Dense Attentive Feature Enhancement for Salient Object Detection

Zun Li, Congyan Lang, Liqian Liang, Jian Zhao, Songhe Feng, Qibin Hou, and Jiashi Feng

***Abstract*—Attention mechanisms have been proven highly effective for salient object detection. Most previous works utilize attention as a self-gated module to reweigh the feature maps at different levels *independently*. However, they are limited to certain-level guidance and could not satisfy the need of both accurately detecting intact objects and maintaining their detailed boundaries. In this paper, we build dense attention upon features from multiple levels *simultaneously* and propose a novel Dense Attentive Feature Enhancement (DAFE) module for efficient feature enhancement in saliency detection. DAFE stacks several attentional units and densely connects attentive feature output from current unit to its all subsequent units. This allows feature maps at deep units to absorb attentive information from shallow units, thus more discriminative information can be efficiently selected at the final output. Note that DAFE is plug and play, which can be effortlessly inserted into any saliency or video saliency models for their performance improvements. We further instantiate a highly effective Dense Attentive Feature Enhancement Network (DAFE-Net) for accurate salient object detection. DAFE-Net constructs DAFE over the aggregation feature that contains both semantics and saliency details, the entire salient objects and their boundaries can be well retained through dense attentions. Extensive experiments demonstrate that the proposed DAFE module is highly effective, and the DAFE-Net performs favorably compared with state-of-the-art approaches.**

***Index Terms*—Salient Object Detection, Dense Attention, Dense Attentive Feature Enhancement.**

I. Introduction

Salient object detection (SOD) aims to identify the most visually conspicuous objects in the given image. It is an important pre-processing step for various computer vision applications, such as image or video segmentation [1][2][3], object retrieval [4], human parsing [5], photo cropping [6], image captioning [7][8], video compression [9][10] and visual tracking [11][12]. Early saliency methods [13][14][15][16] generally rely on hand-crafted visual features and heuristic clues, which have limited capacity of modeling and describing high-level semantics. Recently, the fully convolutional networks (FCNs) [17] based approaches [18][19][20][21][22][23] have greatly promoted the progresses of SOD because of their capability of extracting both high-level semantic information and low-level details. Despite being studied actively, how to select feasible and effective features remains an important problem in deep saliency detection.

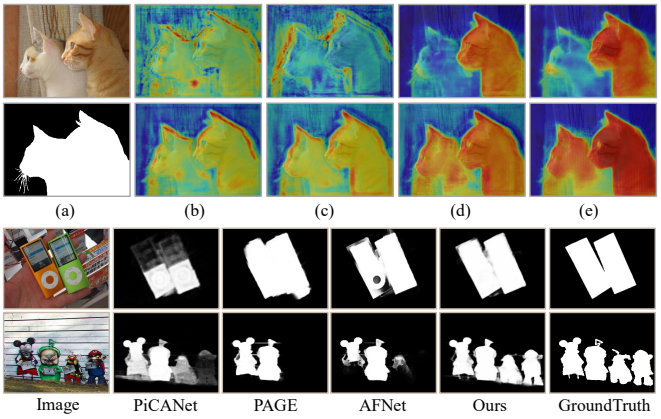


Fig. 1: Top panel: visualizations of attention-enhanced feature maps. (a) The input image and corresponding Ground Truth. (b)-(e) Feature maps after 1∼4 times of refinement. Features in the first row are produced via single-level attention in PiCANet [24], while those in the second row are generated by the proposed dense attention. Bottom panel: examples of saliency maps produced by PiCANet [24], PAGE [25], AFNet [26] and our DAFE-Net. Our model consistently performs better than other state-of-the-arts. Best viewed in color.

To deal with this issue, the recent popular attention mechanism, which posses great ability to dynamically select adaptive features, has been intensively incorporated into FCNs to pursue the state-of-the-art performance in SOD. While integrating the convolutional features, most previous methods [26][27][28][24][25] utilize attention modules as a self-gated mechanism to reweigh the feature maps at different feature levels independently, which attempts to let each convolutional level selectively focus on beneficial contextual information and neglect other saliency-irrelevant information for one salient region. Although great performance improvements can be gained, these methods select features in a certain feature level and ignore feature communications from other levels, hindering from well preserving the beneficial information from high-level and low-level features.

As shown in the top panel of Fig. 1, with independent level attention, the low-level enhanced features tend to introduce some interferences that may miss out on finer details (see *row* 1 in (b)-(c)); while the high-level ones are activated with partly salient regions, resulting the predicted saliency map with incomplete object structures (see *row* 1 in (d)-(e)). Therefore, as illustrated in Fig. 1, when facing various challenging factors presented in the input images, such as objects touching image boundaries (*row* 1), objects with similar appearance with background (*row* 3), and images with complex foreground (*row* 4), accurately locating the entire object while maintaining

Zun Li, Congyan Lang, Liqian Liang and Songhe Feng are with School of Computer and Information Technology, Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing 100044, China (E-mails: 16112072@bjtu.edu.cn, cylang@bjtu.edu.cn, shfeng@bjtu.edu.cn). Jian Zhao is with Institute of North Electronic Equipment, Beijing, China (E-mails: zhaojian90@u.nus.edu, Homepage: https://zhaoj9014.github.io/). Qibin Hou and Jiashi Feng are with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore (E-mails: andrewhoux@gmail.com, elefjia@nus.edu.sg). Congyan Lang is the corresponding author.

the original sharp object boundaries remains a great challenge for existing attention based saliency methods.

In this paper, we propose a novel Dense Attentive Feature Enhancement (DAFE) module for efficient feature selection and enhancement in SOD. This module is built with a new concept, *i.e.,* dense attention, which connects a set of attentional maps in a dense way to control the information propagation. To achieve this, DAFE consists of several couples of efficient attentional units, *i.e.*, Feature Refinement Unit (FRU) and Dense Attentional Collection Unit (DACU). The former one learns an attention map for selecting features from all of its subsequent FRU, while the later one densely correlates a set of attention maps from all its previous FRU to enhance feature from current FRU. In this way, feature maps at deep units can absorb attentive information from shallow units, see *row* 2 in Fig. 1, making the redundant information can be recurrently filtered and the discriminability of the whole module will be progressively improved. As a result, the final output feature map in DAFE contains more efficient information and pays more attention to the salient regions when applying DAFE in saliency detection. Note that DAFE is plug and play, which can be effortlessly inserted into any saliency or video saliency models to enhance their performance.

Based on DAFE, we further instantiate a novel Network, DAFE-Net, for salient object detection. It devises a DAFE module over the aggregation feature to help selecting rich feature and highlighting accurate salient regions, see Fig. 2. In DAFE-Net, we use feature aggregator to fuse features from different convolutional blocks, thus the fused feature not only convey high-level semantics but also low-level details. On top of it, we construct a DAFE module to selectively pass useful saliency information. Since DAFE-Net builds dense attention upon feature from multiple levels simultaneously, more discriminative high-level and low-level information can be progressively selected for helping retaining the entire salient objects and their boundaries in the final prediction. Some examples are visualized in the bottom panel of Fig. 1. Clearly, benefiting from the proposed module, DAFE-Net can predict more complete salient objects with more accurate boundaries. Our contributions are summarized as follows:

- To the best of our knowledge, we are the first to present the concept of dense attention to efficiently control the information propagation. This concept provides a new solution for feature selection in salient object detection.
- We propose a novel attention module, *i.e.,* DAFE, to densely connect a set of attentional units for efficient feature enhancement. Such module can be easily plugged into any advanced saliency or video saliency models for the further improvement.
- We instantiate an effective network, *i.e.,* DAFE-Net, based on the above DAFE module for accurate salient object detection. Experimental results verify the effectiveness of the proposed DAFE module and demonstrate the superiority of DAFE-Net over the state-of-the-arts.

II. Related work

A. Salient Object Detection

Various algorithms have been developed for SOD. Early ones mainly rely on the hand-crafted visual features (*e.g.*, color [29], texture [13], intensity contrast [14]) or some heuristic saliency priors, including color contrast [15], center prior [14][30] and background prior [16][13][31][32], to predict saliency maps. These methods achieve limited performance since high-level semantics of salient objects are insufficient to be represented. Recently, with the rise of deep learning techniques, deep neural network based SOD models have substantially improved. In what follows, we mainly discuss the deep learning based SOD models, and refer the readers to [33] for a detailed survey of this topic.

(1) Multi-level Feature Fusion based SOD: To greatly improve detection performance, unlike early deep saliency models [34][35][36] that are constructed at patch level, some recent studies [37][38][19][39][18][40][41][23][21][42] integrated both high-level and low-level features from multi-layers over the whole image directly. For example, Zhang *et al.* [19] proposed to simultaneously aggregate multi-layer features and predicted salient regions via a bi-directional inference. Hou *et al.* [18] introduced short connections to the Holistically-Nested Edge Detector (HED) [43], and detected salient object via integrating saliency maps from each side-output. Following this method, Zhang *et al.* [21] designed a bi-directional architecture to extract multi-level features and then combined them to predict saliency maps. More recently, Liu *et al.* [23] proposed a PoolNet via plugging topmost level information into FPN [44] fusion branch for detecting the salient objects. Wu *et al.* [22] proposed a cascaded partial decoder framework cascading high-level features to the low-level features. Liu *et al.* proposed a new guidance strategy to effectively integrate multi-level contextual information. They further utilized a group convolution module to improve the feature discriminability. Pang *et al.* [42] first integrated the similar resolution features of adjacent layers, they then embedded several self-interaction modules to extract the multi-scale information from a single layer feature for the saliency inference. Although these methods based on multi-level feature fusion are reasonable, the background noise from feature maps at different levels would interfere the final results if they are directly aggregated.

(2) Stage-wisely Feature Refinement based SOD: To cope with the above mentioned problem, some refinement strategies were proposed in recent works [45][46][47][48][49][50]. For instance, Wang *et al.* [39] adopted a pyramid pooling module to capture the global context information of salient regions, and refined the saliency maps using a multi-stage refinement mechanism. Inspired by [51], Amirul *et al.* [48] proposed an encoder-decoder network to progressively refine the saliency maps from low-resolution to high-resolution. Afterward, Xiao *et al.* [45] first integrated multi-scale features to get a coarse saliency map, then they developed a distraction detection network to refine the coarse map by diagnosing the distracted unsalient information. Wang *et al.* [46] proposed a local boundary refinement module to refine the salient regions that were learned from the global salient localization network.

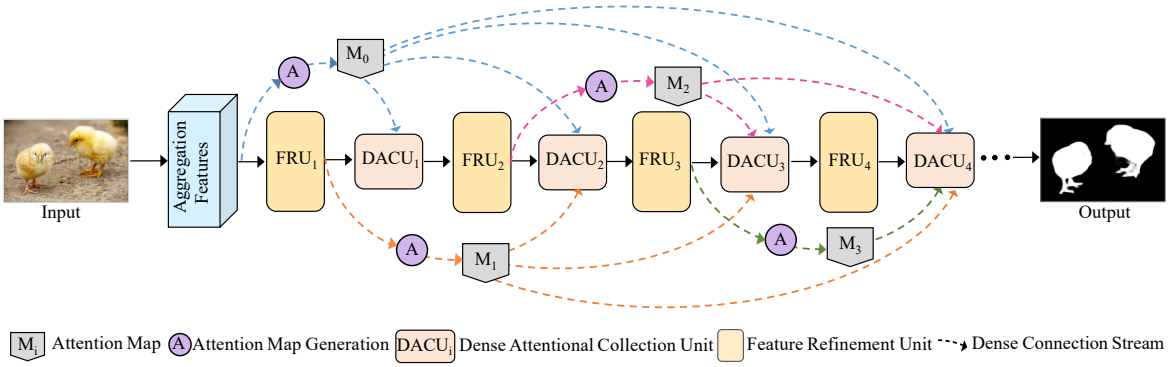


Fig. 2: The framework of DAFE-Net. It first aggregates features from different level to obtain the aggregated feature. On top of this feature, DAFE-Net constructs a DAFE module that stacking a set of feature refinement units (FRUs), correlated with a series of dense attentional collection units (DACUs). The final saliency map is produced from the last FRU. Structures of each FRU and DACU are presented in Fig. 3 and Fig. 4 in detail.

Recently, Wang *et al.* [50] integrated both top-down and bottom-up saliency inference in an iterative and cooperative manner. Although coarse saliency maps can be enhanced with these methods, how to select effective features still an important problem in salient object detection.

B. Attention Mechanism in SOD

The attention mechanism has great ability of filtering information and is widely used in various computer vision tasks, including image captioning [52], machine translation [53], object recognition [54], visual dialog [55][56], and visual question answering [57]. Recently, some saliency models [58][24][48][26][27][59][28][60][61]
citeASIF-Netalso employed attention networks for performance promotion. In particular, Zhang *et al.* [58] introduced an attention guided network that progressively integrated multiple layer-wise attentional features for saliency detection. Liu *et al.* [24] utilized a contextual attention network to predict local and global attention maps, they then incorporated multi-scale attentional features to detect salient object. Islam *et al.* [48] took gate units between each encoder and decoder blocks as attention models. These gate units control the feed-forward message passing for the sake of filtering out ambiguous information. More recently, Zhao *et al.* [27] selected low-level and high-level aggregation features separately via a channel-wise attention model and a spatial attention model. Chen [28] proposed a reverse attention to guide each side-output residual learning in a top-down manner for saliency estimation. Li [62] introduced an attention mechanism to locate the potential salient regions of cross-modal and cross-level complementarity in an attention-weighted fashion. In [60], by simulating how humans process a scene sequentially, Wang *et al.* utilized the fixation prior as a selective mechanism and attentively inferred salient object by considering coarser-to-finer features in a top-down manner. Quite different from these methods that utilize attention to reweigh features at different levels independently, we build dense attention upon aggregated features from all layers and propose to enhance them via densely incorporating a set of attentional units. In this way, feature maps at deep units can absorb attentive information from shallow units, which would progressively optimize the salient object contents and their detailed boundaries.

III. Dense Attention for Salient Object Detection

In this section, we first present the proposed DAFE module in Sec. III-A, where two main components are introduced in Sec. III-A1 and Sec. III-A2 to explain it, respectively. Then we give the instantiated DAFE-Net in Sec. III-B.

A. DAFE Module

As described earlier, DAFE module builds dense attention by stacking a series of Feature Refinement Units (FRUs) and Dense Attentional Collection Units (DACUs). In what following, we describe each of them in detail.

(1) Feature Refinement Unit: Previous saliency models [23][39][26] usually enhance features with a few standard convolutional layers. In spite of good performance have been achieved, its refinement capacity and efficiency are limited when conducting complex refinement tasks over the high resolution features. Correspondingly, we propose to build FRUs in a *deeper* and *wider* manner, by stacking a cascaded of FRU. As shown in Fig. 3, we implement each FRU inspired by grouped convolution [63] and channel shuffling in [64][65], which has a considerably lower computational budgets and stronger representation capability for feature refinement.

Technically, each FRU begins with splitting the input high resolution features into two lower-dimensional branches, each one has half channels of the input. Then each branch applies two 1×1 group convolutions and one 3×3 depth-wise separable convolution to transform the input features. In order to gain broader context information but maintain the computational budgets, FRU adopts dilated convolution with different rates to the depth-wise separable convolution, *i.e.*, rate=2 and rate=4 for the two branches respectively. At last, the outputs from two split branches are merged using concatenation to ensure same numbers of channels for the inputs of FRU. Since each branch focuses on modeling different parts of features, the information communication across all the channels is limited, which may harm the object structure of salient regions. To this end, FRU further conducts channel shuffle over the merged features to enable cross-channel information exchange.

The proposed FRU is not only efficient but also accurate for refining salient regions. First, each FRU encodes the aggregation feature with varying dilation rates to enlarge receptive

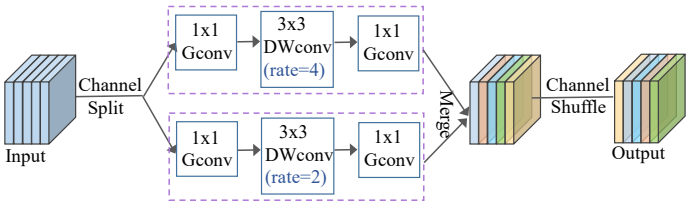


Fig. 3: Illustration of the feature refinement unit in our model. *GConv* refers to the group-wise convolution, and *DWConv* means the dilated depth-wise separable convolution.

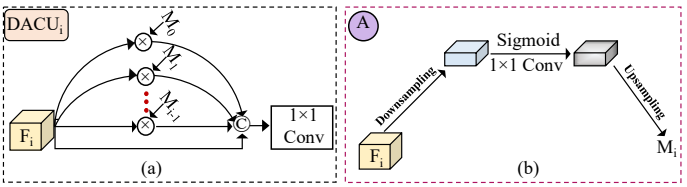


Fig. 4: (a) Structure of DACU. ⊗ refers to element-wise multiplication. ⓒ means concatenation operation. (b) Illustration of attention map generation.

field of the convolution kernels, which is able to transfer the discriminative features from the adjacent highlighted regions to the salient-related regions that have not been discovered, and thus more informative contextual can be accumulated for accurate saliency detection. Second, each FRU randomly shuffles the merged feature channels and then jointly feds them into the next unit. This enlarges the network capacity without significantly increasing complexity.

(2) Dense Attentional Collection Unit: On top of the stacked FRUs, DAFE further correlates them via a sequence of DACUs. Each DACU views the attention maps generated from all its previous FRUs as guidance, and enhances the output from current FRU by densely concatenating multiple attention-enhanced features, schematically depicted in Fig. 4. Below, we start with the attention map generation, and then introduce the structure of DACU in detail.

a) Attentional Map Generation: To properly propagate the information among different FRUs, we propose to learn an attention map from each FRU to guide the information flow. Due to each FRU produces refined features with same spatial dimension to its input features, we thus expect that the attention map enables more scale contexts to be activated, while at the same time without introducing too much computational budgets. Recent studies [23][66] have been pointed out that downsampling enables deeper network to gather more discriminative context with low computational cost. This motivates us to learn the attention map by downsampling and upsampling features along with sigmoid operation, as shown in Fig. 4 (b).

Specifically, given the output feature $\mathbf{F}_i \in \mathbb{R}^{w\times h\times c}$ from FRU$_i$ ($i > 0$ refers to the index of FRUs), DAFE first downsamples it using an average pooling layer with rate of r to produce feature $\mathbf{F}_d \in \mathbb{R}^{\frac{w}{r}\times\frac{h}{r}\times c}$, where w, h and c are width, height and channels of $\mathbf{F}_i$. On top of $\mathbf{F}_d$, DAFE learns a per-pixel, per-channel guided attention map $\mathbf{M}_i \in \mathbb{R}^{\frac{w}{r}\times\frac{h}{r}\times c}$ via a simple 1×1 convolution followed by a sigmoid function. Finally, DAFE upsamples $\mathbf{M}_i$ to the resolution of the input feature $\mathbf{F}_i$ via a bilinear interpolation operation. This attention map will be multiplied to the outputs of its all subsequent FRU to selectively control the information flow. Mathematically, we formulate the Attention Map Generation (AMG) as:

$$\mathbf{M}_i = \mathcal{U}\Big(\sigma\big(\mathrm{Conv}_{\mathbf{1\times 1}}\big(\mathcal{P}(\mathbf{F}_i)\big)\big)\Big), \tag{1}$$

where $\mathcal{U}(\cdot)$ is upsampling operation, and $\mathcal{P}$ is the pooling layer. σ is the sigmoid function $\sigma(z) = \mathbf{1}/(\mathbf{1}+\mathbf{e}^{-z})$. $\mathrm{Conv}_{1\times 1}(\cdot)$ refers to the 1×1 convolution.

Different from the widely used SENet [67] and GENet [68], which generates the attention weights using 1×1 convolution layer and fully-connected (FC) layers over aggregated global context, our AMG has three advantages for SOD. First, benefiting from the downsampling and upsampling, AMG allows each spatial location to adaptively consider its surrounding informative context as scalars in the responses from the original scale space, which effectively enlarges the fields-of-view for strengthening the salient regions. Second, instead of collecting global context, AMG only considers the context around each spatial location, avoiding some contaminating information from irrelevant regions to some extent. Last, only one 1×1 convolution layer is utilized in AMG, which is computationally efficient.

b) Structure of DACU: Fig. 4 gives the structure of one DACU. Specifically, DACU$_i$ receives $\mathbf{F}_i$ and the attentional maps from previous FRUs (*i.e.,* $\{\mathbf{M}_0, \mathbf{M}_1, ..., \mathbf{M}_{i-1}\}$) as inputs, it first enhances $\mathbf{F}_i$ and then densely connects those attention-enhanced features to update $\mathbf{F}_i$:

$$\mathbf{F}_i \leftarrow \mathcal{D}\big((\mathbf{M}_0 \otimes \mathbf{F}_i), (\mathbf{M}_1 \otimes \mathbf{F}_i) \cdots (\mathbf{M}_{i-1} \otimes \mathbf{F}_i), \mathbf{F}_i\big), \tag{2}$$

where '$\leftarrow$' indicates the updating process, $\otimes$ denotes the element-wise multiplication. $\mathcal{D}(\cdot)$ is the update function that is defined as:

$$\mathcal{D}(i, j, k) = i \text{ ⓒ } j \text{ ⓒ } k, \tag{3}$$

where ⓒ means concatenation operation. Afterward, we append a 1×1 convolutional layer to reduce the channel number of $\mathbf{F}_i$ and then fed this update feature into next FRU. With dense attentions in DAFE, features at deeper DACUs can absorb attentive information from shallower FRUs. This contributes to highlighting critical information and mitigating the interference of noised information at the final output.

(3) Overall Architecture of DAFE: Algorithm. 1 presents the detailed process of DAFE module. For an initialization feature $\mathbf{F}_0$, DAFE appends several couples of FRU and DACU for enhancing it. Specifically, it first feds $\mathbf{F}_0$ into the first FRU to produce the first refined feature $\mathbf{F}_1$, and learns an attention map $\mathbf{M}_0$ from $\mathbf{F}_0$. Taking $\mathbf{F}_1$ as well as attention map $\mathbf{M}_0$ as inputs, DACU updates $\mathbf{F}_1$ via densely collecting the attention-enhanced features, as processed in Eqn. (2). We repeat this process several times and output the final refined feature map $\mathbf{F}_t$ from the final FRU.

B. The Instantiated DAFE-Net

Based on the proposed DAFE module, we instantiate a new network, DAFE-Net, for salient object detection, as presented in Fig. 2. In what follows, we first describe its overall architecture, and then elaborate on how to train it.

Algorithm 1 Flowchart of the DAFE module

Input: Initialization feature $\mathbf{F}_0$, number of FRU t
Output: The final refined feature map $\mathbf{F}_t$

for $i \in \{0, 1, ..., (t-2)\}$ **do**
  Learning attentional map $\mathbf{M}_i$ over $\mathbf{F}_i$ using Eqn. (1)
  Sending $\mathbf{F}_i$ into FRU_{i+1} to generate $\mathbf{F}_{i+1}$
  Sending $\mathbf{F}_{i+1}$, $\{\mathbf{M}_0, \mathbf{M}_1, ...\mathbf{M}_i\}$ into DACU_{i+1}
  Update $\mathbf{F}_{i+1}$ based on Eqn. (2)
  Reduce channel of $\mathbf{F}_{i+1}$ via 1×1 convolution layer.
end for
Sending $\mathbf{F}_{t-1}$ into FRU_t and produce $\mathbf{F}_t$

(1) Framework of DAFE-Net: In our DAFE-Net, we use a common pre-trained backbone network, *e.g.,* ResNet [69] or VGG [70], as saliency feature extractor. Each backbone network contains a series of convolution layers, interleaved with pooling and fully connected layers. The features from deeper layers encode more high-level semantics, while the shallower layers carry richer, low-level details. In order to fit SOD task, similar to previous saliency models [39][23][61][25], we remove all the fully-connected layers as well as the last pooling layer in our basic feature extraction phrase. For an input image $\mathbf{I}_i \in \mathbb{R}^{\mathrm{W}\times\mathrm{H}\times3}$, taking ResNet-50 as an example, we abstract features $\mathbf{X}_i (i \in \{0,1,2,3,4\})$ at five levels with resolutions $[\frac{\mathrm{W}}{4}, \frac{\mathrm{H}}{4}]$ for $\mathbf{X}_0$ and $[\frac{\mathrm{W}}{2^{i+1}}, \frac{\mathrm{H}}{2^{i+1}}]$ for other $\mathbf{X}_i (i \in \{1,2,3,4\})$. Afterward, we append a 1×1 convolution layer over each $\mathbf{X}_i$ and reduce channels of $\mathbf{X}_i$ to $\{64, 128, 256, 256, 256\}$ respectively. Next, DAFE-Net integrates these multi-level features via a feature aggregator to obtain the aggregation feature.

Commonly, for feature fusion, the pyramid-like architecture (*e.g.*, UNet [71], FPN [44]) and the direct aggregation network (*e.g.*, HyperNet [72]) are widely used in previous saliency models [23][19][22][39][58]. The former one progressively fuses features at different levels via a top-down pathway and lateral connections, while the later one upsamples features at deeper layers to the spatial of feature at shallow layer and aggregates all of them directly. Empirically, we observe that the progressive fusion is more stable and gives better performance when collaborating with the proposed DAFE module. Following recent studies [23][22][27], We progressively integrate the deeper layer features into the shallower layer ones using FPN [44], and obtain the aggregation feature that contains both high-level semantics and low-level saliency details simultaneously. Note that our DAFE-Net is compatible with any feature aggregation network, which is further illustrated in our ablation studies.

Given the aggregation feature, DAFE-Net enhances it with the proposed DAFE module at first, it then appends a 1×1 convolution layer along with sigmoid activation function to produce the final saliency map. Fig. 7 shows some examples of saliency maps generated from *w/ DAFE* (*col* 3) and *w/o DAFE* (*col* 5) in DAFE-Net. Obviously, inaccurate saliency results, *e.g.*, over-predicted and incomplete objects, blurred object boundaries, get greatly promoted helping with DAFE module in the DAFE-Net. More comparison results would be presented in the experiment section.

(2) Training of DAFE-Net: Given the training dataset $T = \{(\mathbf{I}_i, \mathbf{Y}_i)\}_{i=1}^{N}$ with N training pairs, where $\mathbf{I}_i \in \mathbb{R}^{\mathrm{W}\times\mathrm{H}\times3}$ is the input image, and $\mathbf{Y}_i \in \mathbb{R}^{\mathrm{W}\times\mathrm{H}\times3}$ is the corresponding ground-truth map. For any pixel $\mathrm{y}_k \in \{\mathbf{Y}_i\}$, we denote $\mathrm{y}_k = 1$ as the salient pixel, and $\mathrm{y}_k = 0$ as the non-salient pixel. We train the overall network with local and global saliency prediction jointly. The global saliency prediction ensures that a good attention map can be predicted for guiding the subsequent saliency enhancement. While the local ones aim to ensure that accurate salient objects are uniformly highlighted.

Given the aggregated feature map $\mathbf{F}_0$ with $(\frac{\mathrm{W}}{4}, \frac{\mathrm{H}}{4})$ resolution and 128 channels, we fed it into the DAFE module to learn the enhanced feature map $\mathbf{F}_t \in \mathbb{R}^{\frac{\mathrm{W}}{4}\times\frac{\mathrm{H}}{4}\times128}$. Then the global saliency map $\mathbf{S}_\mathrm{g}$ and local saliency map $\mathbf{S}_\ell$ are predicted using the prediction function $\mathcal{R}$: $\{\mathrm{Conv}(1\times1, 1) \mapsto \mathcal{U}(\mathrm{W}, \mathrm{H}) \rightarrow \sigma(\cdot)\}$ over $\mathbf{F}_0$ and $\mathbf{F}_t$, respectively. Similar to Eqn.(1), $\mathcal{U}(\cdot)$ is upsampling operation, and σ is the sigmoid function. Denote the network parameter for generating the saliency map $\mathbf{S} = \{\mathbf{S}_\ell, \mathbf{S}_\mathrm{g}\}$ as $\Theta = \{\Theta_\ell, \Theta_\mathrm{g}\}$, we train our DAFE-Net by formulating the following loss function:

$$\mathcal{L} = \alpha\mathcal{L}_{bce}(\mathbf{S}_\mathrm{g}, \mathrm{Y}|\Theta_\mathrm{g}) + (1-\alpha)\mathcal{L}_{bce}(\mathbf{S}_\ell, \mathrm{Y}|\Theta_\ell), \quad (4)$$

where α is the balance factor to trade-off the importance between local and global saliency maps. We empirically set it as 0.3. The $\mathcal{L}_{bce}$ is the balanced binary cross entropy loss, which is defined as

$$\begin{aligned}\mathcal{L}_{bce}(\Theta) = &-\beta \sum_{k\in\mathbf{Y}_+} \log\mathrm{P}(\mathrm{y}_k = 1|\Theta) \\ &-(1-\beta) \sum_{k\in\mathbf{Y}_-} \log\mathrm{P}(\mathrm{y}_k = 0|\Theta),\end{aligned} \quad (5)$$

where $\mathbf{Y}_+$ and $\mathbf{Y}_-$ refer to the salient and non-salient label sets, respectively. $\beta = \mathbf{Y}_+/\mathbf{Y}_-$ is the loss weight to balance the losses between salient and non-salient pixels. The salient confidence function $\mathrm{P}(\mathbf{S}) = (1+e^{-\mathbf{S}})^{-1}$ measures how likely the pixel belongs to the salient region. Since Eq. (5) is continuously differentiate, we adopt stochastic gradient descent (SGD) method to train DAFE-Net, and it directly predicts the final saliency without any post-processing.

IV. Experiments

A. Experimental Setup

(1) Datasets: We conduct experiments on six popular benchmarks, including ECSSD [73], PASCAL-S [29], DUT-O [13], HKUIS [35], SOD [74] and DUTS [75]. ECSSD contains 1,000 semantically meaningful but structurally complex images. PASCAL-S provides 850 natural images of complex contents. DUT-O is one of the largest datasets for evaluation of SOD. It includes 5,168 images with complex object content and cluttered background, making a challenging problem for detecting salient objects with sharp boundaries. HKUIS includes 4,447 challenging images, most of which contain multiple salient objects. SOD has 300 images and it is used for image segmentation originally. DUTS contains 10,533 training images, *i.e.*, DUTS-TR, and 5,019 testing images, *i.e.*, DUTS-TE. All the datasets are human-labeled with pixel-wise ground-truth for quantitative evaluations.

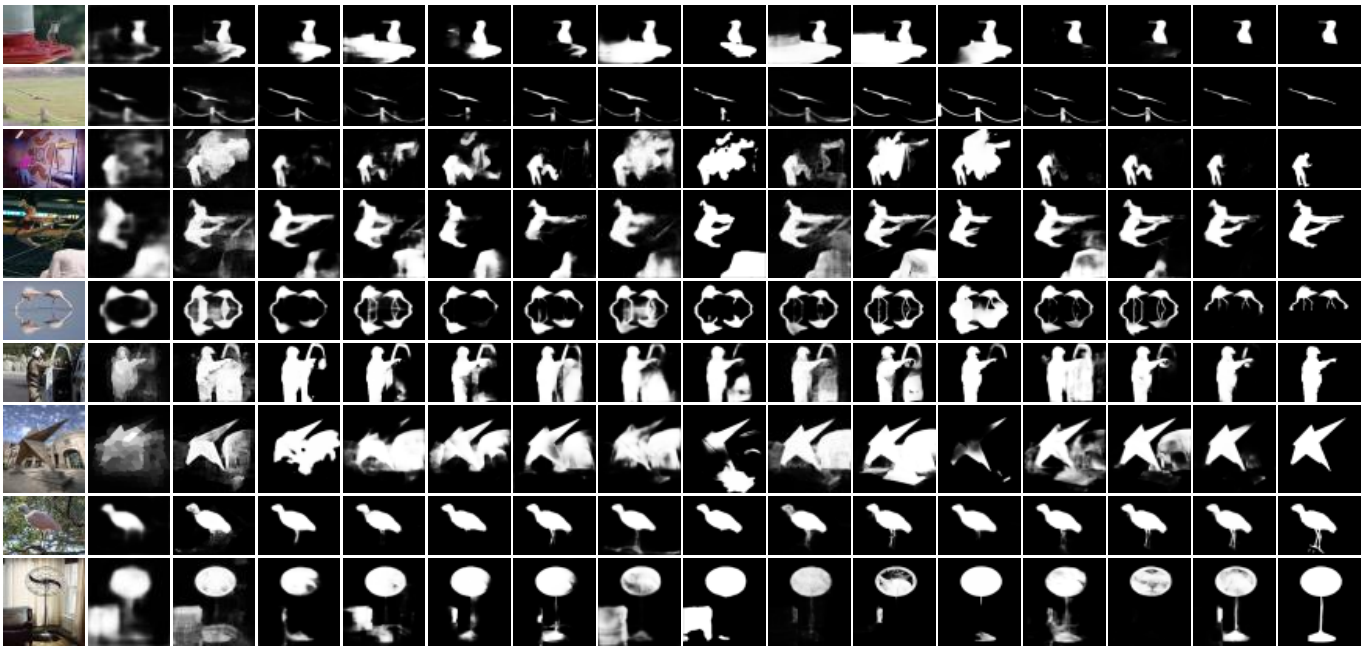


Fig. 5: Visual comparison of 13 state-of-the-art deep SOD models and our model. As can be seen, the proposed model (Ours) produces more coherent and accurate saliency maps than all other models, which are the closest to ground truth (GT).

(2) Evaluation Metrics: We adopt the widely used precision-recall (PR) curves, F-measure and mean absolute error (MAE) as evaluation metrics. The PR curves reflect the prediction precision at different recall rates compared with the ground truth. The precision and recall for one test images are computed by polarizing the predicted salient map with a threshold T_f, which is ranged in [0:1:255]. The F-measure is a harmonic mean of average precision and average recall, which is calculated by:

$$F_\beta = \frac{(1+\beta^2) \times \text{Precision} \times \text{Recall}}{\beta^2 \times \text{Precision} + \text{Recall}}. \quad (6)$$

Generally, β^2 is empirically set to 0.3 to emphasize the precision over recall. There exist two ways to present the F-measure, one is maximum F-measure (denoted as MaxF), which is adopted in studies [23][18][21][22][27][24]. The other is average F-measure (denoted as AvgF), which is adopted in literatures [39][46][19][58]. Following these methods, we report both MaxF and avgF scores for comparison.

The MAE is defined as the average pixelwise absolute difference between the saliency map S and its corresponding ground-truth G:

$$MAE = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |S(x,y) - G(x,y)|, \quad (7)$$

where W and H are width and height of an image, $S(x,y)$ is the saliency value of the pixel at (x,y).

(3) Implementation Details: Following previous works [24][23][50][22], we adopt the split of the DUTS dataset [75], *i.e.* DUTS-TR, as the training set. The training data are augmented with random rotation and horizontal flipping. We use the pre-trained Resnet-50 [69] and VGG-16 [70] as initialized feature extractor, and train the DAFE-Net with 5 and 7 FRUs respectively. During the training phrase, we set the learning rate as 0.01, the weight decay as 0.0005, and the momentum as 0.9, respectively.

B. Comparison with State-of-the-Arts

Using ResNet-50 and VGG-16 as the feature extractor, and choosing FPN [44] as the feature aggregator, we compare DAFE-Net with 21 recent FCN-based SOD methods (given in Table I for brevity). For a fair comparison, we adopt the public evaluation results provided by [76] for all the baselines.

(1) Visual Comparison: Fig. 5 gives a visual comparison of DAFE-Net (Ours) against those of several top-performance competitors. It can be seen that our model predicts saliency maps closest to the ground-truth in various challenging scenarios, *e.g.,* images with complex backgrounds and foregrounds (*row* 3, 4, 6 and 7), objects have similar appearance with backgrounds (*row* 1, 2, and 9), and multiple instances of the same object (*row* 5). Additionally, we find that our model segments the entire objects (*row* 1, 4 and 8) with fine saliency details (*row* 5, 7 and 9) quite well. These visualizations clearly demonstrate the effectiveness of the proposed network for salient object detection.

(2) F-measure and MAE Comparison: Tab. I reports MaxF, AvgF and MAE scores of our model and state-of-the-art methods. It can be seen that DAFE-Net achieves excellent results on most of the datasets, especially for the challenging PASCAL-S [29], DUTS-TE [75], HKUIS [79] and DUT-O [13] datasets, under different backbone networks. In particular, taking VGG as backbone, DAFE-Net shows a greatly improved MaxF compared to the latest method, GateNet, on the DUTS-TE (0.880 *vs* 0.870), HKUIS (0.931 *vs* 0.928) and DUT-O (0.818 *vs* 0.784). And it also improves the AvgF with a margin of 1.5%,

TABLE I: Comparison results with MaxF, AvgF and MAE over six popular SOD datasets: ECSSD [73], PASCAL-S [29], DUT-O [13], HKUIS [35], SOD [74] and DUTS-TE [75]. Results of our model are shown in **blue** and **red** under VGG and ResNet backbone setting, respectively. With different backbones, the proposed method consistently achieves better performance than the previous state-of-the-arts. Best viewed in color.

Methods	Backbone	ECSSD [73]			PASCAL-S [29]			DUTS-TE [75]			HKUIS [35]			SOD [74]			DUT-O [13]		
		MaxF	AvgF	MAE	MaxF	AvgF	MAE	MaxF	AvgF	MAE	MaxF	AvgF	MAE	MaxF	AvgF	MAE	MaxF	AvgF	MAE
VGG backbone																			
MDF [35]	VGG-16	0.832	0.807	0.105	0.768	0.709	0.146	0.730	0.673	0.094	0.861	0.784	0.129	0.787	0.721	0.159	0.694	0.644	0.092
DS [37]	VGG-16	0.882	0.826	0.122	0.765	0.659	0.176	0.777	0.633	0.090	0.865	0.788	0.080	0.784	0.698	0.190	0.745	0.603	0.120
DCL [77]	VGG-16	0.890	0.829	0.088	0.805	0.714	0.125	0.782	0.714	0.088	0.885	0.853	0.072	0.823	0.741	0.141	0.739	0.684	0.097
DHS [38]	VGG-16	0.907	0.872	0.059	0.829	0.779	0.094	0.807	0.724	0.067	0.890	0.855	0.053	0.827	0.774	0.128	-	-	-
UCF [40]	VGG-16	0.911	0.844	0.078	0.828	0.738	0.126	0.771	0.631	0.117	0.886	0.823	0.074	0.803	0.699	0.164	0.734	0.621	0.132
DSS [18]	VGG-16	0.916	0.863	0.053	0.836	0.812	0.096	0.825	0.789	0.057	0.911	0.902	0.041	0.844	0.795	0.121	0.771	0.740	0.066
NLDF [41]	VGG-16	0.905	0.878	0.063	0.831	0.782	0.099	0.812	0.739	0.066	0.902	0.872	0.048	0.841	0.791	0.124	0.753	0.684	0.080
Amulet [19]	VGG-16	0.915	0.868	0.059	0.837	0.771	0.098	0.778	0.678	0.085	0.895	0.843	0.052	0.806	0.755	0.141	0.742	0.647	0.098
RAS [28]	VGG-16	0.921	0.889	0.056	0.837	0.785	0.104	0.831	0.755	0.060	0.913	0.871	0.045	0.850	0.799	0.124	0.786	0.713	0.062
BMPM [21]	VGG-16	0.929	0.869	0.045	0.862	0.769	0.074	0.851	0.751	0.049	0.921	0.871	0.039	0.855	0.763	0.107	0.774	0.692	0.064
PAGR [58]	VGG-19	0.927	0.894	0.061	0.856	0.808	0.093	0.855	0.784	0.056	0.918	0.887	0.048	-	-	-	0.771	0.711	0.071
PiCANet [24]	VGG-16	0.931	0.885	0.047	0.868	0.804	0.077	0.851	0.749	0.054	0.921	0.870	0.042	0.853	0.784	0.102	0.794	0.710	0.068
AFNet [26]	VGG-16	0.935	0.908	0.042	0.868	0.828	0.071	0.862	0.793	0.046	0.923	0.889	0.036	0.856	0.807	0.109	0.797	0.739	0.057
PoolNet [23]	VGG-16	0.936	0.913	0.047	0.857	0.829	0.078	0.876	0.809	0.043	0.928	0.892	0.035	0.859	0.811	0.115	0.817	0.742	0.058
GateNet [61]	VGG-16	0.940	0.911	0.041	0.882	0.830	0.071	0.870	0.798	0.045	0.928	0.890	0.035	-	-	-	0.794	0.713	0.061
Ours	VGG-16	**0.940**	**0.915**	**0.044**	**0.870**	**0.836**	**0.071**	**0.880**	**0.813**	**0.045**	**0.931**	**0.902**	**0.037**	**0.859**	**0.823**	**0.082**	**0.818**	**0.743**	**0.062**
ResNet backbone																			
SRM [39]	ResNet-50	0.917	0.892	0.054	0.847	0.804	0.085	0.827	0.753	0.059	0.906	0.797	0.046	0.843	0.798	0.127	0.769	0.707	0.069
DGRL [46]	ResNet-50	0.922	0.903	0.041	0.854	0.807	0.078	0.829	0.794	0.056	0.910	0.865	0.036	0.845	0.794	0.104	0.774	0.709	0.062
R3Net [47]	ResNeXt-101	0.931	0.894	0.046	0.845	0.786	0.097	0.828	0.787	0.059	0.917	0.870	0.038	0.836	0.789	0.136	0.792	0.701	0.061
PiCANet [24]	ResNet-50	0.935	0.886	0.047	0.881	0.798	0.087	0.860	0.759	0.051	0.919	0.866	0.043	0.858	0.792	0.109	0.803	0.717	0.065
BASNet [23]	ResNet-34	0.942	0.879	0.037	0.854	0.781	0.076	0.791	0.860	0.047	0.928	0.788	0.032	0.851	0.742	0.114	0.805	0.711	0.056
CASNet [22]	ResNet-50	0.939	0.907	0.037	0.864	0.807	0.072	0.865	0.805	0.043	0.925	0.783	0.034	0.860	0.810	0.111	0.797	0.725	0.056
PoolNet [23]	ResNet-50	0.940	0.912	0.042	0.863	0.812	0.075	0.886	0.816	0.040	0.934	0.881	0.032	0.867	0.827	0.100	0.830	0.720	0.055
GateNet [61]	ResNet-50	0.942	0.913	0.040	0.882	0.809	0.071	0.881	0.815	0.042	0.933	0.888	0.034	-	-	-	0.818	0.724	0.057
CSNet [78]	ResNet-50	0.940	0.911	0.041	0.866	0.810	0.073	0.881	0.815	0.042	0.932	0.880	0.035	0.866	0.824	0.106	0.821	0.721	0.055
Ours	ResNet-50	**0.943**	**0.917**	**0.041**	**0.883**	**0.819**	**0.073**	**0.887**	**0.820**	**0.040**	**0.935**	**0.892**	**0.034**	**0.865**	**0.827**	**0.077**	**0.828**	**0.731**	**0.055**

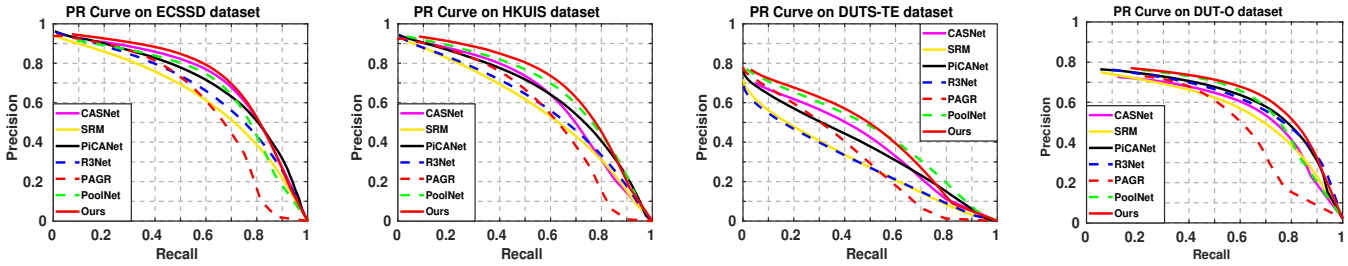
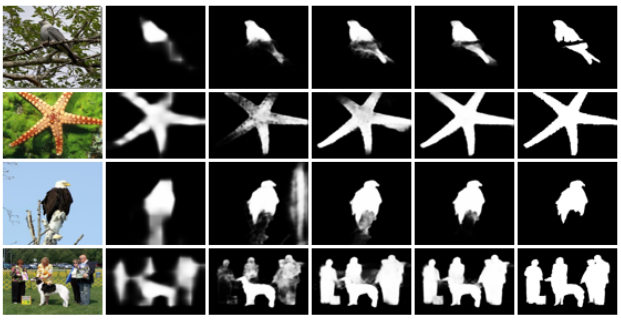


Fig. 6: Quantitative results of PR curves for DAFE-Net and other state-of-the-art models. DAFE-Net (Ours) consistently outperforms other models across all the datasets. Best viewed in color.

1.2% and 3.0% over DUTS-TE, HKUIS and DUT-O datasets, respectively. At the same time, our model generally decreases the MAE scores for most datasets. Using ResNet-50 backbone, DAFE-Net outperforms the latest method, CSNet, by a large margin in terms of MaxF (0.883 *vs* 0.866), AvgF (0.819 *vs* 0.810) on PASCAL-S, and MaxF (0.828 *vs* 0.821), AvgF (0.731 *vs* 0.721) on DUT-O, respectively. These comparison results consistently demonstrate the superior performance of DAFE-Net in complex scenes.

(3) PR Curves Comparison: Fig. 6 gives the PR curves of all methods with ResNet backbone over four datasets. The PR curves depict the rates of true positive. Usually, higher precision and slower attenuation of the curve indicate better capability of the SOD model. From Fig. 6, it can be clearly seen that our model (red solid line) keeps the precision at the highest level with increased recall, which consistently outperforms all baseline models across all datasets, convincingly demonstrating the effectiveness of our DAFE-Net.



(a) (b) (c) (d) (e) (f)

Fig. 7: Visual comparisons of our model against baseline models. (a) Input images. (b) shows saliency maps produced by ResNet-50. Following (b), (c-e) are saliency maps produced by FPN, *w/ FRUs*, and *w/ (FRUs & DACUs)* respectively. (f) Ground truths.

TABLE II: Ablation analysis for studying the effectiveness of components in DAFE module. Base refers to the basic feature aggregator. Results of our model are shown in **red** color.

No.	Base + FRUs + DACUs			DUT-O [13]		PASCAL-S [29]	
	DFA	FRUs	DACUs	MaxF ↑	MAE ↓	MaxF ↑	MAE ↓
1	✓	✗	✗	0.798	0.064	0.862	0.079
2	✓	✓	✗	0.815	0.059	0.872	0.076
3	✓	✗	✓	0.809	0.062	0.868	0.077
4	✓	✓	✓	**0.822**	**0.058**	**0.877**	**0.072**
No.	FPN	FRUs	DACUs	MaxF ↑	MAE ↓	MaxF ↑	MAE ↓
5	✓	✗	✗	0.796	0.065	0.845	0.087
6	✓	✓	✗	0.819	0.059	0.876	0.075
7	✓	✗	✓	0.811	0.061	0.858	0.077
8	✓	✓	✓	**0.828**	**0.056**	**0.883**	**0.073**

C. Ablation Study

In this section, we first analyze the effectiveness of components, *e.g.,* FRU and DACU, in the proposed DAFE module. Then, by connecting different numbers of feature refinement units after the aggregation feature, we study the effect of the depth of FRUs to the model's overall performance. Afterward, we compare different configurations of dense attention generation for better understanding the proposed DACU. All the ablation experiments are conducted with ResNet-50 backbone on the challenging DUT-O [13], SOD [74], PASCAL-S [29] and DUTS-TE [75] datasets.

(1) Effectiveness of Components in DAFE module: We first adopt different basic feature aggragators, including Direct Feature Aggregator (DFA) and Feature Pyramid Network (FPN), to obtain the aggregation feature. Then, on top of this feature, we compare three variants: *w/ FRUs*, *w/ DACUs* and *w/ (FRUs & DACUs)* to verify the effectiveness of each component respectively. Baseline *w/ FRUs* refers to the results obtained by directly connecting several feature refinement units after the aggregation feature. For *w/ DACUs*, we densely connect a set of attention-enhanced aggregation features, where the attention map is obtained from the output of each DACU, and no FRU is involved in the whole process. On top of *w/ FRUs*, the baseline *w/ (FRUs & DACUs)* corresponds to the results obtained by further collecting those FRUs with dense attentions, which is our instantiated DAFE-Net. Tab. II shows the results on two challenging datasets: DUT-O and PASCLA-S, and Fig. 7 gives the corresponding visual comparisons.

• ***w/ FRUs*:** By comparing results of basic models: DFA (No. 1 in Tab. II) and FPN (No. 4 in Tab. II), the addition of FRUs (*e.g., w/ FRUs*) obviously brings performance gain in terms of MaxF and MAE scores. In particular, through appending FRUs to the DFA, *w/ FRUs* improves the MaxF with a margin of 1.7%, 1.0% on DUT-O and PASCAL-S datasets respectively. Similarly, compared to the basic FPN, *w/ FRUs* yields performance gains of 2.3%, 3.1% in terms of MaxF on DUT-O and PASCAL-S respectively. Additionally, we observe a drop in performance (DUT-O: 0.064→0.059 and 0.065→0.059, PASCAL-S: 0.079→0.076 and 0.087→0.075) when using MAE. These results validate the effectiveness of the proposed FRU. The visual comparison between results of FPN and *w/ FRUs* can be found in Fig. 7 (c) and (d). Obviously, the incomplete and object details missed saliency maps from FPN can be well improved by *w/ FRUs*. This further confirms that FRUs are indeed improving SOD.

• **w/ *DACUs*:** By comparing *row* 1 with *row* 3 as well as *row* 5 with *row* 7 in Tab. II, utilizing dense attentions for enhancing aggregation feature uniformly improves the SOD performance, with a margin of 1.1%, 1.5% on DUT-O, and 0.6%, 1.3% on PASCAL-S w.r.t. MaxF, respectively. And a decreased in performance also can be observed in terms of MAE metric. These results demonstrate that DACUs can well boost performance for feature enhancement in SOD. Besides, by comparing *rows* 2, 4 and *rows* 6, 8 in Table II, we find that adding DACUs into *FRUs*, obtains larger MaxF values and smaller MAE scores than other settings, which further illustrates the effectiveness of the proposed DACUs.

• ***w/ (FRUs & DACUs)*:** When we look at the results of using DACUs to connect FRUs, we find that *w/ (FRUs & DACUs)* consistently achieves best performance than all other settings, across two challenging dataset. Specifically, compared to *w/ FRUs*, *w/ (FRUs & DACUs)* yields MaxF gains of 0.7%, 0.9% on DUT-O, and 0.5%, 0.7% on PASCAL-S, respectively. Similarly, compared to *w/ DACUs*, the *w/ (FRUs & DACUs)* uniformly improves the MaxF with a margin of 1.3%, 1.7% on DUT-O, and 0.9%, 2.5% on PASCAL-S, respectively. A drop in performance can be made when using MAE metric under different feature aggregators. These results indicate that the collaboration of DACUs and FRUs is crucial for obtaining better performance. Some examples of saliency maps produced by FPN, *FRUs* and *w/ (FRUs & DACUs)* are displayed in Fig. 7 (c) (d) and (f). It can be seen that complete salient objects and clear object boundaries can be well highlighted by adding DACUs into FRUs. This further illustrates the effectiveness of *w/ (FRUs & DACUs)*.

(2) Effect of the Depth of FRUs: Next, we study the effect of the depth of FRUs in our model. Using ResNet-50 and VGG-16 based FPN as basic model, we train our model (*i.e.,* DAFE-Net-R and DAFE-Net-V under these baselines) with FRU number $N \in \{2, 3, 5, 7\}$, and $N \in \{2, 4, 7, 8\}$ respectively. Fig. 8 presents the MaxF and MAE scores on the PASCAL-S, DUTS-TE, SOD and DUT-O datasets. We can gain some fundamental observations: (1) With 5 and 7

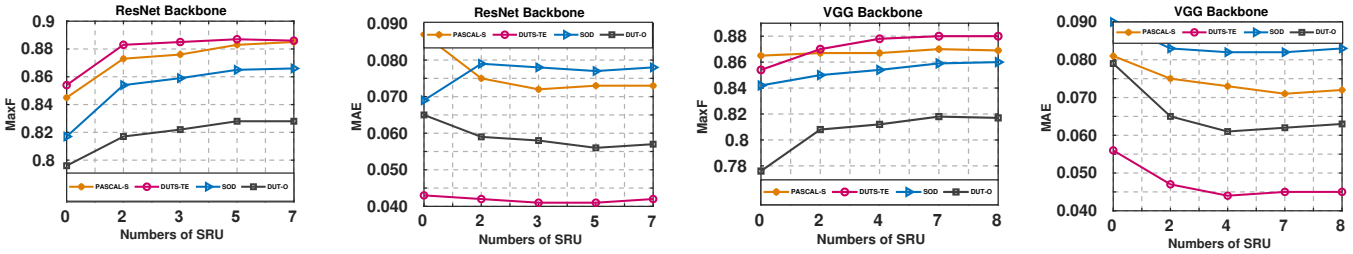
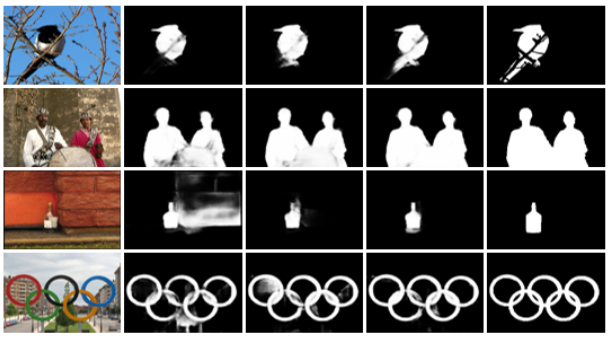


Fig. 8: Quantitative results of MaxF and MAE with different numbers of FRU under VGG-16 [70] and ResNet-50 [69] backbone, respectively.



(a) (b) (c) (d) (e)

Fig. 9: Visual comparison of DAFE-Net with different numbers of FRUs. (a) is input images and (e) is the corresponding ground truth maps. (b)-(d) show saliency maps of DAFE-Net-R implemented with $2, 3, 5$ number of FRUs, respectively.

numbers of FRUs for DAFE-Net-R and DAFE-Net-V respectively, our model performs best; (2) model's performance are gradually improved by increasing the depth of FRUs. This confirms our findings in Sec. III-A1 that a saliency inference module with larger enhancement capacity would further boost performance; (3) as the deepening of the connecting FRU, performance of DAFE-Net tends to be saturated. This may be caused by the over-fitting since limited training samples can be used in the ear of deep learning. Fig. 9 shows the corresponding saliency maps. We can see that by progressively adding the depth of FRUs, more precise salient objects can be gained. Such schema detects multiple objects, suppresses non-salient regions, sharps salient object boundaries, and produces complete salient objects effectively.

(3) Effectiveness of the Proposed AMG: To further understanding the effectiveness of the proposed AMG, we study the effect of dense attention as well as our attention generation manner, respectively. Tab. III reports the corresponding results on the challenging DUTS-TE [75] and SOD [74] datasets.

• ***Effect of dense attention***: To study the effect of dense attention, we compare the following baselines, including:

(I) **No dense attention and dense connection:** It chains 5 FRUs after the aggregation feature in sequence, where the output of an earlier FRU becomes the input of a subsequent FRU.

(II) **No dense attention but dense connection:** It densely connects 5 FRUs after the aggregation feature. Each current FRU takes outputs of all its previous FRUs as input, and it feds its output to the next FRU. No feature selection involved in the whole process.

TABLE III: Ablation analysis for studying the effectiveness of the proposed AMG two challenging datasets. The leading entries are shown in **red** color.

Module	DUTS-TE [75]			SOD [74]		
	MaxF ↑	AvgF ↑	MAE ↓	MaxF ↑	AvgF ↑	MAE ↓
(I)	0.879	0.861	0.042	0.855	0.771	0.079
(II)	0.882	0.863	0.042	0.852	0.780	0.079
(III)	**0.887**	**0.871**	**0.040**	**0.865**	**0.797**	**0.073**
(A)	0.880	0.868	0.042	0.859	0.789	0.077
(B)	0.881	0.867	0.043	0.854	0.783	0.078
(C)	0.882	0.866	0.042	0.859	0.782	0.076
(D)	0.876	0.862	0.043	0.847	0.779	0.081
(E)	**0.884**	**0.865**	**0.040**	**0.861**	**0.792**	**0.074**

(III) **Dense connection with AMG:** It densely connects 5 FRUs after the aggregation feature. For the output of current FRU, we enhance it via the AMG. Then the attention-enhanced features from current FRU and previous FRUs are sent to subsequent FRU, as discussed in Sec. III-A2b.

As shown in the top panel in Tab. III, the baseline with dense attentions and dense connections (*see* module (III) in Tab. III) achieves better performance, compared to those without dense connections and dense attentions module (*see* module (I) in Tab. III) or using only dense connections (*see* module (II) in Tab. III). In particular, the module (III) achieves significant performance improvement in terms of the F-measure compared to the module (II) on the challenging DUTS-TE (0.887 *vs* 0.879) and PASCAL-S (0.865 *vs* 0.855) datasets. Similar improvements can be observed for the MAE metric. These comparison results speak well that utilizing attentional strategy and dense connections into FRUs is essential for inferring accurate salient object.

• ***Effect of our attention generation manner*:** To validate the effectiveness of our AMG design, we compare the following baselines, including:

(A) **Vanilla attention:** It uses a 1×1 conv layer and appends a sigmoid activation function on the output of FRU to learn the corresponding attention map. Then the attention-enhanced features are fed into the corresponding DACUs. The local saliency map $\mathcal{S}_\ell$ is produced by densely connecting several FRUs and DACUs.

(B) **Channel-wise attention:** It applies the channel-wise attention proposed in [80] on the output of each FRU to learn the attentional maps. Local saliency maps are produced as same to baseline (A).

(C) **Spatial-wise attention:** It utilizes the spatial-wise attention proposed in [80] on the output of each FRU to learn

TABLE IV: Computational efficiency. #Param denotes the total number (in million) of trainable parameters. Speed is the running time (with second unit for per frame) for predicting a saliency map.

Methods	#Param	Speed	Testing on dataset of			
			DUT-O [13]		PASCAL-S [29]	
			MaxF ↑	MAE ↓	MaxF ↑	MAE ↓
VGG backbone & Input size 320×320						
Amulet [19]	33.15	8.1	0.763	0.081	0.823	0.129
DSS [18]	62.23	8.4	0.765	0.075	0.837	0.124
DHS [38]	94.04	7.3	0.759	0.084	0.846	0.115
NLDF [41]	35.94	19.2	0.748	0.096	0.835	0.126
PiCANet [24]	32.85	3.9	0.812	0.064	0.857	0.098
DAFE-Net	**23.8**	**12.6**	**0.818**	**0.062**	**0.870**	**0.071**
ResNet backbone & Input size 384×384						
SRM [39]	43.74	13.5	0.787	0.063	0.861	0.104
DGRL [46]	161.74	8.0	0.779	0.061	0.853	0.107
PoolNet [23]	68.26	17.4	0.827	0.056	0.858	0.102
PiCANet [24]	37.02	3.1	0.819	0.057	0.861	0.093
DAFE-Net	**27.9**	**10.7**	**0.828**	**0.055**	**0.883**	**0.073**

the attentional maps. Local saliency maps are produced as same to baseline (B).

(D) **Channel and spatial wise attention:** Following [67], this baseline uses two parallel branches: channel and spatial wise attention on the output of each FRU respectively, and then concatenates the outputs from two branches to generate the final attention map. Local saliency maps are produced as same to baseline (B).

(E) **Directly connect AMG:** It adopts our AMG that down-sampling and upsampling the features along with sigmoid operation for learning the attention map, as discussed in Sec. III-A2a. Local saliency maps are inferred by connecting this mask to its next FRU.

The bottom panel in Tab. III shows the experimental comparison results in terms of the F-measure and MAE scores, from which we can see that model with our AMG design outperforms other settings across all the datasets and different metrics. In particular, by comparing module (E) and other modules, the attentional map generated by AMG yields the best performance. This illustrates the effectiveness of our AMG design for distinguishing foregrounds and backgrounds. Moreover, by comparing modules (E) and (III) in Tab. III, we find that our model improves the MaxF and AvgF performance with a considerable margin on the challenging datasets. Meanwhile, it generally decreases the MAE. This further verifies the effectiveness of our attention map generation design for guiding the FRUs in DAFE module.

D. Computational Efficiency of DAFE-Net

We study the computational efficiency of DAFE-Net and several deep saliency models via PyTorch platform. Such experiment is conducted by uniformly setting the input image with size of 320×320 under VGG backbone, and 384×384 under ResNet backbone, respectively. Tab. IV reports the parameter, runtime speed, and testing performance of state-of-the-arts and DAFE-Net on two datasets. Obviously, it can be seen that our model has less parameters, higher speed and better saliency accuracy than most of the other models under the same setting. First, using VGG backbone, NLDF achieves the fastest execution time than other models, but it has more trainable parameters, which may lead to more training time. Besides, DAFE-Net yields 7.0%, 3.5% improvements using F-measure, and 3.4%, 5.5% improvements using MAE over NLDF on DUT-O and PASCAL-S datasets, respectively. Second, with respect to the ResNet based models, PoolNet [23] outperforms other models in terms of F-measure and MAE. However, by using group and depth-wise convolutions in a *spli-transfer-merge* manner, our model achieves better performance with smaller parameters than the PoolNet. In particular, DAFE-Net outperforms the PoolNet by a large margin in terms of MaxF (0.883 *vs* 0.858), MAE (0.073 *vs* 0.102) on PASCAL-S, and Speed (17.4 *vs* 10.7) respectively. These comparison results clearly illustrate the efficiency of our DAFE-Net.

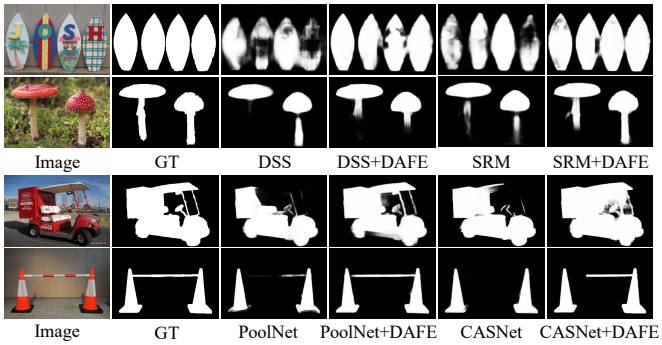


Fig. 10: Visual saliency maps generated from existing saliency models and their improved models with DAFE module. Using DAFE module, saliency maps produced by the improved models are more close to the ground truth.

TABLE V: MaxF and MAE values of the original saliency models and their improved models with DAFE module.

No.	Previous Model	DAFE	DUTS-TE [75]		PASCAL-S [29]	
			MaxF ↑	MAE ↓	MaxF ↑	MAE ↓
1	DSS [18]	✗	0.825	0.057	0.836	0.096
		✓	**0.844**	**0.054**	**0.853**	**0.090**
2	SRM [39]	✗	0.827	0.059	0.847	0.085
		✓	**0.849**	**0.054**	**0.864**	**0.082**
3	CASNet [22]	✗	0.868	0.043	0.864	0.072
		✓	**0.872**	**0.042**	**0.873**	**0.073**
4	PoolNet [23]	✗	0.886	0.040	0.863	0.075
		✓	**0.889**	**0.039**	**0.873**	**0.073**

E. Application in Existing Models

The proposed DAFE module can be applied to further improve performance of existing saliency models. To verify such observation, we first get the aggregation features from four deep saliency models, including DSS [18], SRM [39], CASNet [22] and PoolNet [23], and then append the DAFE module on them to detect the final saliency maps. We implement all the improved models by using their default settings with PyTorch. To achieve better performance, we adopt different numbers of

FRUs and DACUs on these improved models. For DSS and SRM, we train their improved models with 4 FRUs. While for CASNet and PoolNet, due to their aggregation features have contained rich discriminative information already, we adopt 2 FRUs for their feature enhancement. Additionally, when predicting the final saliency maps, we test all the original and improved models without any post-processing.

Tab. V reports the MaxF and MAE scores of the original models and its improved models over two challenging benchmark datasets. Obviously, it can be seen that DAFE helps consistently improving the performance with a large margin, by comparing results from the original models and its improved models. For example, the improved DSS model outperforms DSS itself with the gain of 1.9%, 1.7% and 0.3%, 0.6% in terms of MaxF and MAE over DUTS-TE and PASCAL-S datasets, respectively. Moreover, we also find that DAFE module contributes more to the improved DSS and SRM models than the other two models. This is mainly because the aggregation features from DSS and SRM contain more comprehensive multi-scale features, which is easy to be enhanced by exploiting DAFE module. Such observation further verifies the conclusion in Sec. IV-C1. Overall, the above observations clearly illustrate the effectiveness of DAFE for existing deep SOD models' performance improvements.

In Fig. 10, we present some examples of saliency maps produced by the original and improved models. Clearly, the improved models can better highlight the whole objects (see *col* 4, 6 in the top panel), and produce more sharp object boundaries (see *col* 4, 6 in the bottom panel).

F. Failure Case Analysis

Some failure predictions of our DAFE-Net have been shown in Fig. 11. As can be seen, these failure cases can be categorized into three circumstances in general. The first one is that the salient objects cannot be precisely segmented out in terms of the real-world scenes, as illustrated in the first row of Fig. 11. This is because most datasets assume that an image contains at least one salient object, and thus salient object is forced labeled for the human-labeling. In the second circumstance, the non-salient objects are predicted to be salient, compared to the ground truth. As shown in the middle row of Fig. 11, this case is mostly caused by the incomplete (*i.e., col* 4∼6 of the middle row) or highly controversial human-labeled (*i.e., col* 1∼3 of the middle row) annotations. The last type of failure cases is caused by objects with rich texture structure and/or complex background, as shown in the bottom row of Fig. 11. Though our approach can detect some parts of these objects, to recovery its precise object boundaries is still very difficult, due to the different data distribution between training and testing samples.

For the aforementioned problems, we argue that three possible strategies can be used to solve them. First of all, a promising solution is to improve the labeling rule, especially for the inclusion of non-salient images that are closer to the real-world scenes. Secondly, more powerful and diverse training data with both simple and complex scenes can be presented, which can substantially help improve the performance of our model on both easy and difficult datasets. Another promising solution is that designing more advanced models and then fitting more powerful feature representations to deal with the challenging inputs with complex structures. This actually is the common target for existing and future CNN-based salient object detection methods.

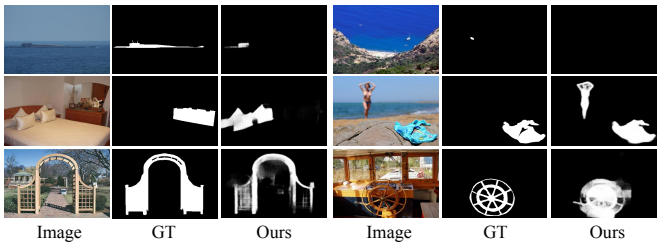


Fig. 11: Failure cases selected from multiple datasets. As can be seen, most cases are caused by complex background and controversial labeling.

V. Conclusion

In this paper, we presented a novel saliency enhancement module, DAFE, that aims to selectively pass useful information over comprehensive features for precise saliency detection. This module is implemented with two essential components: FRU and DACU. The former one recursively refines input features using a set of light-weighted convolutions, enabling more efficient training and accurate refinement performance. The latter one emphasizes on correlating the stacked FRUs via densely incorporating a set of attentional enhanced saliency clues. Collaborating with DACUs, FRUs recurrently filter redundant information and progressively refine the object structures as well as details for the salient objects. Based on the proposed DAFE module, we also instantiate an efficient DAFE-Net for salient object detection. Comprehensive experiments demonstrate that our proposed module as well as the proposed network give superior performance with better computational efficiency.

VI. Acknowledgement

This work was supported by the National Natural Science Foundation of China (No. 62072027, No. 61872032, No. 62006244), the Beijing Natural Science Foundation (Grants No. 4202057, No. 4202058, No. 4202060).

References

[1] Jian Zhao, Jianshu Li, Yu Cheng, Terence Sim, and Jiashi Feng, "Understanding humans in crowded scenes: Deep nested adversarial learning and a new benchmark for multi-human parsing," in *ACM Multimedia*, 2018, pp. 792–800.

[2] Yunchao Wei, Xiaodan Liang, Yunpeng Chen, Xiaohui Shen, Ming Ming Cheng, Jiashi Feng, Yao Zhao, and Shuicheng Yan, "Stc: A simple to complex framework for weakly-supervised semantic segmentation," *TPAMI*, vol. 39, no. 11, pp. 2314–2320, 2015.

[3] Jianbing Shen Wenguan Wang and Fatih Porikli, "Saliency-aware geodesic video object segmentation," in *CVPR*, 2015, pp. 3395–3402.

[4] He-Yu Zhou, An-An Liu, Wei-Zhi Nie, and Jie Nie, "Multi-view saliency guided deep neural network for 3d object retrieval and classification," *TMM*, vol. 22, no. 6, pp. 1496–1506, 2019.

[5] Jian Zhao, Jianshu Li, Hengzhu Liu, Shuicheng Yan, and Jiashi Feng, "Fine-grained multi-human parsing," *International Journal of Computer Vision*, , no. 5, pp. 1–19, 2019.

[6] Wenguan Wang, Jianbing Shen, and Haibin Ling, "A deep network solution for attention and aesthetics aware photo cropping," *TPAMI*, vol. 41, no. 7, pp. 1531–1544, 2019.

[7] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *ICML*, 2015, pp. 2048–2057.

[8] Fang Hao, Saurabh Gupta, Forrest Iandola, Rupesh Srivastava, Deng Li, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, and John C. Platt, "From captions to visual concepts and back," in *CVPR*, 2015, pp. 1473–1482.

[9] Hadi Hadizadeh and Ivan V. Bajic, "Saliency-aware video compression," *TIP*, vol. 23, no. 1, pp. 19–33, 2014.

[10] Fang, Yuming, Lin, Weisi, Chen, Zhenzhong, Tsai, Chia-Ming, and Chia-Wen, "A video saliency detection model in compressed domain.," *TCSVT*, pp. 27–38, 2014.

[11] Wei Feng, Ruize Han, Qing Guo, Jianke Zhu, and Song Wang, "Dynamic saliency-aware regularization for correlation filter-based object tracking," *TIP*, vol. 28, no. 7, pp. 3232–3245, 2019.

[12] Wei Zhang, Ralph R. Martin, and Hantao Liu, "A saliency dispersion measure for improving saliency-based image quality metrics," *TCSVT*, vol. PP, no. 99, pp. 1–1, 2017.

[13] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang, "Saliency detection via graph-based manifold ranking," in *CVPR*, 2013, pp. 3166–3173.

[14] Jingdong Wang, Huaizu Jiang, Zejian Yuan, Ming Ming Cheng, Xiaowei Hu, and Nanning Zheng, "Salient object detection: A discriminative regional feature integration approach," *IJCV*, vol. 123, no. 2, pp. 1–18, 2017.

[15] Ming-Ming Cheng, Niloy J. Mitra, Xiaolei Huang, Philip H. S. Torr, and Shi-Min Hu, "Global contrast based salient region detection," *TPAMI*, vol. 37, no. 3, pp. 569–582, 2015.

[16] Wangjiang Zhu, Shuang Liang, Yichen Wei, and Jian Sun, "Saliency optimization from robust background detection," in *CVPR*, 2014, pp. 2814–2821.

[17] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *TPAMI*, vol. 39, no. 4, pp. 640–651, 2014.

[18] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip Torr, "Deeply supervised salient object detection with short connections," in *CVPR*, 2017, pp. 5300–5309.

[19] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Xiang Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *ICCV*, 2017, pp. 202–211.

[20] Jiejie Zhu, Kegan GG Samuel, Syed Z Masood, and Marshall F Tappen, "Learning to recognize shadows in monochromatic natural images," in *CVPR*, 2010, pp. 223–230.

[21] Lu Zhang, Ju Dai, Huchuan Lu, You He, and Gang Wang, "A bi-directional message passing model for salient object detection," in *CVPR*, 2018, pp. 1741–1750.

[22] Zhe Wu, Li Su, and Qingming Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *CVPR*, 2019, pp. 3907–3916.

[23] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang, "A simple pooling-based design for real-time salient object detection," in *CVPR*, 2019, pp. 3917–3926.

[24] Nian Liu, Junwei Han, and Ming-Hsuan Yang, "Picanet: Learning pixel-wise contextual attention for saliency detection," in *CVPR*, 2018, pp. 3089–3098.

[25] Wenguan Wang, Shuyang Zhao, Jianbing Shen, Steven CH Hoi, and Ali Borji, "Salient object detection with pyramid attention and salient edges," in *CVPR*, 2019, pp. 1448–1457.

[26] Mengyang Feng, Huchuan Lu, and Errui Ding, "Attentive feedback network for boundary-aware salient object detection," in *CVPR*, 2019, pp. 1623–1632.

[27] Ting Zhao and Xiangqian Wu, "Pyramid feature attention network for saliency detection," in *CVPR*, 2019, pp. 3085–3094.

[28] Shuhan Chen, Xiuli Tan, Ben Wang, and Xuelong Hu, "Reverse attention for salient object detection," in *ECCV*, 2018, pp. 236–252.

[29] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille, "The secrets of salient object segmentation," in *CVPR*, 2014, pp. 280–287.

[30] Xiaohui Li, Huchuan Lu, Lihe Zhang, Ruan Xiang, and Ming Hsuan Yang, "Saliency detection via dense and sparse reconstruction," in *ICCV*, 2013, pp. 2976–2983.

[31] Yichen Wei, Wen Fang, Wangjiang Zhu, and Sun Jian, "Geodesic saliency using background priors," in *ECCV*, 2012, pp. 29–42.

[32] Zilei Wang, Dao Xiang, Saihui Hou, and Feng Wu, "Background-driven salient object detection," *TMM*, vol. 19, no. 4, pp. 750–762, 2016.

[33] W. Wang, Q. Lai, H. Fu, J. Shen, H. Ling, and R. Yang, "Salient object detection in the deep learning era: An in-depth survey," *TPAMI*, pp. 1–1, 2021.

[34] Lijun Wang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang, "Deep networks for saliency detection via local estimation and global search," in *CVPR*, 2015, pp. 3183–3192.

[35] Guanbin Li and Yizhou Yu, "Visual saliency based on multiscale deep features," in *CVPR*, 2015, pp. 5455–5463.

[36] Rui Zhao, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang, "Saliency detection by multi-context deep learning," in *CVPR*, 2015, pp. 1265–1274.

[37] Xi Li, Liming Zhao, Lina Wei, Ming-Hsuan Yang, Fei Wu, Yueting Zhuang, Haibin Ling, and Jingdong Wang, "Deepsaliency: Multi-task deep neural network model for salient object detection," *TIP*, vol. 25, no. 8, pp. 3919–3930, 2016.

[38] Nian Liu and Junwei Han, "Dhsnet: Deep hierarchical saliency network for salient object detection," in *CVPR*, 2016, pp. 678–686.

[39] Tiantian Wang, Ali Borji, Lihe Zhang, Pingping Zhang, and Huchuan Lu, "A stagewise refinement model for detecting salient objects in images," in *ICCV*, 2017, pp. 4019–4028.

[40] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Baocai Yin, "Learning uncertain convolutional features for accurate saliency detection," in *ICCV*, 2017, pp. 212–221.

[41] Zhiming Luo, Akshaya Kumar Mishra, Andrew Achkar, Justin A Eichel, Shaozi Li, and Pierre-Marc Jodoin, "Non-local deep features for salient object detection," in *CVPR*, 2017, pp. 6593–6601.

[42] Youwei Pang, Xiaoqi Zhao, Lihe Zhang, and Huchuan Lu, "Multi-scale interactive network for salient object detection," in *ECCV*, 2020, pp. 9413–9422.

[43] Saining Xie and Zhuowen Tu, "Holistically-nested edge detection," in *ICCV*, 2015, pp. 1395–1403.

[44] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, "Feature pyramid networks for object detection," in *CVPR*, 2017, pp. 2117–2125.

[45] Huaxin Xiao, Jiashi Feng, Yunchao Wei, Maojun Zhang, and Shuicheng Yan, "Deep salient object detection with dense connections and distraction diagnosis," *TMM*, vol. 20, no. 12, pp. 3239–3251, 2018.

[46] Tiantian Wang, Lihe Zhang, Shuo Wang, Huchuan Lu, Gang Yang, Xiang Ruan, and Ali Borji, "Detect globally, refine locally: A novel approach to saliency detection," in *CVPR*, 2018, pp. 3127–3135.

[47] Zijun Deng, Xiaowei Hu, Lei Zhu, Xuemiao Xu, Jing Qin, Guoqiang Han, and Pheng-Ann Heng, "R^3Net: Recurrent residual refinement network for saliency detection," in *IJCAI*, 2018, pp. 684–690.

[48] Md Amirul Islam, Mahmoud Kalash, and Neil DB Bruce, "Revisiting salient object detection: Simultaneous detection, ranking, and subitizing of mulieee transactions on image processingle salient objects," in *CVPR*, 2018, pp. 7142–7150.

[49] Keren Fu, Qijun Zhao, and Irene Yu-Hua Gu, "Refinet: A deep segmentation assisted refinement network for salient object detection," *TMM*, vol. 21, no. 2, pp. 457–469, 2018.

[50] W. Wang, J. Shen, M. Cheng, and L. Shao, "An iterative and cooperative top-down and bottom-up inference network for salient object detection," in *CVPR*, 2019, pp. 5961–5970.

[51] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun, "Large kernel matters-improve semantic segmentation by global convolutional network," in *CVPR*, 2017, pp. 4353–4361.

[52] Chen Long, Hanwang Zhang, Jun Xiao, Liqiang Nie, Shao Jian, Liu Wei, and Tat Seng Chua, "Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning," in *CVPR*, 2017, pp. 6298–6306.

[53] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin, "Convolutional sequence to sequence learning," *arXiv preprint arXiv:1705.03122*, 2017.

[54] Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu, "Recurrent models of visual attention," *NeuralPS*, pp. 2204–2212, 2014.

[55] Qi Wu, Peng Wang, Chunhua Shen, Ian Reid, and Anton Van Den Hengel, "Are you talking to me? reasoned visual dialog generation through adversarial learning," in *CVPR*, 2018, pp. 6106–6115.

[56] Hehe Fan, Linchao Zhu, Yi Yang, and Fei Wu, "Recurrent attention network with reinforced generator for visual dialog," *ACM TMCCA*, vol. 16, no. 3, pp. 1–16, 2020.

[57] Huijuan Xu and Kate Saenko, "Ask, attend and answer: Exploring question-guided spatial attention for visual question answering," *ECCV*, pp. 451–466, 2016.

[58] Xiaoning Zhang, Tiantian Wang, Jinqing Qi, Huchuan Lu, and Gang Wang, "Progressive attention guided recurrent network for salient object detection," in *CVPR*, 2018, pp. 714–722.

[59] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand, "Basnet: Boundary-aware salient object detection," in *CVPR*, 2019, pp. 7479–7489.
[60] W. Wang, J. Shen, X. Dong, A. Borji, and R. Yang, "Inferring salient objects from human fixations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 1913–1927, 2020.
[61] Xiaoqi Zhao, Youwei Pang, Lihe Zhang, Huchuan Lu, and Lei Zhang, "Suppress and balance: A simple gated network for salient object detection," in *ECCV*, 2020, pp. 35–51.
[62] Chongyi Li, Runmin Cong, Sam Kwong, Junhui Hou, Huazhu Fu, Guopu Zhu, Dingwen Zhang, and Qingming Huang, "Asif-net: Attention steered interweave fusion network for rgb-d salient object detection," *IEEE transactions on cybernetics*, vol. 51, no. 1, pp. 88–100, 2020.
[63] Andrew G. Howard, Menglong Zhu, Chen Bo, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.
[64] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *CVPR*, 2018, pp. 6848–6856.
[65] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *ECCV*, 2018, pp. 122–138.
[66] Qibin Hou, Li Zhang, Ming-Ming Cheng, and Jiashi Feng, "Strip pooling: Rethinking spatial pooling for scene parsing," in *CVPR*, 2020, pp. 4003–4012.
[67] Jie Hu, Li Shen, and Gang Sun, "Squeeze-and-excitation networks," in *CVPR*, 2018, pp. 7132–7141.
[68] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Andrea Vedaldi, "Gather-excite: Exploiting feature context in convolutional neural networks," in *NeuralPS*, 2018, pp. 9401–9411.
[69] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.
[70] K. Simonyan and A Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2015.
[71] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *ICMICCI*, 2015, pp. 234–241.
[72] Tao Kong, Anbang Yao, Yurong Chen, and Fuchun Sun, "Hypernet: Towards accurate region proposal generation and joint object detection," in *CVPR*, 2016, pp. 845–853.
[73] Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia, "Hierarchical saliency detection," in *CVPR*, 2013, pp. 1155–1162.
[74] Vida Movahedi and James H Elder, "Design and perceptual validation of performance measures for salient object segmentation," in *CVPRW*, 2010, pp. 49–56.
[75] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan, "Learning to detect salient objects with image-level supervision," in *CVPR*, 2017, pp. 136–145.
[76] Mengyang Feng, "Evaluation toolbox for salient object detection," https://github.com/ArcherFMY/sal_eval_toolbox, 2018.
[77] Guanbin Li and Yizhou Yu, "Deep contrast learning for salient object detection," in *CVPR*, 2016, pp. 478–487.
[78] Shang-Hua Gao, Yong-Qiang Tan, Ming-Ming Cheng, Chengze Lu, Yunpeng Chen, and Shuicheng Yan, "Highly efficient salient object detection with 100k parameters," in *ECCV*, 2020, pp. 702–721.
[79] Guanbin Li and Yizhou Yu, "Visual saliency based on multiscale deep features," in *CVPR*, 2015, pp. 5455–5463.
[80] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon, "Cbam: Convolutional block attention module," in *ECCV*, 2018, pp. 3–19.

Zun Li received the BS degree in Zhengzhou University, Zheng Zhou, China, in 2014. Currently, she is working toward the PhD degree in the School of Computer and Information Technology, Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing, 100044, China. She has been a visiting scholar in the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, from 2018 to 2019. Her research interests include computer vision and machine learning, saliency and co-saliency detection for image and video, segmentation, and deep learning.

Congyan Lang received her Ph.D. degree from the School of Computer and Information Technology, Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing, 100044, China, in 2006. She was a Visiting Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, from 2010 to 2011. From 2014 to 2015, she visited the Department of Computer Science, University of Rochester, Rochester, NY, USA, as a Visiting Professor. She is currently a Professor with the School of Computer and Information Technology, Beijing Jiaotong University. Prof. Lang has published more than 80 research papers in various journals and refereed conferences. Her research areas include computer vision, machine learning, object recognition and segmentation.

Liqian Liang received the BS degree in Computer Science from the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China, in 2014. Currently, she is working toward the PhD degree in the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China. She has been a visiting scholar in the School of Computer Science, The University of Adelaide, Australia, from 2016 to 2017. Her research interests include computer vision, deep learning and machine learning.

Jian Zhao received the Bachelor degree from Beihang University in 2012, the Master degree from the National University of Defense Technology in 2014, and the Ph.D. degree from the National University of Singapore in 2019. He is currently an Assistant Professor with the Institute of North Electronic Equipment, Beijing, China. His main research interests include deep learning, pattern recognition, computer vision, and multimedia analysis. He has published over 40 cutting-edge papers. He has received the Young Talent Support Project from China Association for Science and Technology, and Beijing Young Talent Support Project from Beijing Association for Science and Technology, the Lee Hwee Kuan Award (Gold Award) on PREMIA 2019, the Best Student Paper Award on ACM MM 2018, and the top-3 awards several times on worldwide competitions. He is the SAC of VALSE, and the committee member of CSIGBVD. He has served as the invited reviewer of NSFC, T-PAMI, IJCV, NeurIPS (one of the top 30% highest-scoring reviewers of NeurIPS 2018), CVPR, etc.

Songhe Feng received the PhD degree from the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, P.R. China, in 2009. He has been a visiting scholar with the Department of Computer Science and Engineering, Michigan State University, from 2013 to 2014. In 2017, he visited the Department of Computer Science, Dresden University of Technology, Germany. He is currently a full professor with the School of Computer and Information Technology, Beijing Jiaotong University. His research interests include computer vision and machine learning.

Qibin Hou received the PhD degree from the Nankai University, Tian Jin, China, in 2019. He is currently a Research Fellow with the Department of Electrical and Computer Engineering, National University of Singapore, working with Prof. J.S. Feng. His research areas include computer vision, machine learning, segmentation, and deep learning.

Jiashi Feng received the PhD degree from the National University of Singapore (NUS), in 2014. He was a post-doctoral research follow with the University of California, Berkeley. He joined NUS as a faculty member, where he is currently an assistant professor with the Department of Electrical and Computer Engineering. His research areas include computer vision, machine learning, object recognition, detection, segmentation, robust learning and deep learning. He is a member of the IEEE.