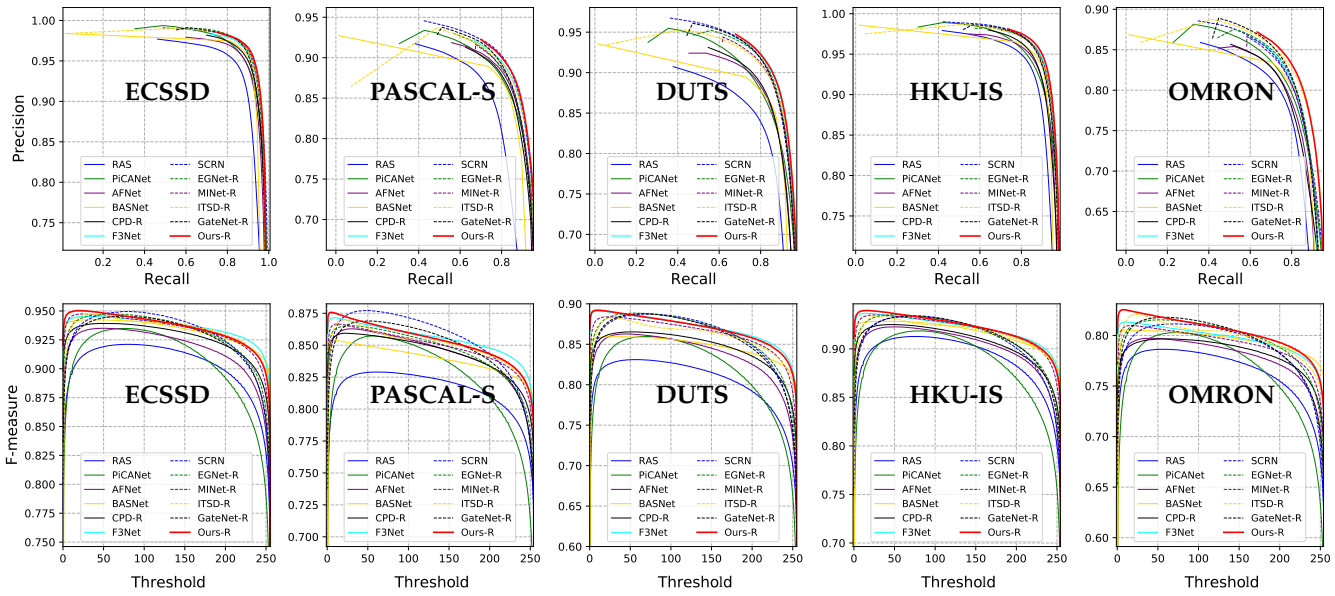


Fig. 8. Precision-recall and F-measure curves of the proposed method and other SOTA algorithms on five popular SOD datasets.

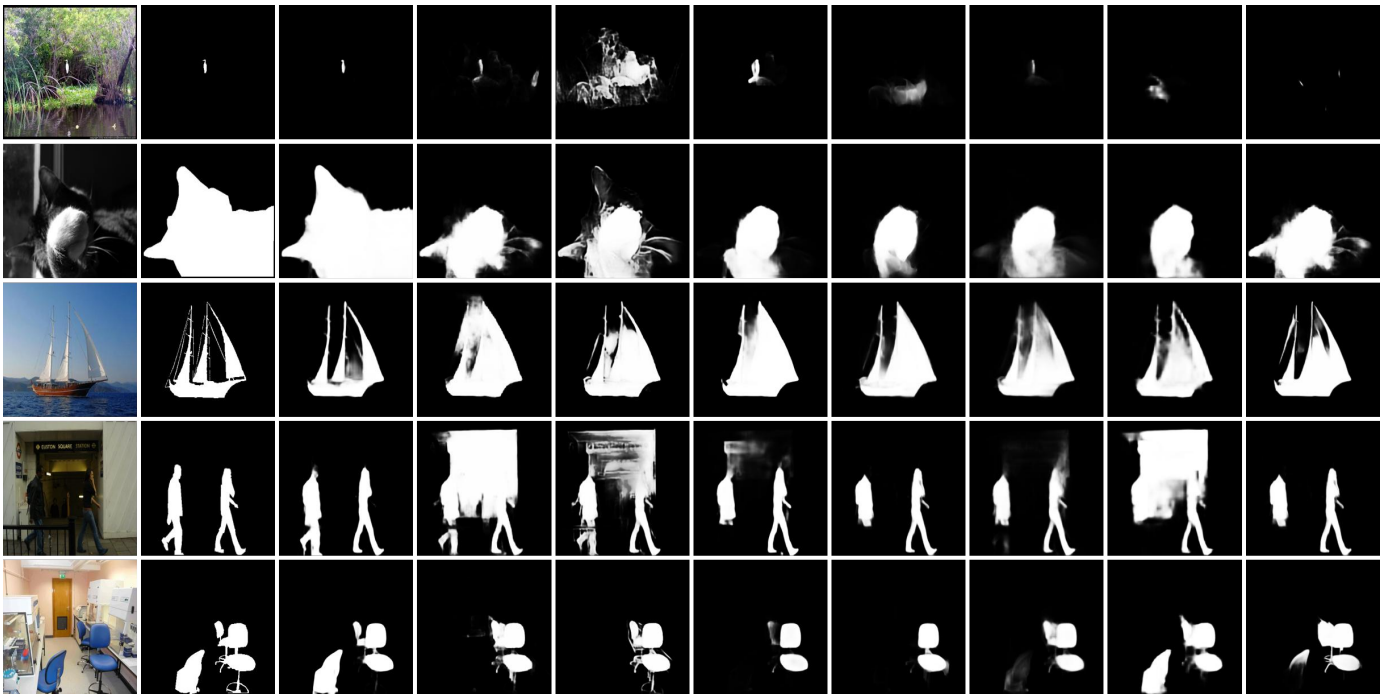


Fig. 9. Qualitative comparison of our model with seven SOTA methods. Unlike other models, our method not only accurately locates the salient object but also produces sharper edges with fewer background distractors for various scenes.

and in terms of nearly all evaluation metrics. This demonstrates its strong capability in dealing with challenging inputs. In addition, we present the precision-recall [13] and F-measure curves [92] in Fig. 8. The solid red lines belonging to the proposed method are obviously higher than the other curves, which further demonstrates the effectiveness of the proposed model and the integrity learning.

4.4.2 Visual Comparisons

Fig. 9 provides visual comparisons between our approach and the baseline methods. As can be observed, ICON generates more accurate saliency maps for various challenging cases, *e.g.*, small objects (1^{st}), large objects (2^{nd} row), delicate structures (3^{rd} row), low-contrast (4^{th} row), and multiple objects (5^{th} row). Besides, our framework can detect salient targets integrally and noiselessly. The above results demonstrate the accuracy and robustness of the proposed method.

4.4.3 Attribute-Based Analysis

In addition to the most frequently used saliency detection datasets, we also test our model on another challenging SOC dataset [18], [105]. Compared with the previous six

TABLE 2
Comparison of the proposed method with other SOTA models on the SOC test set. For ↑ and ↓, higher and lower scores indicate better results, respectively. † indicates that the parameters are trained on the SOC training set.

Attr	Metrics	Amulet [22]	DSS [35]	NLDF [48]	C2SNet [49]	SRM [100]	R3Net [101]	BMPM [102]	DGRL [103]	PiCA-R [46]	RANet [104]	AFNet [98]	CPD [58]	PoolNet [28]	EGNet [51]	BANet [50]	SCRN [55]	ICON-R (Ours)
AC	$S_m \uparrow$	0.752	0.753	0.737	0.755	0.791	0.713	0.780	0.790	0.792	0.708	0.796	0.799	0.795	0.806	0.806	0.809	**0.835**/**0.835**†
	$E_\xi^m \uparrow$	0.791	0.788	0.784	0.807	0.824	0.753	0.815	0.853	0.815	0.765	0.852	0.843	0.846	0.854	0.858	0.849	**0.891**/0.889†
	$F_\beta^w \uparrow$	0.620	0.629	0.620	0.647	0.690	0.593	0.680	0.718	0.682	0.603	0.712	0.727	0.713	0.731	0.740	0.724	**0.784**/**0.784**†
	$M \downarrow$	0.120	0.113	0.119	0.109	0.096	0.135	0.098	0.081	0.093	0.132	0.084	0.083	0.094	0.085	0.086	0.078	**0.062**/0.064†
BO	$S_m \uparrow$	0.574	0.561	0.568	0.654	0.614	0.437	0.604	0.684	**0.729**	0.421	0.658	0.647	0.561	0.528	0.645	0.698	0.714/0.713†
	$E_\xi^m \uparrow$	0.551	0.537	0.539	0.661	0.616	0.419	0.620	0.725	0.741	0.404	0.698	0.665	0.554	0.528	0.650	0.706	0.740/**0.743**†
	$F_\beta^w \uparrow$	0.612	0.614	0.622	0.730	0.667	0.456	0.670	0.786	**0.799**	0.453	0.741	0.739	0.610	0.585	0.720	0.778	0.794/0.794†
	$M \downarrow$	0.346	0.356	0.354	0.267	0.306	0.445	0.303	0.215	0.200	0.454	0.245	0.257	0.353	0.373	0.271	0.224	0.200/**0.199**†
CL	$S_m \uparrow$	0.763	0.722	0.713	0.742	0.759	0.659	0.761	0.770	0.787	0.624	0.768	0.773	0.760	0.757	0.784	0.795	0.789/**0.802**†
	$E_\xi^m \uparrow$	0.789	0.763	0.764	0.789	0.793	0.710	0.801	0.824	0.794	0.715	0.802	0.821	0.801	0.790	0.824	0.820	0.829/**0.855**†
	$F_\beta^w \uparrow$	0.663	0.617	0.614	0.655	0.665	0.546	0.678	0.714	0.692	0.542	0.696	0.724	0.681	0.677	0.726	0.717	0.732/**0.754**†
	$M \downarrow$	0.141	0.153	0.159	0.144	0.134	0.182	0.123	0.119	0.123	0.188	0.119	0.114	0.134	0.139	0.117	0.113	0.113/**0.102**†
HO	$S_m \uparrow$	0.791	0.767	0.755	0.768	0.794	0.740	0.781	0.791	0.809	0.713	0.798	0.803	0.815	0.802	0.819	0.823	0.818/**0.830**†
	$E_\xi^m \uparrow$	0.810	0.796	0.798	0.805	0.819	0.782	0.813	0.833	0.819	0.777	0.834	0.845	0.846	0.829	0.850	0.842	0.852/**0.865**†
	$F_\beta^w \uparrow$	0.688	0.660	0.661	0.668	0.696	0.633	0.684	0.722	0.704	0.626	0.722	0.751	0.739	0.720	0.754	0.743	0.752/**0.771**†
	$M \downarrow$	0.119	0.124	0.126	0.123	0.115	0.136	0.116	0.104	0.108	0.143	0.103	0.097	0.100	0.106	0.094	0.096	0.092/**0.087**†
MB	$S_m \uparrow$	0.712	0.719	0.685	0.720	0.742	0.657	0.762	0.744	0.775	0.696	0.734	0.754	0.751	0.762	0.764	0.792	0.774/**0.821**†
	$E_\xi^m \uparrow$	0.739	0.753	0.740	0.778	0.778	0.697	0.812	0.823	0.813	0.761	0.762	0.804	0.779	0.789	0.803	0.817	0.828/**0.866**†
	$F_\beta^w \uparrow$	0.561	0.577	0.551	0.593	0.619	0.489	0.651	0.655	0.637	0.576	0.626	0.679	0.642	0.649	0.672	0.690	0.699/**0.768**†
	$M \downarrow$	0.142	0.132	0.138	0.128	0.115	0.160	0.105	0.113	0.099	0.139	0.111	0.106	0.121	0.109	0.104	0.100	0.100/**0.076**†
OC	$S_m \uparrow$	0.735	0.718	0.709	0.738	0.749	0.653	0.752	0.747	0.765	0.641	0.771	0.750	0.756	0.754	0.765	0.775	0.771/**0.791**†
	$E_\xi^m \uparrow$	0.763	0.760	0.755	0.784	0.780	0.706	0.800	0.808	0.784	0.718	0.820	0.810	0.801	0.798	0.809	0.800	0.817/**0.831**†
	$F_\beta^w \uparrow$	0.607	0.595	0.593	0.622	0.630	0.520	0.644	0.659	0.638	0.527	0.680	0.672	0.659	0.658	0.678	0.673	0.683/**0.710**†
	$M \downarrow$	0.143	0.144	0.149	0.130	0.129	0.168	0.119	0.116	0.119	0.169	0.109	0.115	0.119	0.121	0.112	0.111	0.106/**0.100**†
OV	$S_m \uparrow$	0.721	0.700	0.688	0.728	0.745	0.624	0.751	0.762	0.781	0.611	0.761	0.748	0.747	0.752	0.779	0.774	0.779/**0.802**†
	$E_\xi^m \uparrow$	0.751	0.737	0.736	0.790	0.779	0.663	0.807	0.828	0.810	0.664	0.817	0.803	0.795	0.802	0.835	0.808	0.834/**0.846**†
	$F_\beta^w \uparrow$	0.637	0.622	0.616	0.671	0.682	0.527	0.701	0.733	0.721	0.529	0.723	0.721	0.697	0.707	0.752	0.723	0.749/**0.768**†
	$M \downarrow$	0.173	0.180	0.184	0.159	0.150	0.216	0.136	0.125	0.127	0.217	0.129	0.134	0.148	0.146	0.119	0.126	0.120/**0.108**†
SC	$S_m \uparrow$	0.768	0.761	0.745	0.756	0.783	0.716	0.799	0.772	0.784	0.724	0.808	0.793	0.807	0.793	0.807	0.809	0.803/**0.824**†
	$E_\xi^m \uparrow$	0.794	0.799	0.788	0.806	0.814	0.765	0.841	0.837	0.799	0.792	0.854	0.858	0.856	0.844	0.851	0.843	0.860/**0.882**†
	$F_\beta^w \uparrow$	0.608	0.599	0.593	0.611	0.638	0.550	0.677	0.669	0.627	0.594	0.696	0.708	0.695	0.678	0.706	0.691	0.714/**0.745**†
	$M \downarrow$	0.098	0.098	0.101	0.100	0.090	0.114	0.081	0.087	0.093	0.110	0.076	0.080	0.075	0.083	0.078	0.078	0.080/**0.073**†
SO	$S_m \uparrow$	0.718	0.713	0.703	0.706	0.737	0.682	0.732	0.736	0.748	0.682	0.746	0.745	0.768	0.749	0.755	0.767	0.763/**0.801**†
	$E_\xi^m \uparrow$	0.745	0.756	0.747	0.752	0.769	0.732	0.780	0.802	0.766	0.759	0.792	0.804	0.814	0.784	0.801	0.797	0.816/**0.848**†
	$F_\beta^w \uparrow$	0.523	0.524	0.526	0.531	0.561	0.487	0.567	0.602	0.566	0.518	0.596	0.623	0.626	0.594	0.621	0.614	0.634/**0.689**†
	$M \downarrow$	0.119	0.109	0.115	0.116	0.099	0.118	0.096	0.092	0.095	0.113	0.089	0.091	0.087	0.098	0.090	0.082	0.087/**0.073**†

SOD datasets, this dataset contains many more complicated scenes. In addition, the SOC dataset categorizes images according to nine different attributes, including AC (appearance change), BO (big object), CL (clutter), HO (heterogeneous object), MB (motion blur), OC (occlusion), OV (out-of-view), SC (shape complexity), and SO (small object).

In Table 2, we present comparisons between our ICON and 16 state-of-the-art models, including Amulet [22], DSS [35], NLDF [48], C2SNet [49], SRM [100], R3Net [101], BMPM [102], DGRL [103], PiCANet-R (PiCA-R) [25], RANet [104], AFNet [98], CPD [58], PoolNet [28], EGNet [51], BANet [50] and SCRN [55] in terms of attribute-based performance. As seen, our ICON achieves clear performance improvement against the existing methods.

4.5 Failure Cases

Although the proposed ICON method outperforms other SOD algorithms and rarely generates completely incorrect prediction results, there are still some failure cases, as shown in Fig. 10. Specifically, in the first row, which shows a tidy room, our method is confused by whether the pillow or the bed and wall is the salient object. Meanwhile, in the second image, the three lamp lights are the salient regions, but our method cannot detect them. Similarly, other SOTA methods also fail for these samples. We believe there are several reasons for these failure cases: (1) strong color contrast influencing the model's judgment (*e.g.*, 1^{st} row); (2) lack of sufficient training samples (see the 2^{nd} row) and (3) controversial annotations (*i.e.*, 1^{st} row).

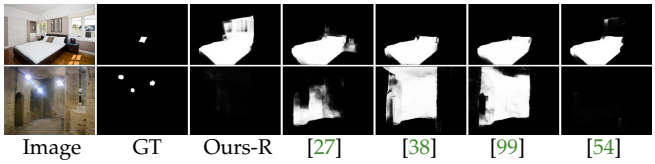


Fig. 10. Failure cases. The first and second columns are the input images, and ground-truth masks. The others are prediction results of ICON and our competitors.

TABLE 3
Ablation analysis of gradually including the proposed components. The best performances are shown in **bold**.

ID	Component Settings	OMRON [15]				HKU-IS [77]				DUTS-TE [75]			
		$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
1	Baseline	0.832	0.854	0.731	0.064	0.902	0.930	0.866	0.043	0.861	0.879	0.801	0.049
2	+DFA	0.837	0.857	0.740	0.063	0.913	0.939	0.875	0.035	0.879	0.886	0.818	0.046
3	+DFA+ICE	0.840	0.869	0.753	0.059	0.918	0.951	0.895	0.031	0.887	0.916	0.825	0.038
4	+DFA+ICE+PWV	**0.844**	**0.876**	**0.761**	**0.057**	**0.920**	**0.953**	**0.902**	**0.029**	**0.888**	**0.924**	**0.836**	**0.037**

TABLE 4
Ablation analysis of different feature enhancement methods (FEMs). The best performances are shown in **bold**.

ID	FEMs Settings	OMRON [15]				HKU-IS [77]				DUTS-TE [75]			
		$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
2	+DFA	0.837	**0.857**	**0.740**	0.063	**0.913**	0.939	**0.875**	0.035	**0.879**	**0.886**	**0.818**	0.046
5	+Inception [64]	0.839	0.853	**0.740**	0.064	0.909	0.936	0.869	0.037	0.875	**0.886**	0.817	**0.043**
6	+ASPP [67]	**0.840**	0.855	0.738	**0.061**	0.912	**0.943**	0.869	**0.034**	0.877	0.882	**0.818**	**0.043**
7	+PSP [65]	0.835	0.851	0.738	0.063	0.906	0.935	0.870	0.036	0.878	0.884	0.816	0.045
8	+DFA (3xOriConv)	0.833	0.855	0.733	0.062	0.909	0.938	0.868	0.035	0.874	0.875	0.810	0.046
9	+DFA (3xAtrConv [67])	0.830	0.849	0.729	0.066	0.906	0.933	0.869	0.038	0.872	0.880	0.811	0.047
10	+DFA (3xAsyConv [66])	0.837	0.854	0.737	0.064	0.909	0.937	0.873	0.035	**0.879**	0.885	0.817	0.044

TABLE 5
Ablation analysis of ICE and related attention mechanisms.

ID	Attention Settings	OMRON [15]				HKU-IS [77]				DUTS-TE [75]			
		$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
3	+DFA+ICE	0.840	**0.869**	**0.753**	0.059	**0.918**	**0.951**	**0.895**	**0.031**	0.887	**0.916**	0.825	**0.038**
11	+DFA+SE [68]	0.839	0.861	0.720	0.061	0.909	0.943	0.877	0.034	**0.888**	0.895	0.831	0.039
12	+DFA+CBAM [69]	**0.842**	0.864	0.739	**0.058**	0.917	0.946	0.891	**0.031**	0.885	0.907	**0.832**	0.039
13	+DFA+GCT [71]	0.838	0.857	0.712	0.062	0.901	0.937	0.874	0.033	0.883	0.905	0.821	0.041

4.6 Ablation Study

4.6.1 Effectiveness of Different Components

To demonstrate the effectiveness of different components in our ICON, we report the quantitative results of several simplified versions of our model. We start from the encoder-decoder baseline (a UNet-like network with skip connections) and progressively extend it with different modules, including DFA, ICE, and PWV. As shown in Table 3, we first test the Baseline (ID: 1) and DFA (ID: 2) elements, which demonstrate an obvious performance promotion. This is reasonable because DFA has the ability to search for objects with diverse cues. Then we add the ICE module (ID: 3), which again shows a substantial improvement. Finally, as anticipated, adding all components (ID: 4) to the proposed model achieves the best performance.

4.6.2 DFA vs. Other Feature Enhancement Methods

DFA, ASPP [67], Inception [64], and PSP [65] are four feature enhancement methods (FEMs), which share some common ideas to stimulate representative feature learning. Differently, our DFA is designed to enhance feature sub-spaces without enlarging the receptive field, which yields more diverse representations. In Table 4, DFA clearly outperforms or is on par with other FEMs, with fewer[2] convolutional blocks. However, DFA also brings some drawbacks. For instance, it generates higher MAE scores compared with other FEMs. We argue that one possible reason is that DFA not only brings feature diversity but also some noise. Besides, our experiments (ID: 2 vs. ID: 8∼10) reveal that combining three different types of convolutions can achieve the best score. Meanwhile, using only 3xAsyConv yields better results than only using 3xOriConv or 3xAtrConv.

2. DFA has only 4 blocks, while other FEMs have at least 5 blocks.

4.7 ICE vs. Attention Methods

In Table 5, we make an additional control group (*i.e.*, ID: 3, 11∼13) to verify the improvement brought by the ICE mechanism. Following the same setting (ID: 3), we conduct experiments to compare ICE with SE [68], CBAM [69] and GCT [71]. We observe that CBAM achieves an acceptable performance and ranks second among these modules. However, the alternative methods using SE and GCT would lead to a noticeable drop in performance. One possible explanation is ICE can strengthen the *integrity* of features and highlight potential salient candidates through our designed attention mechanisms.

4.7.1 Evaluation of Different Routing Algorithms

To evaluate the performance of EM routing [33] (ID: 4), we also conduct additional experiments (see Table 6) replacing it with dynamic routing (DR) [72] and self-routing [106]. We observe that former (ID: 14) also achieves reasonable performance, but the latter (ID: 15) yields worse performance, compared to using EM routing. One possible reason is that SR does not have the routing-by-agreement mechanism, making it incompatible with our PWV scheme.

4.7.2 Evaluation of Loss Function

To demonstrate the effectiveness of the L_{CPR} loss, we conduct another experiment comparing it to L_{BCE} in our

TABLE 6
Ablation analysis of routing mechanism in PWV.

ID	Routing Settings	OMRON [15]				HKU-IS [77]				DUTS-TE [75]			
		$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
4	+DFA+ICE+PWV	**0.844**	**0.876**	**0.761**	**0.057**	0.920	**0.953**	**0.902**	**0.029**	**0.888**	**0.924**	**0.836**	**0.037**
14	+DFA+ICE+PWV (DR) [72]	**0.844**	0.868	0.757	0.058	**0.923**	0.950	**0.902**	0.030	**0.888**	0.919	0.832	0.039
15	+DFA+ICE+PWV (SR) [106]	0.837	0.862	0.745	0.060	0.912	0.843	0.895	0.030	0.881	0.903	0.831	0.042

TABLE 7
Ablation analysis of loss function.

ID	Loss Settings	OMRON [15]				HKU-IS [77]				DUTS-TE [75]			
		$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$	$S_m\uparrow$	$E_\xi^m\uparrow$	$F_\beta^w\uparrow$	$M\downarrow$
4	ICON+L_{CPR}	**0.844**	**0.876**	**0.761**	**0.057**	**0.920**	**0.953**	**0.902**	**0.029**	0.888	**0.924**	**0.836**	**0.037**
16	ICON+L_{BCE}	0.840	0.866	0.757	0.060	0.918	0.950	0.899	0.031	**0.889**	0.918	0.831	**0.037**

ICON architecture. The results reported in Table 7 indicate that, after using the L_{CPR} loss in the training process, our model can significantly improve the SOD performance across all metrics. Note that combining the IoU and BCE loss is a common training setting, which has also been used in many recent works [52], [62].

5 Conclusion

We present a novel **I**ntegrity **Co**gnition **N**etwork, called **ICON**, to detect salient objects from given image scenes. It is based on the observation that mining integral features (at both a micro and macro level) can substantially benefit the salient object detection process. Specifically, in this work, three novel network modules are designed: the diverse feature aggregation module, the integrity channel enhancement module, and the part-whole verification module. By integrating these modules, ICON is able to capture diverse features at each feature level and enhance feature channels that highlight the potential integral salient objects, as well as further verify the part-whole agreement between the mined salient object regions. Comprehensive experiments on seven benchmark datasets are conducted. The experimental results demonstrate the contribution of each newly proposed component, as well as the superior performance of our ICON.

Acknowledgments

The authors would like to thank the anonymous reviewers and editor for their helpful comments on this manuscript. And we thank Jing Zhang for sharing codes of their work.

References

[1] A. Borji, M.-M. Cheng, H. Jiang, and J. Li, "Salient object detection: A benchmark," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5706–5722, 2015.
[2] D.-P. Fan, Z. Lin, Z. Zhang, M. Zhu, and M.-M. Cheng, "Rethinking rgb-d salient object detection: Models, data sets, and large-scale benchmarks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 2075–2089, 2021.
[3] W. Wang, Q. Lai, H. Fu, J. Shen, and H. Ling, "Salient object detection in the deep learning era: An in-depth survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021.
[4] D. Zhang, J. Han, L. Zhao, and D. Meng, "Leveraging prior-knowledge for weakly supervised object detection under a collaborative self-paced curriculum learning framework," *Int. J. Comput. Vis.*, vol. 127, no. 4, pp. 363–380, 2019.
[5] G. Liu and D. Fan, "A model of visual attention for natural image retrieval," in *Int. Conf. Inf. Sci. Cloud Comput. Companion*, 2013, pp. 728–733.
[6] 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 Trans. Pattern Anal. Mach. Intell.*, 2022.
[7] M. Zhuge, D. Gao, D.-P. Fan, L. Jin, B. Chen, H. Zhou, M. Qiu, and L. Shao, "Kaleido-bert: Vision-language pre-training on fashion domain," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021, pp. 12 647–12 657.
[8] X. Qin, Z. Zhang, C. Huang, M. Dehghan, O. R. Zaiane, and M. Jagersand, "U2-net: Going deeper with nested u-structure for salient object detection," *Pattern Recognition*, vol. 106, p. 107404, 2020.
[9] L. Hoyer, M. Munoz, P. Katiyar, A. Khoreva, and V. Fischer, "Grid saliency for context explanations of semantic segmentation," in *Adv. Neural Inform. Process. Syst.*, 2019, pp. 6462–6473.
[10] Y. Wei, X. Liang, Y. Chen, X. Shen, M.-M. Cheng, J. Feng, Y. Zhao, and S. Yan, "Stc: A simple to complex framework for weakly-supervised semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2314–2320, 2016.
[11] Y. Zeng, Y. Zhuge, H. Lu, and L. Zhang, "Joint learning of saliency detection and weakly supervised semantic segmentation," in *IEEE Int. Conf. Comput. Vis.*, 2019, pp. 7223–7233.
[12] W. Wang, J. Shen, R. Yang, and F. Porikli, "Saliency-aware video object segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 1, pp. 20–33, 2017.
[13] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, 2014.
[14] 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, 1998.
[15] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 3166–3173.
[16] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 2814–2821.
[17] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient object detection: A discriminative regional feature integration approach," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 2083–2090.
[18] D.-P. Fan, M.-M. Cheng, J.-J. Liu, S.-H. Gao, Q. Hou, and A. Borji, "Salient objects in clutter: Bringing salient object detection to the foreground," in *Eur. Conf. Comput. Vis.*, 2018, pp. 186–202.
[19] A. Borji, M.-M. Cheng, Q. Hou, H. Jiang, and J. Li, "Salient object detection: A survey," *Comput. Vis. Media*, pp. 1–34, 2014.
[20] J. Han, D. Zhang, G. Cheng, N. Liu, and D. Xu, "Advanced deep-learning techniques for salient and category-specific object detection: a survey," *IEEE Signal Process. Mag.*, vol. 35, no. 1, pp. 84–100, 2018.
[21] B. Cheng, R. Girshick, P. Dollár, A. C. Berg, and A. Kirillov, "Boundary iou: Improving object-centric image segmentation evaluation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021, pp. 15 334–15 342.

[22] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *IEEE Int. Conf. Comput. Vis.*, 2017, pp. 202–211.

[23] Z. Luo, A. Mishra, A. Achkar, J. Eichel, S. Li, and P. Jodoin, "Non-local deep features for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 6593–6601.

[24] T. Zhao and X. Wu, "Pyramid feature attention network for saliency detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 3085–3094.

[25] N. Liu, J. Han, and M.-H. Yang, "Picanet: Pixel-wise contextual attention learning for accurate saliency detection," *IEEE Trans. Image Process.*, vol. 29, pp. 6438–6451, 2020.

[26] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 5968–5977.

[27] X. Zhao, Y. Pang, L. Zhang, H. Lu, and L. Zhang, "Suppress and balance: A simple gated network for salient object detection," in *Eur. Conf. Comput. Vis.*, 2020, pp. 35–51.

[28] 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 Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 3917–3926.

[29] J. Wei, S. Wang, Z. Wu, C. Su, Q. Huang, and Q. Tian, "Label decoupling framework for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020, pp. 13 025–13 034.

[30] R. Wu, M. Feng, W. Guan, D. Wang, H. Lu, and E. Ding, "A mutual learning method for salient object detection with intertwined multi-supervision," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 8150–8159.

[31] M. Amirul Islam, M. Kalash, and N. D. Bruce, "Revisiting salient object detection: Simultaneous detection, ranking, and subitizing of multiple salient objects," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 7142–7150.

[32] S. He, J. Jiao, X. Zhang, G. Han, and R. W. H. Lau, "Delving into salient object subitizing and detection," in *IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1059–1067.

[33] G. E. Hinton, S. Sabour, and N. Frosst, "Matrix capsules with em routing," in *Int. Conf. Learn. Represent.*, 2018.

[34] S. Xie and Z. Tu, "Holistically-nested edge detection," *IJCV*, vol. 125, no. 1-3, pp. 3–18, 2017.

[35] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. H. Torr, "Deeply supervised salient object detection with short connections," *PAMI*, vol. 41, no. 4, pp. 815–828, 2019.

[36] X. Hu, L. Zhu, J. Qin, C.-W. Fu, and P.-A. Heng, "Recurrently aggregating deep features for salient object detection," in *AAAI Conf. Art. Intell.*, 2018, pp. 6943–6950.

[37] K. Zhao, S. Gao, W. Wang, and M.-M. Cheng, "Optimizing the f-measure for threshold-free salient object detection," in *IEEE Int. Conf. Comput. Vis.*, 2019, pp. 8849–8857.

[38] Y. Pang, X. Zhao, L. Zhang, and H. Lu, "Multi-scale interactive network for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020, pp. 9413–9422.

[39] X. Zhao, Y. Pang, L. Zhang, H. Lu, and L. Zhang, "Suppress and balance: A simple gated network for salient object detection," in *Eur. Conf. Comput. Vis.*, 2020, pp. 35–51.

[40] J.-J. Liu, Z.-A. Liu, P. Peng, and M.-M. Cheng, "Rethinking the u-shape structure for salient object detection," *TIP*, vol. 30, pp. 9030–9042, 2021.

[41] Y.-F. Ma and H.-J. Zhang, "Contrast-based image attention analysis by using fuzzy growing," in *ACM Int. Conf. Multimedia*, 2003, pp. 374–381.

[42] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Adv. Neural Inform. Process. Syst.*, 2006.

[43] P. Hu, B. Shuai, J. Liu, and G. Wang, "Deep level sets for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 2300–2309.

[44] S. He, R. W. Lau, W. Liu, Z. Huang, and Q. Yang, "Supercnn: A superpixelwise convolutional neural network for salient object detection," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 330–344, 2015.

[45] G. Li and Y. Yu, "Deep contrast learning for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 478–487.

[46] N. Liu, J. Han, and M.-H. Yang, "PiCANet: Learning pixel-wise contextual attention for saliency detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 3089–3098.

[47] J. D. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *ICML*, 2001.

[48] Z. Luo, A. Mishra, A. Achkar, J. Eichel, S. Li, and P.-M. Jodoin, "Non-local deep features for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 6593–6601.

[49] X. Li, F. Yang, H. Cheng, W. Liu, and D. Shen, "Contour knowledge transfer for salient object detection," in *Eur. Conf. Comput. Vis.*, 2018, pp. 355–370.

[50] J. Su, J. Li, Y. Zhang, C. Xia, and Y. Tian, "Selectivity or invariance: Boundary-aware salient object detection," in *IEEE Int. Conf. Comput. Vis.*, 2019, pp. 3799–3808.

[51] J.-X. Zhao, J.-J. Liu, D.-P. Fan, Y. Cao, J. Yang, and M.-M. Cheng, "EGNet: Edge guidance network for salient object detection," in *IEEE Int. Conf. Comput. Vis.*, 2019, pp. 8779–8788.

[52] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "BASNet: Boundary-aware salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 7479–7489.

[53] Y. Zeng, P. Zhang, J. Zhang, Z. Lin, and H. Lu, "Towards high-resolution salient object detection," in *IEEE Int. Conf. Comput. Vis.*, 2019, pp. 7234–7243.

[54] J. Wei, S. Wang, and Q. Huang, "F^3net: Fusion, feedback and focus for salient object detection," in *AAAI Conf. Art. Intell.*, vol. 34, no. 07, 2020, pp. 12 321–12 328.

[55] Z. Wu, L. Su, and Q. Huang, "Stacked Cross Refinement Network for Edge-Aware Salient Object Detection," in *IEEE Int. Conf. Comput. Vis.*, 2019, pp. 7264–7273.

[56] N. Liu, N. Zhang, K. Wan, L. Shao, and J. Han, "Visual saliency transformer," in *IEEE Int. Conf. Comput. Vis.*, 2021.

[57] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 770–778.

[58] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 3907–3916.

[59] Y. Liu, Q. Zhang, D. Zhang, and J. Han, "Employing deep part-object relationships for salient object detection," in *IEEE Int. Conf. Comput. Vis.*, 2019, pp. 1232–1241.

[60] Zuyao Chen and Qianqian Xu and Runmin Cong and Qingming Huang, "Global Context-Aware Progressive Aggregation Network for Salient Object Detection," in *AAAI Conf. Art. Intell.*, vol. 34, no. 07, 2020, pp. 10 599–10 606.

[61] Z. Wu, S. Li, C. Chen, A. Hao, and H. Qin, "A deeper look at image salient object detection: Bi-stream network with a small training dataset," *IEEE Trans. Multimedia*, 2020.

[62] Y. Mao, J. Zhang, Z. Wan, Y. Dai, A. Li, Y. Lv, X. Tian, D.-P. Fan, and N. Barnes, "Generative transformer for accurate and reliable salient object detection," *arXiv preprint arXiv:2104.10127*, 2021.

[63] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[64] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 2818–2826.

[65] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 2881–2890.

[66] X. Ding, Y. Guo, G. Ding, and J. Han, "ACNet: Strengthening the kernel skeletons for powerful cnn via asymmetric convolution blocks," in *IEEE Int. Conf. Comput. Vis.*, 2019, pp. 1911–1920.

[67] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, 2017.

[68] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *PAMI*, vol. 42, no. 08, pp. 2011–2023, 2020.

[69] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "Cbam: Convolutional block attention module," in *Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[70] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "Gcnet: Non-local networks meet squeeze-excitation networks and beyond," in *IEEE Int. Conf. Comput. Vis. Worksh.*, 2019, pp. 0–0.

[71] Z. Yang, L. Zhu, Y. Wu, and Y. Yang, "Gated channel transformation for visual recognition," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020, pp. 11 794–11 803.

[72] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Adv. Neural Inform. Process. Syst.*, 2017, pp. 3856–3866.

[73] R. LaLonde and U. Bagci, "Capsules for object segmentation," in *International conference on Medical Imaging with Deep Learning*, 2018.

[74] G. Máttyus, W. Luo, and R. Urtasun, "Deeproadmapper: Extracting road topology from aerial images," in *IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3438–3446.

[75] 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 Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 136–145.

[76] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 1155–1162.

[77] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 5455–5463.

[78] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 280–287.

[79] V. Movahedi and J. H. Elder, "Design and perceptual validation of performance measures for salient object segmentation," in *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2010, pp. 49–56.

[80] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Int. Conf. Learn. Represent.*, 2015.

[81] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *IEEE Int. Conf. Comput. Vis.*, 2021.

[82] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pvtv2: Improved baselines with pyramid vision transformer," *Comput. Vis. Media*, vol. 8, no. 3, pp. 1–10, 2022.

[83] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *IEEE Int. Conf. Comput. Vis.*, 2021.

[84] S. Chen, E. Xie, C. Ge, D. Liang, and P. Luo, "Cyclemlp: A mlp-like architecture for dense prediction," in *Int. Conf. Learn. Represent.*, 2022.

[85] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1026–1034.

[86] L. Bottou, "Stochastic gradient descent tricks," in *Neural networks: Tricks of the trade*, 2012, pp. 421–436.

[87] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps?" in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 248–255.

[88] L. Zhang, J. Wu, T. Wang, A. Borji, G. Wei, and H. Lu, "A multistage refinement network for salient object detection," *TIP*, vol. 29, pp. 3534–3545, 2020.

[89] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Salient object detection with recurrent fully convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1734–1746, 2018.

[90] Y. Piao, W. Ji, J. Li, M. Zhang, and H. Lu, "Depth-Induced Multi-Scale Recurrent Attention Network for Saliency Detection," in *IEEE Int. Conf. Comput. Vis.*, 2019, pp. 7254–7263.

[91] G. Li, Y. Xie, L. Lin, and Y. Yu, "Instance-level salient object segmentation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 2386–2395.

[92] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 1597–1604.

[93] M.-M. Cheng and D.-P. Fan, "Structure-measure: A new way to evaluate foreground maps," *IJCV*, vol. 129, no. 9, pp. 2622–2638, 2021.

[94] D.-P. Fan, G.-P. Ji, X. Qin, and M.-M. Cheng, "Cognitive vision inspired object segmentation metric and loss function," *SCIENTIA SINICA Informationis*, 2021.

[95] Z. Tian, C. Shen, and H. Chen, "Conditional convolutions for instance segmentation," in *ECCV*. Springer, 2020, pp. 282–298.

[96] A. Kirillov, Y. Wu, K. He, and R. Girshick, "Pointrend: Image segmentation as rendering," in *CVPR*, 2020, pp. 9799–9808.

[97] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *Eur. Conf. Comput. Vis.*, 2018, pp. 234–250.

[98] M. Feng, H. Lu, and E. Ding, "Attentive feedback network for boundary-aware salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 1623–1632.

[99] H. Zhou, X. Xie, J.-H. Lai, Z. Chen, and L. Yang, "Interactive two-stream decoder for accurate and fast saliency detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020, pp. 9141–9150.

[100] T. Wang, A. Borji, L. Zhang, P. Zhang, and H. Lu, "A stagewise refinement model for detecting salient objects in images," in *IEEE Int. Conf. Comput. Vis.*, 2017, pp. 4039–4048.

[101] Z. Deng, X. Hu, L. Zhu, X. Xu, J. Qin, G. Han, and P.-A. Heng, "R3Net: Recurrent residual refinement network for saliency detection," in *Int. Joint Conf. Artif. Intell.*, 2018, pp. 684–690.

[102] L. Zhang, J. Dai, H. Lu, Y. He, and G. Wang, "A bi-directional message passing model for salient object detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 1741–1750.

[103] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 3127–3135.

[104] S. Chen, X. Tan, B. Wang, H. Lu, X. Hu, and Y. Fu, "Reverse attention-based residual network for salient object detection," *IEEE Trans. Image Process.*, vol. 29, pp. 3763–3776, 2020.

[105] D.-P. Fan, J. Zhang, G. Xu, M.-M. Cheng, and L. Shao, "Salient objects in clutter," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2022.

[106] T. Hahn, M. Pyeon, and G. Kim, "Self-routing capsule networks," in *Adv. Neural Inform. Process. Syst.*, 2019, pp. 7658–7667.

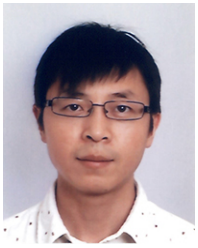
Mingchen Zhuge is a research intern in Inception Institute of Artificial Intelligence (IIAI). He received the M.S. degree in Computer Science from China University of Geosciences in 2021. In 2019, he won the champion of intelligent scheduling group in the 2019 ZTE algorithm competition. And in 2020, he worked as an algorithm intern on Alibaba Group. His main research interests include computer vision, multimodal AI.

Deng-Ping Fan received his PhD degree from the Nankai University in 2019. He joined Inception Institute of Artificial Intelligence (IIAI) in 2019. He has published about 25 top journal and conference papers such as TPAMI, IJCV, CVPR, *etc.* His research interests include computer vision and visual attention, especially on RGB salient object detection (SOD), RGB-D SOD, Video SOD, Co-SOD. He won the Best Paper Finalist Award at IEEE CVPR 2019, the Best Paper Award Nominee at IEEE CVPR 2020.

Nian Liu is a researcher with Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE. He received the Ph.D. degree and the B.S. degree from School of Automation at Northwestern Polytechnical University, in 2020 and 2012, respectively. His research interests include computer vision and machine learning, especially on saliency detection and deep learning.

Dingwen Zhang received the PhD degree from Northwestern Polytechnical University, China, in 2018. He is a professor with the School of Automation, Northwestern Polytechnical University, China. From 2015 to 2017, he was a visiting scholar at the Robotic Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania. His research interests include computer vision and multimedia processing, especially on saliency detection and weakly supervised learning.

Dong Xu is Chair in Computer Engineering at the School of Electrical and Information Engineering, The University of Sydney, Australia. He received the B.Eng. and PhD degrees from University of Science and Technology of China, in 2001 and 2005, respectively. While pursuing the PhD degree, he worked at Microsoft Research Asia and The Chinese University of Hong Kong for more than two years. He also worked as a postdoctoral research scientist at Columbia University from 2006 to 2007 and a faculty member at Nanyang Technological University from 2007 to 2015. His current research interests include computer vision, multimedia, machine learning and biomedical image analysis. He has published more than 100 papers in IEEE Transactions and top tier conferences including CVPR, ICCV, ECCV, ICML, ACM MM and MICCAI. He is on the editorial boards of IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Neural Networks and Learning Systems and IEEE Transactions on Circuits and Systems for Video Technology.



Ling Shao is the Chief Scientist of Terminus Group and the President of Terminus International. He was the founding CEO and Chief Scientist of the Inception Institute of Artificial Intelligence, Abu Dhabi, UAE. His research interests include computer vision, deep learning, medical imaging and vision and language. He is a fellow of the IEEE, the IAPR, the BCS and the IET.